

Moduł szkoleniowy PHP

PHP: Hypertext Preprocessor

poziom średnio zaawansowany 15 godzinny

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Spis treści

1	Wprowadzenie	5
2	Cel	5
3	Opis sposobu realizacji celów	6
4	Treści kształcenia	6
5	Opis założonych osiągnięć ucznia	6
6	Sposoby osiągania celów	6
6.1	Lista czynności w zależności od roli	7
7	Szczegółowe cele kształcenia oraz treści nauczania w korelacji z treściami podstawy programowej	8
8	Propozycje kryteriów oceny i metod sprawdzania osiągnięć ucznia	12
8.1	Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne: ...	12
9	Test końcowy sprawdzający wiedzę	13
10	Lekcje	15
10.1	Lekcja 1 - Bezpieczeństwo aplikacji (Rodzaje ataków na portale internetowe, Testowanie poszczególnych zagrożeń, Zabezpieczanie aplikacji)	15
10.1.1	Cel lekcji	15
10.1.2	Treść - slajdy z opisem	16
10.1.3	Ćwiczenia	36
10.1.4	Opis założonych osiągnięć ucznia	37
10.2	Lekcja 2 - Architektura usług sieciowych (Klient – serwer, Peer to peer, Architektura hybrydowa)	37
10.2.1	Cel lekcji	37
10.2.2	Treść - slajdy z opisem	37
10.2.3	Ćwiczenia	53
10.2.4	Opis założonych osiągnięć ucznia	53
10.3	Lekcja 3 - Protokół http (budowa protokołu HTTP 1.1, nagłówki żądania i odpowiedzi, komunikacja http)	53
10.3.1	Cel lekcji	53
10.3.2	Treść - slajdy z opisem	53
10.3.3	Ćwiczenia	69
10.3.4	Opis założonych osiągnięć ucznia	69
10.4	Lekcja 4 - Modyfikacja nagłówków http (Lista nagłówków, Wykorzystanie funkcji header, Zastosowania w skryptach PHP)	69

Człowiek - najlepsza inwestycja



10.4.1	Cel lekcji.....	69
10.4.2	Treść - slajdy z opisem	70
10.4.3	Ćwiczenia.....	91
10.4.4	Opis założonych osiągnięć ucznia	91
10.5	Lekcja 5 – Cookie - informacje o użytkowniku (Budowa i działanie cookie, Używanie cookie w aplikacjach, Definiowanie i obsługa z poziomu PHP, Bezpieczeństwo).....	92
10.5.1	Cel lekcji.....	92
10.5.2	Treść - slajdy z opisem	92
10.5.3	Ćwiczenia.....	104
10.5.4	Opis założonych osiągnięć ucznia	104
10.6	Lekcja 6 - Sesje - współdzielenie informacji (Struktura i sposób działania, Kontrola sesji w PHP, Wykorzystanie sesji w aplikacjach internetowych)	104
10.6.1	Cel lekcji.....	104
10.6.2	Ćwiczenia.....	104
10.6.3	Treść - slajdy z opisem	105
10.6.4	Opis założonych osiągnięć ucznia	117
10.7	Lekcja 7 - Poczta - systemy SMTP (Systemy pocztowe w Internecie, Budowa protokołu SMTP, Komunikacja SMTP, Struktura nagłówków wiadomości, Obsługa poczty w aplikacjach PHP)	117
10.7.1	Cel lekcji.....	117
10.7.2	Treść - slajdy z opisem	117
10.7.3	Ćwiczenia.....	137
10.7.4	Opis założonych osiągnięć ucznia	137
10.8	Lekcja 8 - MIME - rozszerzenia multimedialne (Struktura nagłówków MIME, Kodowanie Base64 i Quoted-Printable, Wiadomości wieloczęściowe, Obsługa poczty multimedialnej w PHP)	137
10.8.1	Cel lekcji.....	137
10.8.2	Treść - slajdy z opisem	138
10.8.3	Ćwiczenia.....	153
10.8.4	Opis założonych osiągnięć ucznia	153
10.9	Lekcja 9 - UML 2.1 (Diagramy czynności, Projektowanie złożonych aplikacji)	153
10.9.1	Cel lekcji.....	153

Człowiek - najlepsza inwestycja

10.9.2	Treść - slajdy z opisem	154
10.9.3	Ćwiczenia	171
10.9.4	Opis założonych osiągnięć ucznia	171
10.10	Lekcja 10 - Dobre praktyki programowania w PHP	171
10.10.1	Cel lekcji.....	171
10.10.2	Treść - slajdy z opisem	172
10.10.3	Ćwiczenia	184
10.10.4	Opis założonych osiągnięć ucznia	184
10.11	Lekcja 11 - Interdyscyplinarny projekt książki teleadresowej	184
10.11.1	Cel lekcji.....	184
10.11.2	Treść - slajdy z opisem	185
10.11.3	Opis założonych osiągnięć ucznia	195

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



1 Wprowadzenie

Obecnie PHP jest jednym z najpopularniejszych języków programowania na świecie. W roku 1997 język PHP/FI obsługiwało około 50 tysięcy domen internetowych. Do roku 2005 udział PHP rósł. Później uległ stabilizacji na poziomie ok. 20 milionów domen (9% wszystkich). Obecnie jest dominującym językiem programowania i obejmuje około 75% domen internetowych.

Jest on rozwijany na zasadach licencji Open Source, więc każdy może mieć wgląd w jego kod źródłowy. Należy zaznaczyć, że PHP jest językiem skryptowym, więc aplikacje w nim napisane nie są kompilowane do postaci binarnej, lecz interpretowane przy każdym uruchomieniu.

PHP służy przede wszystkim do tworzenia dynamicznych stron WWW, czyli np. sklepów internetowych, systemów rejestracji użytkowników, statystyk itp. Zwyczajne, statyczne strony WWW to zwykłe pliki tekstowe (najczęściej z rozszerzeniem *.htm lub *.html) zawierające instrukcje zapisane w języku HTML. Przeglądarka łącząc się z serwerem, pobiera zawartość takiego pliku, a następnie odpowiednio go interpretuje. W ten sposób na ekranie swoich komputerów możemy zobaczyć wszelkiego rodzaju interaktywne elementy, odnośniki, animacje, obrazy itp.

Możliwości PHP są bardzo duże, możemy np. napisać:

- sklepy internetowe;
- systemy ankiet;
- systemy quizów;
- fora dyskusyjne;
- systemy rejestracji użytkowników;
- statystyki odwiedzin strony;
- systemy dodawania aktualności;

Jednym z najpopularniejszych portali napisanych w PHP jest Allegro.

2 Cel

Poznanie zagadnień z zakresu języka PHP oraz protokołów sieciowych takich jak HTTP, SMTP oraz MIME. Następnie wprowadzenie do mechanizmów ciasteczek i sesji. Zdobycie wiedzy na temat podstawowych architektur aplikacji internetowych oraz bezpieczeństwa stron, bądź portali internetowych. Zobrazowanie jak radzić sobie z atakami na strony, przez wykorzystanie podstawowych metod zabezpieczeń. Kurs ma również na celu zachęcić uczestnika do dalszego rozwoju, poprzez wskazanie mu bibliotek, frameworków oraz narzędzi upłynniających i ułatwiających pracę z językiem PHP.

Człowiek - najlepsza inwestycja

3 Opis sposobu realizacji celów

10 półtoragodzinnych lekcji składających się z przedstawienia materiału wraz z przykładami oraz ćwiczeniami praktycznymi. Po zakończeniu całego cyklu - przeprowadzenie egzaminu sprawdzającego wiedzę.

Dodatkowo wprowadzono lekcję 11 zawierającą część projektu interdyscyplinarnego łączącego 4 dziedziny wiedzy z zakresu informatyki (SQL, PHP, JavaScript oraz JAVA) w zakresie zaprojektowania strony internetowej, która komunikuje się z bazą danych oraz serwisu REST API do komunikacji z klientem na telefonie komórkowym.

4 Treści kształcenia

Treść kursu została podzielona na 10 bloków tematycznych – po jednym do każdej lekcji:

1. Bezpieczeństwo aplikacji (Rodzaje ataków na portale internetowe, Testowanie poszczególnych zagrożeń, Zabezpieczanie aplikacji).
2. Architektura usług sieciowych (Klient – serwer, Peer to peer, Architektura hybrydowa);
3. Protokół http (budowa protokołu HTTP 1.1, nagłówki żądania i odpowiedzi, komunikacja http);
4. Modyfikacja nagłówków http (Lista nagłówków, Wykorzystanie funkcji header, Zastosowania w skryptach PHP);
5. Cookie - informacje o użytkowniku (Budowa i działanie cookie, Używanie cookie w aplikacjach, Definiowanie i obsługa z poziomu PHP, Bezpieczeństwo);
6. Sesje - współdzielenie informacji (Struktura i sposób działania, Kontrola sesji w PHP, Wykorzystanie sesji w aplikacjach internetowych);
7. Poczta - systemy SMTP (Systemy pocztowe w Internecie, Budowa protokołu SMTP, Komunikacja SMTP, Struktura nagłówków wiadomości, Obsługa poczty w aplikacjach PHP);
8. MIME - rozszerzenia multimedialne (Struktura nagłówków MIME, Kodowanie Base64 i Quoted-Printable, Wiadomości wieloczęściowe, Obsługa poczty multimedialnej w PHP);
9. UML 2.1 (Diagramy czynności, Projektowanie złożonych aplikacji);
10. Dobre praktyki programowania w PHP
11. Projekt interdyscyplinarny – baza teleadresowa kolegów i koleżanek ze szkoły

5 Opis założonych osiągnięć ucznia

Uczeń po odbyciu kursu pozna protokoły sieciowe, mechanizm ciasteczek i sesji, nauczy się zabezpieczać swoje aplikacje, pozna podstawowe architektury aplikacji internetowych i nauczy się je projektować, będzie posiadał wytyczoną ścieżkę dalszego rozwoju.

6 Sposoby osiągania celów

Podczas procesu kształcenia, uczeń powinien wykonywać ćwiczenia oraz przykłady zawarte w materiałach edukacyjnych. Po ukończeniu szkolenia, każdy uczestnik musi zaliczyć odpowiedni test,

Człowiek - najlepsza inwestycja



który pomoże zweryfikować jego świeżo nabytą wiedzę oraz określić ewentualne luki w wiedzy, który powinien uzupełnić.

6.1 Lista czynności w zależności od roli

Wykonywane czynności są takie same dla każdej z lekcji, jedynie materiał którego dotyczą jest różny. Może się zdarzyć iż w przypadku barku materiału w danej lekcji dla danej czynności, czynność nie powinna być wykonana.

Czynności nauczyciela	Czynności ucznia
Prezentowanie przygotowanego materiału.	Sporządzanie notatek z prezentowanego materiału
Zadawanie pytań kontrolnych odnośnie już pokazanego materiału	Udzielanie odpowiedzi na pytania nauczyciela
Zadawanie pytań odnośnie materiału, który dopiero będzie pokazany, jeśli możliwe jest uzyskanie odpowiedzi w drodze dedukcji	
Objaśnianie ćwiczeń lekcyjnym, pilnowanie czasu ich wykonania przez uczniów, udzielanie wskazówek	Rozwiązywanie zadań lekcyjnych
Objaśnienie zadań na koniec lekcji, udzielanie wskazówek podczas rozwiązywania, zlecenie dokończenia zadań jako pracę domową	Odrabianie zadań domowych
Sprawdzanie ćwiczeń lekcyjnych i zadań domowych: należy zwrócić uwagę na poprawne formatowanie kodu (wcięcia) poprawne użycie konstrukcji języka dokumentowanie kodu komentarzami otrzymywanie poprawnych rezultatów	

Człowiek - najlepsza inwestycja

Czynności nauczyciela	Czynności ucznia
Zapisywanie uwag odnośnie materiałów szkoleniowych i sugestie ewentualnych korekt.	

7 Szczegółowe cele kształcenia oraz treści nauczania w korelacji z treściami podstawy programowej

Programowanie w języku PHP		
Cele kształcenia	Treści kształcenia	Podstawa programowa
Uczeń zna zagadnienia związane z bezpieczeństwem stron i portali internetowych oraz umie radzić sobie z atakami na strony przez wykorzystanie podstawowych metod zabezpieczeń.	Lekcja 1: Bezpieczeństwo aplikacji (Rodzaje ataków na portale internetowe, Testowanie poszczególnych zagrożeń, Zabezpieczanie aplikacji)	Uczeń: <ul style="list-style-type: none"> Korzysta z wbudowanych typów danych Przestrzega zasad programowania Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania Tworzy własne funkcje, procedury, obiekty, metody wybranych języków programowania Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger Kompiluje i uruchamia kody źródłowe Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera Testuje tworzoną aplikację i modyfikuje jej kod źródłowy Zabezpiecza dostęp do tworzonych aplikacji
Uczeń poznaje oraz umie rozróżnić podstawowe architektury aplikacji internetowych.	Lekcja 2: Architektura usług sieciowych (Klient – serwer, Peer to peer, Architektura hybrydowa)	Uczeń: <ul style="list-style-type: none"> Przestrzega zasad programowania
Uczeń poznaje zagadnienia z zakresu protokołów sieciowych (HTTP)	Lekcja 3: Protokół http (budowa protokołu HTTP 1.1, nagłówki żądania i odpowiedzi, komunikacja	Uczeń: <ul style="list-style-type: none"> Przestrzega zasad programowania Stosuje instrukcje, funkcje, procedury, obiekty, metody

Człowiek - najlepsza inwestycja

	http)	wybranych języków programowania
Uczeń potrafi modyfikować nagłówki protokołu http.	Lekcja 4: Modyfikacja nagłówków http (Lista nagłówków, Wykorzystanie funkcji header, Zastosowania w skryptach PHP)	Uczeń: <ul style="list-style-type: none"> Korzysta z wbudowanych typów danych Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger Kompiluje i uruchamia kody źródłowe Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera Testuje tworzoną aplikację i modyfikuje jej kod źródłowy
Uczeń potrafi tworzyć, modyfikować oraz usuwać ciasteczka oraz poznaje zasadę ich działania.	Lekcja 5: Cookie - informacje o użytkowniku (Budowa i działanie cookie, Używanie cookie w aplikacjach, Definiowanie i obsługa z poziomu PHP, Bezpieczeństwo)	Uczeń: <ul style="list-style-type: none"> Korzysta z wbudowanych typów danych Przestrzega zasad programowania Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania Tworzy własne funkcje, procedury, obiekty, metody wybranych języków programowania Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger Kompiluje i uruchamia kody źródłowe Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera Testuje tworzoną aplikację i modyfikuje jej kod źródłowy
Uczeń potrafi tworzyć,	Lekcja 6: Sesje -	Uczeń:

Człowiek - najlepsza inwestycja

modyfikować oraz usuwać sesje PHP oraz poznaje zasadę ich działania.	współdzielenie informacji (Struktura i sposób działania, Kontrola sesji w PHP, Wykorzystanie sesji w aplikacjach internetowych)	<ul style="list-style-type: none"> • Korzysta z wbudowanych typów danych • Przestrzega zasad programowania • Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania • Tworzy własne funkcje, procedury, obiekty, metody wybranych języków programowania • Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger • Kompiluje i uruchamia kody źródłowe • Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera • Testuje tworzoną aplikację i modyfikuje jej kod źródłowy • Zabezpiecza dostęp do tworzonych aplikacji
Uczeń poznaje zagadnienia z zakresu protokołów sieciowych (SMTP). Potrafi wysłać wiadomość e-mail przy pomocy skryptu napisanego w PHP.	Lekcja 7: Poczta - systemy SMTP (Systemy pocztowe w Internecie, Budowa protokołu SMTP, Komunikacja SMTP, Struktura nagłówków wiadomości, Obsługa poczty w aplikacjach PHP)	Uczeń: <ul style="list-style-type: none"> • Korzysta z wbudowanych typów danych • Przestrzega zasad programowania • Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania • Tworzy własne funkcje, procedury, obiekty, metody wybranych języków programowania • Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger • Kompiluje i uruchamia kody źródłowe • Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera • Testuje tworzoną aplikację i modyfikuje jej kod źródłowy
Uczeń poznaje	Lekcja 8: MIME -	Uczeń:

Człowiek - najlepsza inwestycja

zagadnienia z zakresu protokołów sieciowych (MIME)	rozszerzenia multimedialne (Struktura nagłówków MIME, Kodowanie Base64 i Quoted-Printable, Wiadomości wieloczęściowe, Obsługa poczty multimedialnej w PHP)	<ul style="list-style-type: none"> Korzysta z wbudowanych typów danych Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger Kompiluje i uruchamia kody źródłowe Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera Testuje tworzoną aplikację i modyfikuje jej kod źródłowy
Uczeń poznaje architektury aplikacji; zdobywa umiejętności projektowania systemów informatycznych.	Lekcja 9: UML 2.1 (Diagramy czynności, Projektowanie złożonych aplikacji)	Uczeń: <ul style="list-style-type: none"> Przestrzega zasad programowania
Uczeń poznaje możliwości swojego przyszłego rozwoju, poprzez wskazanie mu bibliotek, frameworków oraz narzędzi upłynniających i ułatwiających pracę z językiem PHP. Zna narzędzia i rozwiązania stosowane przez programistów podczas ich codziennej pracy zawodowej.	Lekcja 10: Dobre praktyki programowania w PHP	Uczeń: <ul style="list-style-type: none"> Korzysta z wbudowanych typów danych Przestrzega zasad programowania Stosuje instrukcje, funkcje, procedury, obiekty, metody wybranych języków programowania Tworzy własne funkcje, procedury, obiekty, metody wybranych języków programowania Wykorzystuje środowisko programistyczne: edytor, kompilator i debugger Kompiluje i uruchamia kody źródłowe Wykorzystuje języki programowania do tworzenia aplikacji internetowych realizujących zadania po stronie serwera

Człowiek - najlepsza inwestycja

		<ul style="list-style-type: none"> • Wykorzystuje frameworki do tworzenia własnych aplikacji • Testuje tworzoną aplikację i modyfikuje jej kod źródłowy • Zamieszcza opracowane aplikacje w internecie
--	--	---

8 Propozycje kryteriów oceny i metod sprawdzania osiągnięć ucznia

Ocena zaliczenia przedmiotu powinna być składową dwóch ocen: testu oraz ocen częściowych za ćwiczenia :

Kryteria oceny testu:

- 0-7 poprawnych odpowiedzi - 1
- 8-9 poprawnych odpowiedzi - 2
- 10-11 poprawnych odpowiedzi - 3
- 12-13 poprawnych odpowiedzi - 4
- 14-15 poprawnych odpowiedzi - 5

Kryteria oceny ćwiczeń:

- Rozwiązania ćwiczeń spełniają podstawową funkcjonalność - 3
- Rozwiązania ćwiczeń spełniają podstawową funkcjonalność oraz działają poprawnie w każdym przypadku - 4
- Spełniają powyższe wymagania oraz są zabezpieczone na błędne dane – 5

8.1 Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne:

Oceną zaliczającą kurs będzie wynik testu sprawdzającego wiedzę z zakresu kursu, ocena pracy ucznia w trakcie semestru oraz przygotowanie prostego projektu (dla chętnych).

Ocenę dopuszczającą (2) otrzymuje uczeń, który potrafi podłączyć i uruchomić skrypt na stronie internetowej, zna podstawowe pojęcia związane z językiem PHP, potrafi własnymi słowami wyjaśnić podstawowe instrukcje języka.

Ocenę dostateczną (3) otrzymuje uczeń, który spełnia wymagania oceny dopuszczającej oraz potrafi stworzyć własne skrypty pozwalające na operowanie danymi i wyświetlanie ich na stronie WWW. Zna podstawowe struktury danych w języku PHP (zmienne i tablice). Wykonuje ćwiczenia z niewielką

Człowiek - najlepsza inwestycja



pomocą nauczyciela. Potrafi wskazać podstawowe zastosowania języka PHP w tworzeniu stron internetowych

Ocenę dobrą (4) otrzymuje uczeń, który spełnia wymagania na ocenę dostateczną oraz potrafi stworzyć skrypt i podłączyć skrypty reagujące na zdarzenia na stronie internetowej, zna i potrafi wykorzystać obiektowy model programowania w języku PHP. Na lekcjach wykonuje samodzielnie zadane ćwiczenia. Umiejętnie korzysta z mechanizmów pozwalających na modyfikację witryny internetowej.

Ocenę bardzo dobrą (5) otrzymuje uczeń, który spełnia wymagania na ocenę dobrą oraz bardzo dobrze zna teorię omawianą na zajęciach, w trakcie wykonywania ćwiczeń wykazuje inicjatywę, potrafi wskazać kilka rozwiązań zadania, tworzy skrypty języka z wykorzystaniem dobrych praktyk programistycznych. Wykorzystuje funkcje/klasę dostępne w bibliotekach PEAR/PECL.

Ocenę celującą (6) otrzymuje uczeń, którego wiedza wykracza poza omawiany zakres. Np. potrafi stworzyć aplikację/grę w języku PHP. Tworzy zaawansowane interaktywne witryny www, udziela się na forach i grupach dyskusyjnych poświęconych językowi PHP.

9 Test końcowy sprawdzający wiedzę

15 zadań na 30 minutowy test sprawdzający wiedzę. Pogrubioną czcionką zaznaczono prawidłowe odpowiedzi.

1. Czym różni się usługa REST w stosunku od innych webservice'ów opartych o WSDL:
 - a. Brak stanowości zasobów**
 - b. Posiada opis formatu zapytań i odpowiedzi
 - c. Jest kompilowany
 - d. Wykorzystuje metody HTTP**
2. Która metoda z niżej wymienionych nie jest metodą HTTP w wersji 1.1:
 - a. CONNECT**
 - b. PUT
 - c. OPTIONS
 - d. TRACE
3. Jeżeli w nagłówku HTTP chcemy określić główny język dokumentu jako polski, stosujemy składnie:
 - a. Content-Language: en, pl**
 - b. Content-Language: pl, en

Człowiek - najlepsza inwestycja

- c. **Content-Language: pl**
 - d. Content-Language: en, pl, de
- 4. Jak usunąć ciasteczko o nazwie 'hello' w języku php:
 - a. **setcookie("hello", "", time()-3600);**
 - b. deletecookie("hello");
 - c. unset("hello");
 - d. unsetcookie("hello");
- 5. Czy możemy ustawić zmienną w sesji przed wywołaniem funkcji session_start:
 - a. Tak, za pomocą tablicy \$_SESSION;
 - b. Możemy, ale musimy użyć specjalnej funkcji;
 - c. **Nie;**
- 6. Podaj numer domyślnego portu SMTP?
Odp: Port o numerze 25.
- 7. Kodowania transportowe to:
 - a. **Base64**
 - b. **Quoted-Printable**
 - c. ISO 8859-2
 - d. Utf-16
- 8. Do diagramów języka UML nie zaliczamy:
 - a. Diagramu klas
 - b. Diagramu pakietów
 - c. **Diagramu konstelacji**
 - d. Diagramu czynności
- 9. Rodzajem bezpośredniego ataku na portale internetowe lub ich serwery nie jest:
 - a. PHP Injection
 - b. XSS Injection
 - c. SQL Injection
 - d. **Phishing**
- 10. Szkieletem aplikacji języka PHP nie jest:
 - a. Zend Framework
 - b. Symfony
 - c. **Composer**
 - d. Yii
- 11. Do pisania jakich systemów służy język php:
 - a. **Systemów działających po stronie serwera**
 - b. Aplikacji desktopowych
 - c. Systemów do obsługi serwera WWW
 - d. Sterowników urządzeń
- 12. Mechanizm sesji przechowuje dane (nie licząc identyfikatora) po stronie:
 - a. Klienta
 - b. **Serwera**
 - c. Serwera oraz klienta
 - d. W dowolnej bazie danych, wybranej przez użytkownika

Człowiek - najlepsza inwestycja



- e. Mechanizm sesji nie przechowuje żadnych danych
- 13. Który protokół z wymienionych poniżej, nie jest protokołem pocztowym:
 - a. **FTP**
 - b. SMTP
 - c. POP3
 - d. IMAP
- 14. Do czego służy narzędzie ping:
 - a. Jest to system zarządzania treścią
 - b. Generowania dokumentacji
 - c. **Automatyzacji instalacji w języku php**
 - d. Wysyłania poczty
- 15. Możliwą wartość type w nagłówku Content-Type protokołu MIME nie jest:
 - a. text
 - b. appliacion
 - c. **graphic**
 - d. audio

10 Lekcje



10.1 Lekcja 1 - Bezpieczeństwo aplikacji (Rodzaje ataków na portale internetowe, Testowanie poszczególnych zagrożeń, Zabezpieczanie aplikacji)

10.1.1 Cel lekcji

Celem lekcji jest poznanie rodzajów ataków na portale internetowe, opisanie możliwości testowania zagrożeń oraz podanie podstawowych technik do zabezpieczania aplikacji.

Człowiek - najlepsza inwestycja


10.1.2 Treść - slajdy z opisem


Slajd 1	  <p>PHP: Hypertext Preprocessor</p> <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany Lekcja 1 Bezpieczeństwo aplikacji</p>   <p>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</p> 	Słowa wstępu.
Slajd 2	  <p>PHP: Hypertext Preprocessor</p> <h3>Przykładowy atak</h3> <pre><html> <form> Imie: <input type="text" name="firstname">
 Nazwisko: <input type="text" name="lastname"> <input type="submit" value="Submit"> </form> </html> <?php if (isset(\$_GET['firstname'])) { echo "Imie: " . \$_GET['firstname'] . "
"; echo "Nazwisko: " . \$_GET['lastname'] . "
"; } ?></pre>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	

Wykonajmy prosty formularz HTML oraz fragment kodu PHP wyświetlający wprowadzone dane na stronie.

Człowiek - najlepsza inwestycja

Slajd
3


B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor


Przykładowy atak JS injection

Wprowadzamy zamiast imienia fragment kodu JavaScript:

```
<script>alert("Hacked");</script>
```

lub


```
<script>alert(document.cookie);</script>
```



 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Aby atak zadziałał w pliku php.ini magic_quotes_gpc musi być wyłączone:
magic_quotes_gpc = Off

Slajd
4



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

Rodzaje ataków

- przerwanie (ang. interruption)
- modyfikacja (ang. modification)
- przechwycenie (ang. interception)
- podrobienie (ang. fabrication)


 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany



Największą zaletą aplikacji internetowych jest ich powszechność i ogólnodostępność. Zaleta ta jednak może zamienić się w wadę, ponieważ, czym więcej osób ma do czegoś dostęp, tym większe ryzyko związane z atakami. Nowe technologie sieciowe, bądź te, które są jeszcze w fazie rozwoju (do których należy PHP), narażone są na luki we własnej implementacji, bądź ich niewłaściwe stosowanie. Jest wiele ataków na systemy komputerowe, jednak my zajmiemy się tylko częścią związaną z bezpośrednim dostępem do aplikacji, czyli atakami na portale internetowe. Podstawowymi typami ataków na portale są:

- Przerwanie – polega na przerwaniu świadczenia usług przez serwer danej aplikacji np. DoS lub DDoS
- Modyfikacja –

Człowiek - najlepsza inwestycja

		<p>wprowadzenie zmian w aplikacji przez osoby nieupoważnione do wykonania takowych zmian</p> <ul style="list-style-type: none"> • Przechwycenie – uzyskanie kontroli nad aplikacją, bądź jej częścią lub dostępu do obszarów/zasobów, do których nie powinny mieć dostępu osoby nieuprawnione • Podrobienie – wprowadzenie fałszywych danych do aplikacji
--	--	---


Slajd
5

PHP: Hypertext Preprocessor

Atak brute-force

- Najprostszy atak na systemy komputerowe.
- Wykorzystanie bazy danych
- Metody zapobiegawcze





PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Brute – force to najprostszy rodzaj ataku na systemy komputerowe, w tym portale internetowe. Polega na znalezieniu formularza dostępu do aplikacji, konta użytkownika, bądź bazy danych, często też znalezieniu loginu i wpisywaniu wszystkich możliwych kombinacji haseł. W wersji ulepszonej z racji tego, że dla lepszego zapamiętania hasła ludzie stosują naturalne ciągi znaków jak wyrazy, korzysta się z zewnętrznej bazy danych haseł lub słowników np. imion. Jest wiele metod zapobiegawczych tego typu atakom. Do najważniejszych zaliczamy

Człowiek - najlepsza inwestycja

w przypadku formularza logowania, który jest ogólnodostępny: losowe generatory haseł, określona z góry złożoność hasła, podpisywanie w aplikacji użytkownika inną nazwą, niż jego login (np. imieniem i nazwiskiem), blokowanie formularza dostępu po określonej ilości prób wpisania błędnego hasła (np. po trzech), systemy wymuszające zmianę hasła, co jakiś czas. W przypadku paneli administracyjnych oraz aplikacji do obsługi baz danych typu phpMyAdmin, oprócz wyżej wymienionych zabezpieczeń możemy głęboko ukryć link oraz przypisać określony adres ip dostępu do panelu, bądź zezwolić na dostęp tylko z hosta lokalnego, a nie z zewnątrz. Jest jeszcze wiele innych metod, na przykład autoryzacja innych adresów IP drogą mailową lub SMSesową.


Slajd
6

PHP: Hypertext Preprocessor

Atak brute-force

```
password P@5sword! - failed
password P@5su0rd! - failed
password P@55word! - failed
password P@55w0rd! - success!
```





PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Konsola przedstawiająca próby łamania hasła metodą brute-force – sprawdzanie wielu kombinacji haseł.

Screen ze strony <http://www.virtualizationadmin.com/articles-tutorials/terminal-services/security/brute-force-hacking-terminal-server-environments.html>

Człowiek - najlepsza inwestycja






SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Atak 'Głębokie ukrycie'







- Ukrywanie dokumentów/zasobów pod adresami, których nie jest w stanie wpisać/odgadnąć człowiek.
- Metody zapobiegawcze






PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Głębokie ukrycie to termin dotyczący bezpieczeństwa dokumentów na portalach internetowych. Umieszczamy dokument pod adresem, którego nie jest w stanie wpisać człowiek oraz nie indeksujemy go w wyszukiwarkach, w czym pomocny jest htmlowy rel="nofollow" lub ustawienia pliku robots. Metoda ataku polega na znalezieniu niezabezpieczonej ścieżki i dotarciu do dokumentów. Prosty, a mocnym zabezpieczeniem jest użycie pliku .htaccess i zabronienie dostępu osobom z zewnątrz lub bez autoryzacji po http. Możemy również zabezpieczyć dostęp z poziomu systemu operacyjnego np. dla systemów unixowych, zmieniamy właściciela katalogu na serwer apache, który ma wszystkie uprawnienia i zabieramy wszelkie uprawnienia innym użytkownikom.

Człowiek - najlepsza inwestycja


Slajd 8	  <div>PHP: Hypertext Preprocessor</div> <h3>Account lockout</h3> <ul style="list-style-type: none"> • Blokowanie kont w aplikacjach internetowych. • Metody zapobiegawcze  PHP: Hypertext Preprocessor - poziom średnio zaawansowany	Atak polega na blokowaniu kont w aplikacjach internetowych, dlatego używając wcześniej wymienionego zabezpieczenia, które blokuje dostęp do konta, warto zadbać o prostą i szybką możliwość odblokowania konta na przykład drogą mailową, bądź smsową, czy w przypadku banków i innych instytucji zaufania publicznego, telefoniczną lub za pomocą wydrukowanych wcześniej kodów.
Slajd 9	  <div>PHP: Hypertext Preprocessor</div> <h3>Web parameter tampering</h3> <ul style="list-style-type: none"> • Próba przekazania parametrów, które różnią się od tych dostępnych w aplikacji. • Metody zapobiegawcze  PHP: Hypertext Preprocessor - poziom średnio zaawansowany	Metoda polega na przekazywaniu metodą GET lub POST, parametrów które różnią się od tych dostępnych w aplikacji. Użytkownik, może mieć w ten sposób dostęp do funkcji lub danych, do których nie powinien go mieć. Warto więc zwrócić uwagę na to co przekazujemy ww. metodami oraz pomyśleć o szyfrowaniu parametrów, jak i ich walidowaniu. Warto też zwrócić uwagę na to w jaki sposób nasze strony komunikują się z poziomu JS -> PHP, jeżeli wykorzystujemy metody AJAX-owe oraz czy nasza aplikacja, nie wypisuje bezpośrednio tego co podajemy jako parametry, w


Człowiek - najlepsza inwestycja

		<p>przeciwnym wypadku możemy przekazać fragment kodu PHP, poprzez parametr i narobić szkód na portalu.</p>
Slajd 10	  <p>PHP: Hypertext Preprocessor</p> <h3>Path i information disclosure</h3> <ul style="list-style-type: none"> • Próba uzyskania szczegółowych informacji o umiejscowieniu aplikacji, konfiguracji serwera itp. • Metody zapobiegawcze  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Jest to rodzaj ataku, który sam w sobie jest nieszkodliwy, aczkolwiek może przynieść wiele korzyści osobie, która próbuje się włamać na naszą stronę. Polega on na znalezieniu informacji o umiejscowieniu naszej aplikacji na serwerze, jego ustawień lub innych ważnych informacji. Informacje te można znaleźć przy błędach aplikacji oraz w obiektach konfiguracyjnych. Zapobiegać im można na wiele sposobów np. w środowisku produkcyjnym wyłączamy wyświetlanie na ekran błędów aplikacji, piszemy własną obsługę błędów, debuggujemy aplikację, tworzymy loadery do zasobów takich jak pliki js, obrazki, arkusze styli oraz innych dokumentów, które nie wyświetlają ścieżek do zasobów i informacji o błędach w przypadku nie istnienia zasobu.</p>

Człowiek - najlepsza inwestycja

Slajd
11



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Forced browsing

- Polega na spreparowaniu adresu url, w celu wykonania operacji, bądź uzyskania informacji, do których atakująca osoba nie powinna mieć dostępu
- Metody zapobiegawcze


DAILY
 GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Atak polega na spreparowaniu adresu url, w celu wykonania operacji, bądź uzyskania informacji, do których atakująca osoba nie powinna mieć dostępu. Problem rozwiązuje odpowiednio zaimplementowany system autoryzacji lub odpowiednia architektura aplikacji np. MVC (z którego powinniśmy, korzystać w większości aplikacji). System autoryzacji powinien za każdym razem sprawdzać czy dana osoba, nie próbuje uzyskać dostępu do odpowiedniego miejsca. Aplikacja w architekturze MVC, ułatwia tę sprawę tak, że wszystko jest sterowane za pomocą odpowiedniego kontrolera, który jest dostępny tylko dla zalogowanych użytkowników.







Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
 NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
 EUROPEJSKI
 FUNDUSZ SPOŁECZNY




<p>Slajd 12</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Path traversal</h2> <ul style="list-style-type: none"> • Polega na zmianie nazwy pliku, w adresie do zasobu, który zwróciła aplikacja • Metody zapobiegawcze  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Atak ten polega na zmianie nazwy pliku, w adresie do zasobu, który zwróciła aplikacja. Zapobiegać możemy jemu na wiele sposobów, poprzez np. tworzenia osobnego katalogu (ścieżki) dla zasobu, który udostępniamy lub za pomocą szyfrowania nazw pliku, co utrudni identyfikację nazw pozostałych plików.</p>
<p>Slajd 13</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Unicode encoding</h2> <ul style="list-style-type: none"> • Polega na przesłaniu za pomocą formularza zakodowanego kodu, który w przypadku wykonania, w którymś momencie aplikacji, może wywołać niepożądane efekty • Metody zapobiegawcze  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Atak ten polega na przesłaniu za pomocą formularza zakodowanego kodu, który w przypadku wykonania, w którymś momencie aplikacji, może wywołać niepożądane efekty. Bronić się przed nim można przede wszystkim, odpowiednim przetwarzaniem formularza, nigdy nie pobieramy czystych danych z formularza i operujemy na nich. Zawsze trzeba je w odpowiedni sposób przetworzyć. Używajmy również walidacji treści pól formularza oraz określonej dziedziny znaków, która jest dostępna w odpowiednim miejscu np. Przy imieniu i nazwisku, nie potrzebne są cyfry oraz znaki inne, niż znaki alfabetu.</p>

Człowiek - najlepsza inwestycja

Slajd
 14



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

PHP injection


- Metoda polega na wstrzyknięciu kodu PHP do aplikacji, a następnie poprzez odpowiednią manipulację aplikacją uruchomienie złośliwego kodu
- Metody zapobiegawcze



 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Metoda polega na wstrzyknięciu kodu PHP do aplikacji, a następnie poprzez odpowiednią manipulację aplikacją uruchomienie złośliwego kodu. W szczególności należy uważać przy uploadowaniu plików na serwer, zawsze trzeba walidować ich rozszerzenie oraz zawartość. Przy ich otwieraniu bezpośrednio w przeglądarce należy je przeformatować i odpowiednio wyświetlić np. jako tekst, aby przypadkiem ich nie wykonać.

 Slajd
 15



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

SQL injection

- Polega na wstrzyknięciu kodu zawierającego zapytanie SQL, które następnie wykona niepożądane działanie na bazie danych.
- Metody zapobiegawcze


 DAILY GROUP

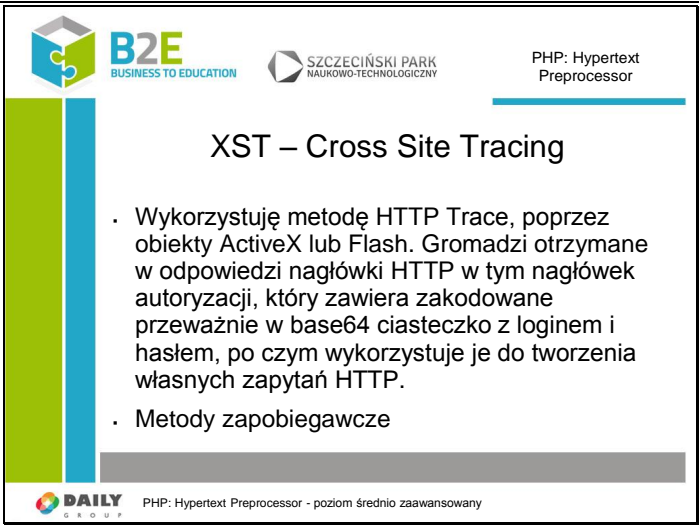
PHP: Hypertext Preprocessor - poziom średnio zaawansowany







Jest to jedna z najpopularniejszych metod ataku na serwisy internetowe, poprzez

Człowiek - najlepsza inwestycja

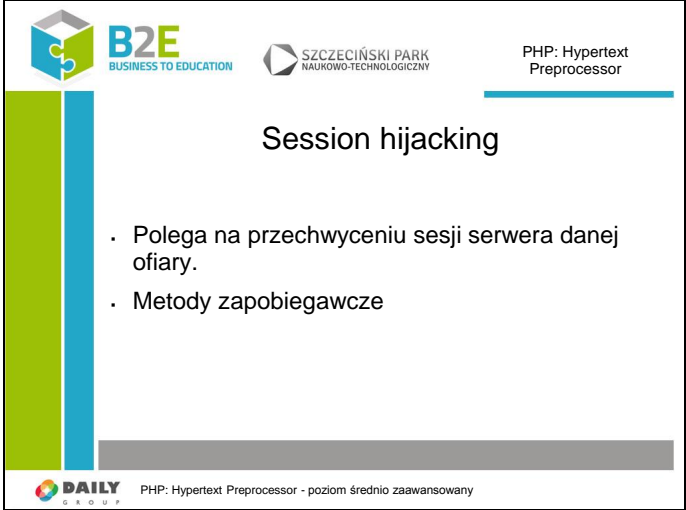
	<p>nieodpowiednie wyescapowanie treści z formularzy, bądź nieodpowiednie składanie poleceń sql. Istnieje możliwość przekazania do nich fragmentów zapytań jako parametrów, co może spowodować wykonanie nieodpowiednich akcji na bazie danych, jak np. zmianę hasła administratora, modyfikacja parametrów konfiguracyjnych lub też usunięcie ważnych danych. Jak już wcześniej wspomniano, nie należy wstawiać czystych danych z formularzy do kodu naszej aplikacji. Zalecane jest również używanie odpowiednich obiektów, bibliotek lub adapterów, jako warstwę do komunikacji z bazą danych np. PHP Database Object w skrócie PDO, który świetnie nadaje się do użycia jako warstwa do obsługi bazy danych.</p>
Slajd 16	<div> <p>SQL injection</p> <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p> </div> <p>Przerywnik – o co tutaj chodzi? Zabawna implementacja ataku SQL injection. Użytkownik samochodu znając strukturę bazy danych systemu radarowego, próbuje skasować całą zawartość tabeli „TABLICE” INSERT INTO Tablice (numer_rejestracji, mandat, kwota) VALUES ('?', '?', ?) INSERT INTO Tablice (numer_rejestracji, mandat, kwota) VALUES ('ZU 0666', 0, 0); DROP DATABASE TABLICE; -- `?,?)</p>
Slajd 17	<div> <p>XSS – Cross Site Scripting</p> <ul style="list-style-type: none"> • Polega na umieszczeniu na stronie skryptu np. js, który będzie powodował niepożądane akcje (np wysyłał dane logowania na inny serwer) • Metody zapobiegawcze <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p> </div> <p>Atak polegający na umieszczeniu na stronie skryptu np. js, który będzie powodował niepożądane akcję przez użytkowników, np. wysyłanych ich dane logowania na inny serwer lub wykonywał inne złośliwe operacje. Stosuje się go głównie w stronach, które pozwalają użytkownikom na generowanie treści. Osadzenie skryptu w stronie HTML-owej i jego wykonanie, można</p>

Człowiek - najlepsza inwestycja

		<p>wyeliminować z poziomu aplikacji poprzez np. stworzenie własnego języku znaczników lub edytora graficznego do tworzenia treści, wyświetlanie treści generowanej z odpowiednim enkodowaniem lub wyszukiwaniem skryptów w zawartości treści generowanej przez użytkownika i jego wycinaniem.</p>
Slajd 18		<p>Atak ten wykorzystuję metodę HTTP Trace, poprzez obiekty ActiveX lub Flash. Gromadzi otrzymane w odpowiedzi nagłówki HTTP w tym nagłówki autoryzacji, który zawiera zakodowane przeważnie w base64 ciasteczko z loginem i hasłem, po czym wykorzystuje je do tworzenia własnych zapytań HTTP. Obecnie najlepszym rozwiązaniem jest nie wykorzystywanie tych technologii na stronach www. Wszystko co można zrobić we flashu, można zrobić za pomocą javascriptu i mechanizmów HTML5. A jeżeli musimy używać flasha to należy zabezpieczyć proces ładowania oraz ładować go asynchronicznie.</p>



<p>Slajd 19</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>XSRF – Cross Site Request Forgery</h2> <ul style="list-style-type: none"> • Polega na wysyłaniu żądań do serwera za pomocą osób trzecich, które są w nim autoryzowane. • Metody zapobiegawcze  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Metoda polega na wysyłaniu żądań do serwera za pomocą osób trzecich, które są w nim autoryzowane. Osoba wykorzystuje uprawnienia osoby, pod którą się podszywa do wykonania akcji, wymagających zezwolenie tej osoby. Aby zapobiec tym atakom należy za każdym razem wylogować się z serwisu po zakończeniu korzystania z niego. Czas sesji serwera powinien być adekwatny do jej poufności. Żądania, które wymagają szczególnej ostrożności, jak np. przelewy, powinny wymagać dodatkowej autoryzacji np. via sms. Stosować dodatkowe ukryte dane przy autentykacji np. pseudolosową liczbę.</p>
<p>Slajd 20</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Click jacking</h2> <ul style="list-style-type: none"> • Polega na przechwyceniu zdarzenia kliknięcia użytkownika oraz wykonania innej akcji niż oczekiwana przez niego w celu uzyskania poufnych informacji lub wykonania odpowiedniej akcji. • Metody zapobiegawcze  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Atak ten polega na przechwyceniu zdarzenia kliknięcia użytkownika oraz wykonania innej akcji niż oczekiwana przez niego w celu uzyskania poufnych informacji lub wykonania odpowiedniej akcji, np. aktywacji mikrofonu, czy kamery internetowej poprzez technologie flash. W dzisiejszych czasach wiele stron używa tzw. Likejacking'u, poprzez kliknięcia w obrazki lub</p>

Człowiek - najlepsza inwestycja

		<p>inne komponenty strony, które w tajemniczy sposób lubimy, bądź udostępniamy stronę na facebooku.</p> <p>Zabezpieczyć się można poprzez wtyczki w przeglądarce lub odpowiednią jej konfigurację.</p>
Slajd 21		<p>Metoda ta polega na przechwyceniu sesji serwera danej ofiary. Możemy do tego celu użyć kodu javascript poprzez XSS, który zwróci nam ciasteczko z identyfikatorem sesji osoby zalogowanej do serwisu. Zabezpieczyć się można, korzystając z zapamiętywania adresu IP danej sesji, połączenia poprzez SSL lub enkrypcję danych przesyłanych w odpowiedziach i żądaniach. Można również zaimplementować własny mechanizm sesji, oparty np. o bazę danych.</p>

Człowiek - najlepsza inwestycja


Slajd
22



PHP: Hypertext Preprocessor

Session fixation



- Polega na wykorzystaniu identyfikatora sesji, uzyskanego np. przy pomocy ataku session hijacking i użycie go poprzez wstawienie w odpowiednie miejsce (np. adres URL czy parametr POST)
- Metody zapobiegawcze



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Metoda ta jest metodą pochodną, poprzedniej. Po uzyskaniu identyfikatora sesji, należy znaleźć sposób na umieszczenie w odpowiednim miejscu identyfikatora sesji, czyli obejście wcześniej wspomnianych w poprzedniej metodzie zabezpieczeń, w sposób zależny od metody zabezpieczenia.


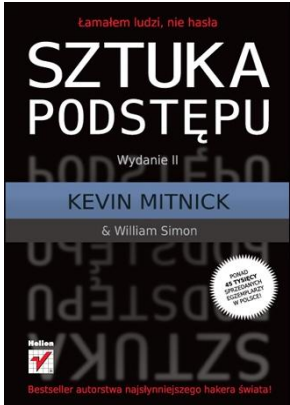
Slajd
23



PHP: Hypertext Preprocessor

Phishing

- Polega na podszycie się pod zaufaną osobę, instytucję lub firmę, w celu wyłudzenia poufnych danych np. loginu i hasła użytkownika.
- Metody zapobiegawcze




PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Atak polega na podszycie się pod zaufaną osobę, instytucję lub firmę, w celu wyłudzenia poufnych danych np. loginu i hasła użytkownika. Osoba atakująca może np. podać się za firmę hostingową, po wcześniejszym sprawdzeniu dostawcy usług internetowych i poprzez odpowiedni schemat konwersacji, uzyskać odpowiednie dane. Popularność metoda ta zawdzięcza najsłynniejszemu hakerowi w historii informatyki Kevinowi Mitnickowi, który jak twierdzi łamał ludzi, a nie hasła. Metoda ta bazuje na socjotechnice, czyli metodzie manipulacji osobami w kontakcie bezpośrednim. Zabezpieczenie przed nią jest oczywiste, nigdy nie wolno podawać żadnych danych poufnych osobom z zewnątrz, gdyż nawet jak same firmy często zaznaczają, nikt z obsługi o takie dane prosić nie


Człowiek - najlepsza inwestycja

będzie.

Slajd
24



B2E
BUSINESS TO EDUCATION

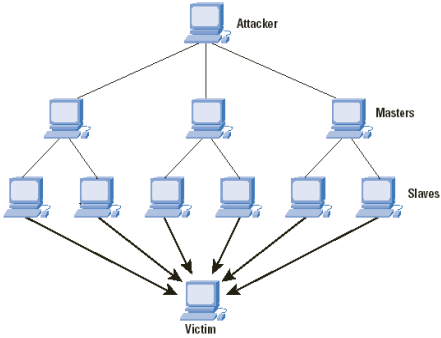


SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP: Hypertext
Preprocessor

Ataki DoS i DDoS

- polegają na zablokowaniu serwera, zasypując go ogromną ilością żądań, najczęściej wykorzystuje się do tego odpowiednie oprogramowanie, np. boty lub oprogramowanie do testowania wydajności serwera.
- Metody zapobiegawcze



Źródło: www.cisco.com



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Ataki DoS i DDoS, czyli odpowiednio Denial of Service i Disturbed Denial of Service, polegają na zablokowaniu serwera, zasypując go ogromną ilością żądań, najczęściej wykorzystuje się do tego odpowiednie oprogramowanie, np. boty lub oprogramowanie do testowania wydajności serwera. Serwer otrzymując ogromną liczbę żądań, nie jest w stanie przetwarzać kolejnych co powoduje wstrzymanie jego pracy lub po wyczerpaniu pamięci operacyjnej jego zawieszenie. Przerwa w pracy serwera w niektórych przypadkach, może spowodować ogromne straty finansowe. Nie ma skutecznej metody zapobiegania tej metodzie.
(grafika z www.cisco.com)

Człowiek - najlepsza inwestycja



Testowanie zagrożeń

Jak testować bezpieczeństwo swojego portalu?

- Stosując metody automatyczne, np. programy analizujące znane luki (Burp site, WebScarab, w3af, Nikto Web Scanner)
- Próbując dokonać ataku samodzielnie, znając własny kod łatwiej jest przeprowadzić atak
- Śledząc nowości dotyczące znalezienia nowych luk przez społeczność zajmującą się bezpieczeństwem



Jest wiele narzędzi służących do testowania aplikacji internetowych, jedne są bardziej skuteczne, drugie mniej. Należy pamiętać o tym, że testują one tylko te ataki, które są im znane. Nie są one jednak w stanie przetestować wszystkich luk w aplikacjach. Do tego celu najlepszy jest człowiek, który jest istotą kreatywną i jest w stanie znajdować luki nikomu wcześniej nie znane oraz często publikować wynik swoich badań w internecie np. na swoim blogu. Dlatego najlepszą metodą testów jest połączenie trzech czynników: programy analizujące wszystkie znane luki, wiedza ludzi, którzy szukają nowych dziur w technologiach, aplikacjach oraz architekturach, jak i własny wkład, gdyż jako programiści, znamy najlepiej swoje aplikacje i rozumiemy ich potencjalne słabe punkty.





Ćwiczenia praktyczne

Proszę wykonać formularz HTML zawierający pole z adresem e-mail oraz tabelę w bazie danych z kluczem głównym „id” (autoincrement int) oraz polem „adres”. Następnie przygotuj skrypt PHP, który będzie wyświetlał zawartość stworzonej tabeli, oraz umożliwiał dodanie nowego adresu e-mail do bazy. Na tak przygotowanym materiale proszę wykonać atak SQL Injection tak aby:

- osoba atakująca skasowała tabelę w bazie danych
- osoba atakująca podmieniła adres e-mail innego użytkownika o znanym adresie

Proponowane dodatkowe zadania związane z tym ćwiczeniem:

- Zabezpiecz skrypt przed atakiem poprzez dodanie walidacji pola formularza



Gotowe rozwiązania tego zadania znajdują się w plikach:

- skrypt.sql – skrypt tworzący tabelę w bazie danych
- sqlInjection.php – formularz HTML, połączenie do bazy danych, dodawanie adresu e-mail do bazy, wyświetlenie istniejących danych
- przykłady ataku SQL Injection – kasowanie tabeli, zmiana adresu dla istniejącego użytkownika

Do prawidłowego przetestowania tego kodu w pliku php.ini trzeba wyłączyć zabezpieczenie poprzez wyłączenie flagi magic_quotes_gpc = Off

Proszę upewnić się również, czy php posiada możliwość użycia php_mssql.dll potrzebnego do połączenia się z bazą danych (plik php.ini, linijka extension=php_mssql.dll powinna być odkomentowana).





Ćwiczenia praktyczne

Proszę przygotować skrypt PHP, dzięki któremu możliwe będzie wykonanie ataku typu brute-force, polegające na złamaniu hasła zakodowanego algorytmem MD5. Zakodowane hasło to:

- 098f6bcd4621d373cade4e832627b4f6

Proponowane dodatkowe zadania związane z tym ćwiczeniem:

- zmień złożoność hasła i zbadaj jaki ma to wpływ na czas łamania hasła
- zmień sposób kodowania hasła np na SHA-1 i zbadaj czy takie samo hasło jest łamane w tym samym czasie dla 2 różnych algorytmów (np MD5 i SHA-1)





Gotowe rozwiązania tego zadania znajdują się w plikach:

- brute.php – przy pomocy trudniejszej metody rekursji (funkcja wywołuje samą siebie)
- Brute1.php – przy pomocy łatwiejszej metody dla hasła 4 literowego



Slajd
28

 B2E
BUSINESS TO EDUCATION

 SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne

Proszę wykonać formularz HTML zawierający dowolne pole, oraz fragment skryptu PHP odczytujący zawartość tego pola metodą GET. Następnie proszę zrealizować atak JS Injection poprzez wpisanie w pole formularza kodu JS, który wykona się w przeglądarce.

Proponowane dodatkowe zadania związane z tym ćwiczeniem:

- Zabezpiecz skrypt przed atakiem poprzez dodanie walidacji pola formularza

 DAILY
G R O U P

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Gotowe rozwiązania tego zadania znajdują się w pliku:

- atak.php

Do prawidłowego przetestowania tego kodu w pliku php.ini trzeba wyłączyć zabezpieczenie poprzez wyłączenie flagi magic_quotes_gpc = Off

Człowiek - najlepsza inwestycja






KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego


Slajd
29



PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne

Przy pomocy skryptu curlLoop.php wraz z kolegami z klasy, sróbućcie wykonać atak DoS na komputer jednego z kolegów. Jeden komputer udostępnia dowolny zasób pod podanym adresem HTTP, pozostali uczniowie z wielu komputerów wykonują skrypt, który wielokrotnie w pętli próbuje ten zasób pobrać.



 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Gotowe rozwiązania tego zadania znajdują się w pliku:

- curlLoop.php


Aktualnie skrypt wykonuje próbę pobrania zasobu w pętli 10 razy, można próbować manipulować tym parametrem aby bardziej dociążyć atakowany komputer.

Slajd
30




PHP: Hypertext
Preprocessor

Dziękujemy za uwagę



Człowiek - najlepsza inwestycja

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Słowa na zakończenie.

10.1.3 Ćwiczenia

Ćwiczenia należy wykonać z każdym uczniem z osobna.

Człowiek - najlepsza inwestycja



Ćwiczenie 1 Wymień i opisz trzy wybrane przez siebie ataki na portale internetowe.

Ćwiczenie 2 Opisz metody zapobiegania atakom na portale internetowe, dla wcześniej wymienionych rodzajów ataków.

Ćwiczenie 3 Utworzyć formularz zawierający jedno pola tekstowe oraz napisać skrypt, który dokona jego walidacji, do wyboru (przydziela nauczyciel):

- Pole z imieniem (tylko znaki alfabetu)
- Numer telefonu (cyfry spacje i znak +)
- Adres (tylko znaki alfanumeryczne)

10.1.4 Opis założonych osiągnięć ucznia


Po tej lekcji uczniowie będą posiadać wiedzę o popularnych atakach na portale internetowe, poznają podstawowe metody prewencji oraz będą widzieć jak testować bezpieczeństwo w swoich aplikacjach.

10.2 Lekcja 2 - Architektura usług sieciowych (Klient – serwer, Peer to peer, Architektura hybrydowa)




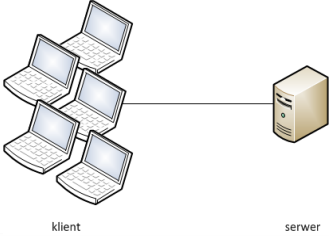

10.2.1 Cel lekcji

Celem lekcji jest wyjaśnienie podstawowych architektur aplikacji sieciowych: klient – serwer, p2p oraz architektury mieszanej. Nauczyciel opisuje i pokazuje ich schematy. Opis podstawowych architektur zorientowanych na usługi SOA. Opis jej najważniejszych implementacji SOAP, WebService, REST, DCOM, CORBA, WCF.

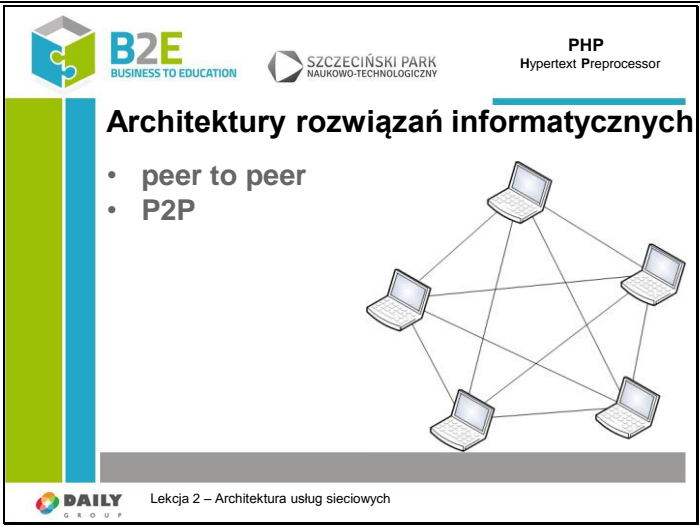
10.2.2 Treść - slajdy z opisem

Slajd 1		<p>Na lekcji zostaną omówione najczęściej spotykane architektury rozwiązań informatycznych, ich wady i zalety.</p> <p>W kolejnej części omówione będą tematy SOA i najczęściej używane architektury usług sieciowych.</p>
------------	---	---



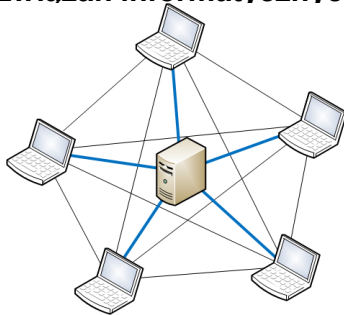

Człowiek - najlepsza inwestycja

<p>Slajd 2</p>	<div data-bbox="284 197 967 712">  <h2 data-bbox="379 293 967 327">Architektury rozwiązań informatycznych</h2> <ul data-bbox="379 349 611 450" style="list-style-type: none"> • client – server • peer to peer • hybrid <div data-bbox="304 674 667 701">  Lekcja 2 – Architektura usług sieciowych </div> </div>	<p>Wyróżniamy architektury:</p> <ul style="list-style-type: none"> • klient - serwer - najbardziej popularna architektura oprogramowania w której klient (aplikacja kliencka) łączy się z serwerem (farmą serwerów), • peer to peer - rozproszona architektura, której założeniem jest możliwość połączenia się wszystkich aplikacji między sobą • hybrydowa - szerokie pojęcie np. architektura mieszana klient - serwer + peer to peer.
<p>Slajd 3</p>	<div data-bbox="284 851 967 1357">  <h2 data-bbox="379 936 967 969">Architektury rozwiązań informatycznych</h2> <ul data-bbox="379 992 611 1025" style="list-style-type: none"> • client – server <div data-bbox="616 1048 946 1283">  </div> <div data-bbox="304 1317 667 1344">  Lekcja 2 – Architektura usług sieciowych </div> </div>	<p>Wyróżniamy architektury:</p> <ul style="list-style-type: none"> • klient - serwer - najbardziej popularna architektura oprogramowania w której klient (aplikacja kliencka) łączy się z serwerem (farmą serwerów), <p>Przykładem jest serwer poczty elektronicznej (klient – np. outlook, serwer – exchange), serwery WWW (klient-przeglądarka, serwer-serwer WWW), serwery plików czy serwery aplikacji.</p> <p>Stroną inicjującą komunikację jest klient. Stroną wykonującą usługę jest serwer.</p> <p>W zależności od zastosowań wyróżnia się wiele typów architektury klient serwer:</p> <ul style="list-style-type: none"> • dwuwarstwowa (klient komunikuje się z serwerem np. serwer plików), • trójwarstwowa (klient łączy się z serwerem aplikacji (warstwa druga) który przetwarza żądania, korzystając z baz danych (warstwa trzecia) lub innych zasobów odpowiada na żądanie,

Człowiek - najlepsza inwestycja

		<ul style="list-style-type: none"> wielowarstwowa w której logika biznesowa została zaimplementowana na wielu poziomach. <p>Zalety: wspólne dane w jednym miejscu, zapewniona spójność i brak konfliktów wersji, łatwość zabezpieczenia danych Wady: ograniczona wydajność (można stosować klastry wydajnościowe), awaria serwera powoduje niedostępność aplikacji/danych (stosuje się klastry niezawodnościowe)</p>
Slajd 4		<ul style="list-style-type: none"> peer to peer - rozproszona architektura której założeniem jest możliwość połączenia się wszystkich aplikacji między sobą <p>P2P zakłada możliwość połączenia się wszystkich aplikacji między sobą. Każdy element architektury jest na tym samym równorzędym z innymi poziomem. Każdy element architektury jest zatem i klientem i serwerem. W przypadku odłączenia jednego elementu transmisja nadal jest prowadzona z wykorzystaniem zasobów innych elementów.</p> <p>Wadą takiego rozwiązania jest konieczność przepytывania wszystkich elementów o potrzebne zasoby (np. pliki).</p>

Człowiek - najlepsza inwestycja

<p>Slajd 5</p>	<div data-bbox="300 197 941 280">  B2E BUSINESS TO EDUCATION </div> <div data-bbox="542 235 715 268">  SZCZECIŃSKI PARK NAUKOWO-TECHNOLOGICZNY </div> <div data-bbox="790 219 938 257"> PHP Hypertext Preprocessor </div> <div data-bbox="379 286 970 324"> <h3>Architektury rozwiązań informatycznych</h3> </div> <div data-bbox="379 342 593 409"> <ul style="list-style-type: none"> • Hybrid • P2P + server </div> <div data-bbox="609 318 954 631">  </div> <div data-bbox="306 672 667 705">  Lekcja 2 – Architektura usług sieciowych </div>
--------------------	--

Człowiek - najlepsza inwestycja

Slajd
7

 **B2E**
BUSINESS TO EDUCATION

 SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Architektura usług sieciowych

- SOA – Service Oriented Architecture
- Usługa sieciowa




 Lekcja 2 – Architektura usług sieciowych


SOA jest koncepcją tworzenia systemów informatycznych w której poszczególne moduły oprogramowania (aplikacje SOA) działają autonomicznie udostępniając zestaw usług do komunikacji.

Usługi implementują funkcjonalność potrzebną klientowi lub innej aplikacji SOA. Aplikacja SOA działa na zasadzie skrzynki, przykrywając szczegóły implementacji, zapewniając ściśle określoną funkcjonalność.

Usługą może być każdy, niezależnie działający program zapewniający poprzez interfejs dostęp do określonej funkcjonalności.

Slajd
8

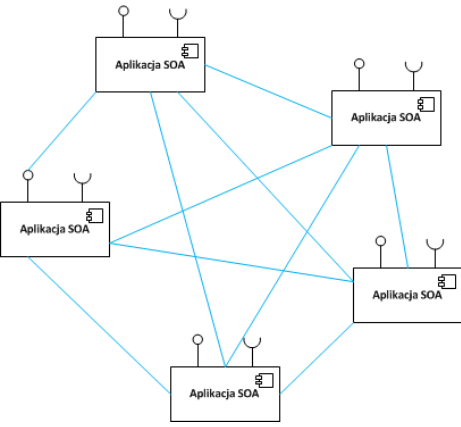
 **B2E**
BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Architektura usług sieciowych

- SOA




 Lekcja 2 – Architektura usług sieciowych


Człowiek - najlepsza inwestycja

W szczególności, możliwe jest współdziałanie między sobą wielu aplikacji SOA (np. systemy zintegrowane). Implementacja połączenia każdy z każdym jest bardzo skomplikowana i pracochłonna...

Slajd
9



B2E
BUSINESS TO EDUCATION

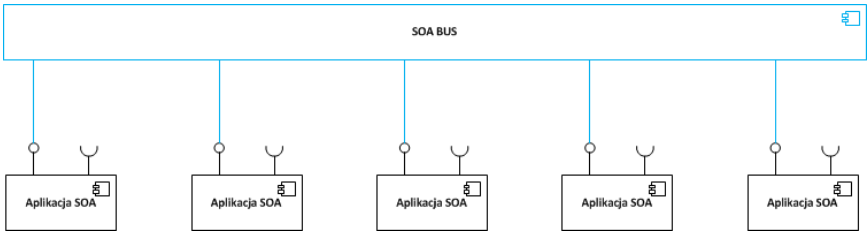



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Architektura usług sieciowych

- **SOA BUS**





Lekcja 2 – Architektura usług sieciowych

...dlatego stosuje się rozwiązania typu ESB (Enterprise SOA Bus) zapewniające możliwość rejestracji usług każdej aplikacji SOA oraz definiowania procesów integracyjnych pomiędzy aplikacjami.

Slajd
10



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Implementacje SOA

- SOAP
- WebServices
- REST
- DCOM
- CORBA
- WCF



Lekcja 2 – Architektura usług sieciowych

Komunikacja między aplikacjami SOA i realizacja interfejsów może zostać wykonana na wiele sposobów. Przybliżymy następujące implementacje SOA:

- SOAP
- WebService
- REST
- DCOM
- COBRA
- WCF

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
11





PHP
Hypertext Preprocessor

Implementacje SOA

SOAP

- Protokół zdalnego dostępu
- Format na bazie XML

SOAP-ENV: Envelope

SOAP-ENV: Header

SOAP-ENV: Body



Lekcja 2 – Architektura usług sieciowych

SOAP – (Simple Object Access Protocol) protokół służący do komunikacji między systemami wykorzystujący format XML. Dokument SOAP ma określoną strukturę i składa się z:

Koperty (envelope)

Nagłówek (header)

I ciała komunikatu (body) – w którym zawarto informację jaką usługę i metodę należy wywołać

Transmisję SOAP można realizować protokołami HTTP lub RPC

Człowiek - najlepsza inwestycja




KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Slajd
12

 **B2E**
BUSINESS TO EDUCATION

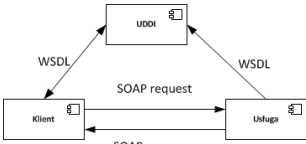
 SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP
Hypertext Preprocessor

Implementacje SOA

WebServices

- Protokół HTTP/HTTPS
- Komunikat XML (SOAP)
- Opis interfejsu WSDL




 **DAILY**
GROUP


Lekcja 2 – Architektura usług sieciowych

Web service jest metodą komunikacji pomiędzy dwoma urządzeniami/aplikacjami w sieci z wykorzystaniem protokołu HTTP/HTTPS. Formatem komunikatu jest SOAP (Simple Object Access Protocol) – dokument XML o określonej strukturze

Jest wyposażona w mechanizm opisu usługi WSDL (Web Services Description Language). Pozwala on na opis usługi w taki sposób żeby aplikacja kliencka (klient) mógł poprawnie wczytać format zapytania i format odpowiedzi. Definicje poszczególnych usług i metod mogą być częścią usługi albo mogą zostać wyniesione do pośrednika (broker) UDDI (Universal Description Discovery and Integration).

Slajd
13

 **B2E**
BUSINESS TO EDUCATION

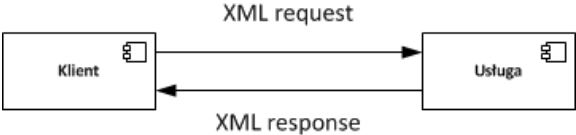
 SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP
Hypertext Preprocessor

Implementacje SOA

REST

- Protokół HTTP/HTTPS
- Standardowe operacje HTTP: GET, PUT, POST, DELETE
- Komunikat XML



 **DAILY**
GROUP


Lekcja 2 – Architektura usług sieciowych

Człowiek - najlepsza inwestycja


Odmianą Web service jest REST (Representational State Transfer). W odróżnieniu od WSDL w REST nie ma opisu formatu zapytania i odpowiedzi. Zamiast tego wykorzystywane są standardowe operacje HTTP jak GET, POST, PUT, DELETE pozwalające na realizację operacji CRUD (Create, Read, Update, Delete).

Założeniem REST jest brak stanowości zasobów, uproszczenie implementacji i zarządzania w porównaniu do WSDL.

Slajd
14

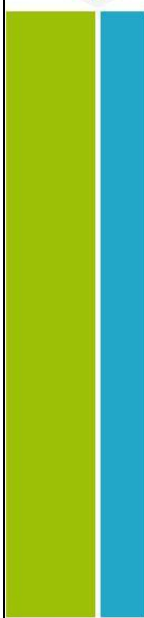


B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

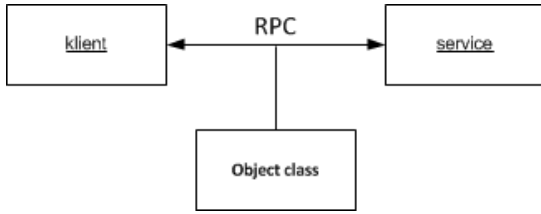
PHP
Hypertext Preprocessor



Implementacje SOA


DCOM,

- Serializowany komunikat
- Microsoft
- Protokół RPC



```

graph LR
    klient[klient] <-->|RPC| service[service]
    ObjectClass[Object class] --- RPC
            
```



Lekcja 2 – Architektura usług sieciowych

DCOM – (Distributed Component Object Model) - interfejs programistyczny zapewniający komunikację między aplikacjami na poziomie obiektów. Jest technologią firmy Microsoft przygotowaną jako konkurencję do CORBA. Technologia rozwinięta w .NET Framework stała się częścią COM+ powszechnie wykorzystywaną w Windows. Jest opatentowaną technologią firmy Microsoft.

Formatem komunikatu jest serializowany obiekt określonej klasy. Zatem w celu przygotowania integracji aplikacji ze sobą wymagana jest znajomość definicji obiektów jakie używane są po stronie usługi.

Protokół: RPC, później stosowano HTTP

Problemy z implementacją, komunikacją poprzez firewalle oraz ze zwalnianiem pamięci.

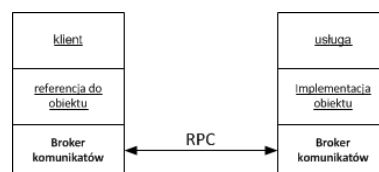
Człowiek - najlepsza inwestycja



Implementacje SOA

CORBA,









- Broker komunikatów
- Obiekt i mapowanie
- Protokół InterORB i jego odmiany



CORBA– (Common Object Request Broken Architecture) – standard zapewniający modułom oprogramowania napisanym w różnych językach, na różnych platformach do wzajemnej komunikacji. Standard został zdefiniowany przez OMG (Object Management Group).

CORBA wymaga zastosowania mechanizmu ORB (Object Request Broker) do wymiany komunikatów. Komunikaty muszą zostać prawidłowo zmapowane do wspólnego interfejsu (wprowadzony IDL – interface definition language) .

Protokół: InterORB, SSL InterORB, HTIOP – HyperText InterORB, Zipped IOP

Slajd 16	<div>    </div> <h2>Implementacje SOA</h2> <p>WCF,</p> <ul style="list-style-type: none"> • WSDL • Różne formy komunikacji • Różne formaty komunikatu <div>  Lekcja 2 – Architektura usług sieciowych </div>	<p>WCF – (Windows Communication Foundation) – nowoczesny zestaw narzędzi programistycznych na platformę Microsoft .NET pozwalających na budowę aplikacji SOA. WCF mocno rozszerza paletę funkcjonalności SOA o np. dynamiczną adresację (routing), zapewnienie dostarczenia komunikatu oraz kontrolę uprawnień.</p> <p>Usługa pozwala na wybór formy komunikacji np. SOAP HTTP, SOAP TCP, SOAP MQ.</p> <p>Format komunikatu to uproszczony komunikat SOAP lub JSON (JavaScript Object Notation).</p>
Slajd 17	<div>    </div> <h2>Ćwiczenia praktyczne - WSDL</h2> <p>Przeanalizuj i omów plik HelloService.wsdl</p> <div>  Lekcja 2 – Architektura usług sieciowych </div>	<p>Opis na stronie: http://www.tutorialspoint.com/wsdl/wsdl_example.htm</p>

Człowiek - najlepsza inwestycja

Slajd
18



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Ćwiczenia

1. Zaznacz implementacje SOA

- Peer2peer,
- SOAP,
- REST,
- Klient-serwer,
- PHP,
- WebServices,
- DCOM,
- COBRA,
- WCF,
- Windows



Lekcja 2 – Architektura usług sieciowych

1. Gwiazdką zaznaczono prawidłowe odpowiedzi
peer2peer,
SOAP*,
REST*,
klient-serwer,
PHP,
WebServices*,
DCOM*,
COBRA*,
WCF*,
Windows

Slajd
19

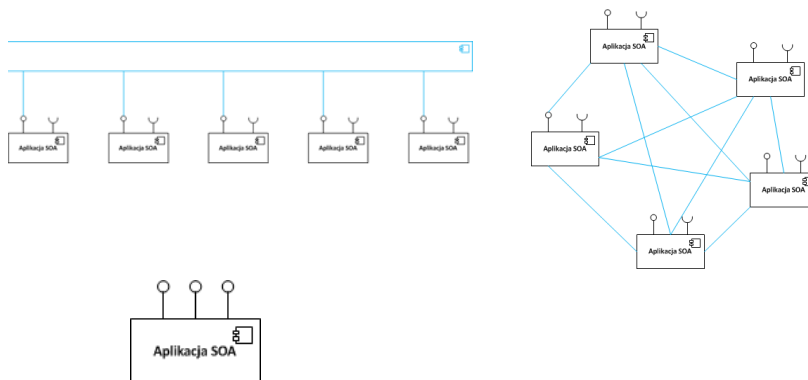


SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Ćwiczenia

2. Który diagram reprezentuje SOA BUS



Lekcja 2 – Architektura usług sieciowych

1. Odpowiedź prawidłowa to diagram w lewym górnym rogu.

Człowiek - najlepsza inwestycja





KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY




Slajd
20

PHP
Hypertext Preprocessor


Pytania

1. Co to jest usługa sieciowa?
2. Rozwiń i omów skrót SOA.
3. Wymień 4 implementacje SOA.
4. Co to jest WSDL?
5. Omów strukturę komunikatu SOAP.
6. Na jakie formy komunikacji pozwala WCF?
7. Do czego stosuje się SOA BUS?



Lekcja 2 – Architektura usług sieciowych


Slajd
21




PHP
Hypertext Preprocessor

Ćwiczenia praktyczne

Proszę przygotować skrypt, który wywoła webserwis spod adresu <http://footballpool.dataaccess.eu/data/info.wso?wsdl> udostępniający informację na temat piłki nożnej. Przeanalizuj plik WSDL oraz w prawidłowy sposób wyświetl 5, 10 lub 20 najlepszych piłkarzy.



Lekcja 2 – Architektura usług sieciowych

Rozwiązanie w pliku test.php
Pomocny może być analizator plików wsdl, np <https://www.wsdl-analyzer.com/>

Człowiek - najlepsza inwestycja

Slajd
22



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP

Hypertext Preprocessor

Ćwiczenia praktyczne

Proszę przygotować skrypt, który wywoła webserwis spod adresu <http://footballpool.dataaccess.eu/data/info.wso?wsdl> udostępniający informację na temat piłki nożnej. Przeanalizuj plik WSDL oraz w prawidłowy sposób wyświetl wszystkich piłkarzy, którzy otrzymali żółte lub czerwone kartki podczas gry na boisku.



Lekcja 2 – Architektura usług sieciowych

Rozwiązanie w pliku test2.php
Pomocny może być analizator plików wsdl, np <https://www.wsdl-analyzer.com/>

Slajd
23



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP

Hypertext Preprocessor

Ćwiczenia praktyczne

Proszę przygotować skrypt, który wywoła webserwis spod adresu <http://footballpool.dataaccess.eu/data/info.wso?wsdl> udostępniający informację na temat piłki nożnej. Przeanalizuj plik WSDL oraz stwórz formularz, który pobierał będzie informacje o ilości najlepszych piłkarzy do wyświetlenia. Wprowadź walidację, polegającą na ograniczeniu wprowadzanej wartości z zakresu od 1 do 20.



Lekcja 2 – Architektura usług sieciowych




Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



	Rozwiązanie w pliku index.php Pomocny może być analizator plików wsdl, np https://www.wsdl-analyzer.com/
Slajd 24	<div>   <div> PHP Hypertext Preprocessor </div> </div> <h2>Ćwiczenia praktyczne</h2> <p>Proszę przygotować skrypt, który wywoła webserwis spod adresu http://footballpool.dataaccess.eu/data/info.wso?wsdl udostępniający informację na temat piłki nożnej. Przeanalizuj plik WSDL oraz wybierz samodzielnie jakie informacje z podanego webserwisu chcesz pobrać. W odpowiedni sposób wywołaj wybraną metodę zwracając uwagę na listę parametrów, które ta metoda przyjmuje.</p> <div>  Lekcja 2 – Architektura usług sieciowych </div>
	Pomocny może być analizator plików wsdl, np https://www.wsdl-analyzer.com/

Człowiek - najlepsza inwestycja

Slajd
25



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Ćwiczenia praktyczne

Przygotuj REST API bazując na frameworku SLIM (<http://www.slimframework.com/>). Stwórz dowolną tabelę w bazie danych, pobierz dane, oraz udostępnij przy pomocy stworzonego przez Ciebie API. (Np. Pobierz listę wszystkich adresów e-mail oraz udostępnij je w lokalizacji <http://localhost/api/email/>)



Lekcja 2 – Architektura usług sieciowych

Slajd
26



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Ćwiczenia praktyczne

Przetestuj przygotowane we wcześniejszym ćwiczeniu API przy pomocy narzędzia „Advanced REST client” dostępnego w przeglądarce Chrome.



Advanced REST client
Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego



Lekcja 2 – Architektura usług sieciowych

Advanced REST client trzeba wcześniej w Chrome zainstalować. Dodatek ten nie jest domyślnie dostępny.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



10.2.3 Ćwiczenia

Ćwiczenie zostało przedstawione na slajdzie 6, 17, 18, 19. Pytania na slajdzie 20.

10.2.4 Opis założonych osiągnięć ucznia

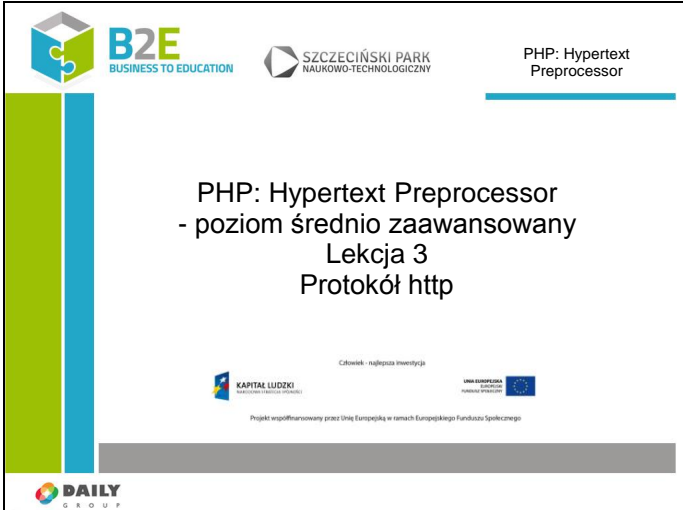
Uczeń powinien znać i potrafić omówić (narysować) podstawowe architektury aplikacji sieciowych oraz posiadać wiedzę na temat podanych implementacji architektury SOA.

10.3 Lekcja 3 - Protokół http (budowa protokołu HTTP 1.1, nagłówki żądania i odpowiedzi, komunikacja http)







10.3.1 Cel lekcji

Celem lekcji jest zapoznanie ucznia z protokołem HTTP, a w szczególności jego wersji 1.1. Pokazanie schematu tworzenia żądań i rozumienia odpowiedzi od serwera. Poznanie metod protokołu HTTP oraz sposobu komunikacji pomiędzy klientem i serwerem. Zastosowanie podstawowych metod \$GET oraz \$POST w języku PHP.

10.3.2 Treść - slajdy z opisem


Slajd 1		Słowa wstępu.
------------	--	---------------


Człowiek - najlepsza inwestycja

<p>Slajd 2</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Protokół HTTP</h2> <ul style="list-style-type: none"> • Protokół HTTP (ang. Hypertext Transfer Protocol) – to najpopularniejszy protokół wykorzystywany w sieciach komputerowych. Mówiąc prostym językiem jest to ścisły schemat postępowania (zbiór zasad) dla urządzeń komunikacyjnych w jaki sposób przysyłać dokumenty hipertekstowe. • Do czego służy?  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Protokół HTTP (ang. Hypertext Transfer Protocol) – to najpopularniejszy protokół wykorzystywany w sieciach komputerowych. Mówiąc prostym językiem jest to ścisły schemat postępowania (zbiór zasad) dla urządzeń komunikacyjnych w jaki sposób przysyłać dokumenty hipertekstowe. Rozpowszechniony dzięki sieci WWW (ang. World Wide Web), za jego pomocą odbywa się przysyłanie danych oraz komunikacja w większości aplikacji internetowych.</p>
<p>Slajd 3</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Protokół HTTP 1.1</h2> <ul style="list-style-type: none"> • Formalna wersja od 2001 r. • Aktualna wersja z 2012 r. • Proxy – serwer pośredniczący • Stałe połączenia • Serwery wirtualne • Autoryzacja • Cache  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Protokół HTTP w wersji 1.1 powstał w 2001 r. Obecny standard pochodzi z grudnia 2012 r. Obecna wersja wprowadziła szereg zmian w stosunku do poprzedniej. Możemy wykorzystać protokół jako protokół pośredniczący np. pomiędzy klientem, a innym protokołem np. SMTP albo FTP. Dzięki protokołowi TCP/IP najnowszy standard wspiera stałe połączenia z serwerem. Możliwa jest również obsługa serwerów wirtualnych. Stworzono mechanizm autoryzacji poprzez HTTP oraz mechanizm Cache'owania czyli przechowywania poprzednich wyników.</p>

Człowiek - najlepsza inwestycja

Slajd
4

 **B2E**
BUSINESS TO EDUCATION

 SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Sposób działania

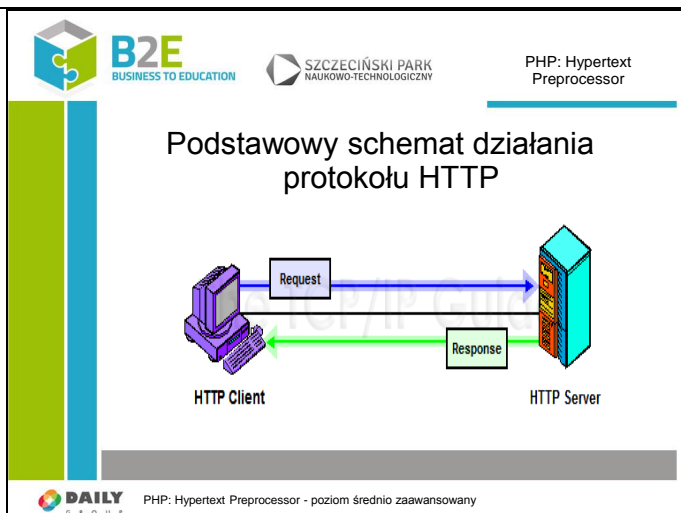
- Wykorzystywany rodzaj architektury
- Bezstanowość protokołu
- Mechanizmy sesji i ciasteczek

 **DAILY**
GROUP


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Protokół HTTP wykorzystuje architekturę klient – serwer, klient (w tym przypadku przeglądarka) wysyła żądanie do serwera, który wysyła odpowiedź w zależności od otrzymanej wcześniej treści. Standaryzują on formę zapytań oraz odpowiedzi. W związku z tym, że protokół nie zapamiętuje informacji o wcześniejszych transakcjach (transakcja = żądanie + odpowiedź), określa się go mianem protokołu bezstanowego. Brak możliwości przechowywania informacji jest swoistą przeszkodą, jeśli chodzi o komunikacje w sieci (np. autentykacja), dlatego istnieją inne mechanizmy przechowujące informacje('cookies' – po stronie klienta, 'session' po stronie serwera).


Slajd
5



Człowiek - najlepsza inwestycja



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Opis podstawowego schematu działania protokołu HTTP

1. Nawiązanie połączenia
2. Wysłanie żądania HTTP
3. Otrzymanie odpowiedzi



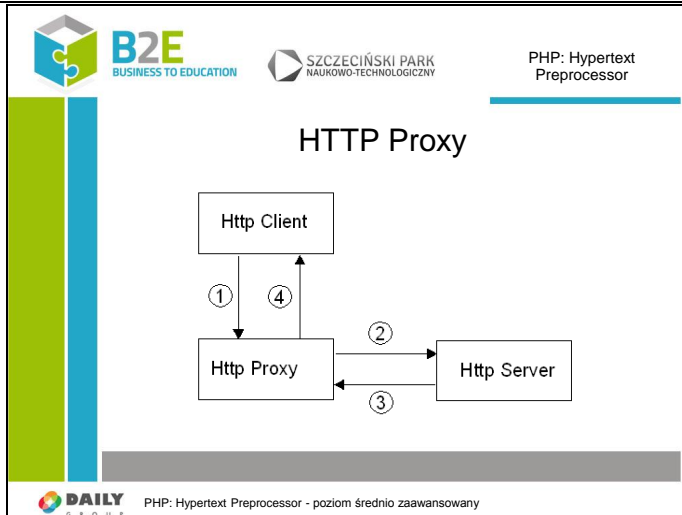
DAILY
GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

1. Nawiązanie połączenia
– w pierwszej kolejności klient używając adresu serwera (URL lub IP), nawiązuje połączenie najczęściej za pomocą protokołu TCP/IP. Domyślny port 80.
2. Następnie klient wysyła żądanie HTTP do serwera, które jest ciągiem znaków ASCII zakończona znakami „CR LF”
3. Ostatnim krokiem komunikacji jest otrzymanie odpowiedzi od serwera w postaci ciągu znaków w określonym standardzie np. HTML

Istnieje również bezpieczna wersja protokołu (HTTPS), która działa na porcie 443, a jej transmisja jest szyfrowana.

Slajd
7



Slajd
8

PHP: Hypertext Preprocessor

Opis działania HTTP Proxy

1. Nawiązanie połączenia z serwerem Proxy i wysłanie żądania HTTP
2. Translacja adresu i przekazanie żądania
3. Przekazanie odpowiedzi do serwera proxy
4. Otrzymanie odpowiedzi

DAILY GROUP PHP: Hypertext Preprocessor - poziom średnio zaawansowany


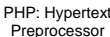



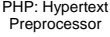


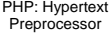

1. Nawiązanie połączenia – w pierwszej kolejności klient używając adresu serwera(URL lub IP), nawiązuje połączenie najczęściej za pomocą protokołu TCP/IP. Domyślny port 80. Następnie wysyła żądanie HTTP do serwera, które jest ciągiem znaków ASCII zakończona znakami „CR LF”

2. Translacja adresu zewnętrznego na adres wewnętrzny przez serwer Proxy. Przekazanie żądania HTTP z serwera Proxy do wewnętrznego serwera HTTP.

3. Wysłanie odpowiedzi przez serwer wewnętrzny do serwera proxy.



4. Ostatnim krokiem komunikacji jest otrzymanie odpowiedzi od serwera proxy w postaci

Człowiek - najlepsza inwestycja

		ciągu znaków w określonym standardzie np. HTML
Slajd 9	   <p>Przykładowe żądanie HTTP (metoda POST)</p> <pre>POST /enlighten/calais.asmx/Enlighten HTTP/1.1 Host: api.opencalais.com Content-Type: application/x-www-form-urlencoded Content-Length: length licenseID=string&content=string&paramsXML=string</pre>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	
Slajd 10	   <p>Przykładowa odpowiedź HTTP (XML)</p> <pre>HTTP/1.1 200 OK Content-Type: text/xml; charset=utf-8 Content-Length: length <?xml version="1.0" encoding="utf-8"?> <string xmlns="http://clearforest.com/">string</string></pre>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	
Slajd 11	   <p>Ćwiczenie</p> <p>Wykonać skrypty get.php oraz post.php. Przeanalizować zwracaną zawartość.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	

Człowiek - najlepsza inwestycja


Slajd
12

PHP: Hypertext Preprocessor

Metody HTTP


- GET
- HEAD
- POST
- DELETE
- TRACE
- PUT
- OPTIONS



PHP: Hypertext Preprocessor - poziom średnio zaawansowany


Metody służą do komunikacji z serwerem poprzez protokół HTTP. Najpopularniejsze metody wykorzystywane w skryptach PHP to POST oraz GET. Są one dostępne w podstawowych składnikach aplikacji jak tablice superglobalne, formularze oraz adres URI (ang. Uniform Resource Identifier). HTTP posiada jednak więcej metod, którymi możemy się w odpowiedni sposób posłużyć.

Slajd
13

PHP: Hypertext Preprocessor

Metoda GET



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Metoda GET służy do pobierania zasobu wskazanego przez URI, może mieć postać warunkową jeśli w nagłówku występują pola warunkowe takie jak "If-Modified-Since,, Na zdjęciu przedstawiono przykładowy adres z dwoma parametrami GET. Zmienna locale przyjmuje wartość en, natomiast zmienna app wartość forestkit. Metoda GET posiada ograniczenia dotyczące ilości użytych w adresie znaków. Ograniczenie to zależy od konfiguracji serwera oraz użytej przeglądarki. W przypadku wystąpienia zbyt długiego adresu, serwer powinien zwrócić błąd 414 (Request-URI Too Long).

Człowiek - najlepsza inwestycja

Slajd
14



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Metoda POST

```
<form method="post" action="index.php">  
<input type="text" name="value" value="" />  
<input type="submit" value="Submit" />  
</form>
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Metoda ta służy do przyjmowania danych przesłanych przez klienta np. podczas przetwarzania formularzy.

Metodę tę używamy również do przesyłania plików (przykład w materiałach dołączonych do lekcji, plik upload.html)

Slajd
15



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Metoda PUT



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Metoda PUT służy do przesyłania danych w postaci pliku od klienta do serwera


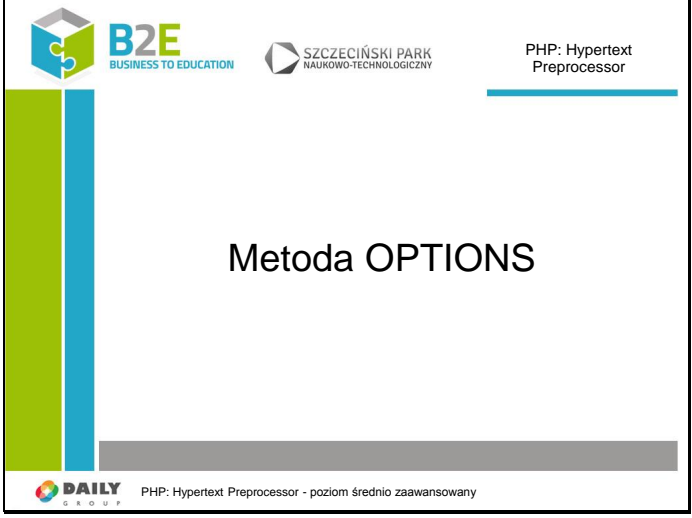

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



<p>Slajd 16</p>		<p>Metoda DELETE wykorzystywana jest jako żądanie usunięcia zasobu z serwera, ze względów bezpieczeństwa poprzedzona jest autoryzacją</p>
<p>Slajd 17</p>		<p>Metoda OPTIONS służy do pobierania informacji o opcjach na danym kanale komunikacyjnym dla zarejestrowanego użytkownika</p>
<p>Slajd 18</p>		<p>Metoda HEAD służy do pobierania informacji o zasobie, sprawdzania jego dostępności</p>

Człowiek - najlepsza inwestycja

Slajd
19



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Metoda TRACE




DAILY
GROUP


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Metoda TRACE służy do diagnostyki, analizy kanału komunikacyjnego.

Slajd
20



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład zastosowania metody GET

Metoda ta przekazuje dane poprzez przekazanie parametrów z adresem URI:


np. <http://adres.pl/index.php?action=default>

Przykład sklejenia adresu z poziomu kodu:

```
$action = 'default';  
$url .= '?action='.$action;
```

Pobieranie wartości:

```
$action = (isset($_GET['action'])) ? $_GET['action'] : null;
```



DAILY
GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Podstawowy mechanizm w języku PHP, zastosowania żądań typu GET oraz jednocześnie najłatwiejsza metoda skomunikowania ze sobą skryptów na serwerze HTTP. Wykorzystujemy ją dodając parametry do adresu skryptu.

Pobieranie wartości przy pomocy kodu:

```
$action = (isset($_GET['action'])) ? $_GET['action'] : null;
```

jest równoznaczne z:

```
if(isset($_GET['action'])) {  
    $action = $_GET['action'];  
}
```

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



```
else {
    $action = null;
}
```

I polega na sprawdzeniu czy zmienna action istnieje w tablicy \$_GET (czy istnieje w adresie). Jeśli tak – pobieramy jej wartość, jeśli nie – ustawiamy na null.

Slajd
21

PHP: Hypertext
Preprocessor

Przykład zastosowania metody POST


Metoda najczęściej wykorzystywana przy przesyłaniu formularzy:

Setter:

```
<form action="adres_skryptu" method="post">
<input name="field_name" type="text" />
</form>
```

Getter:

```
$form_data = $_POST['field_name']
```


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Drugi podstawowy mechanizm komunikacji dostępny w języku PHP, najczęściej stosowany do przetwarzania formularzy lub asynchronicznych zapytań HTTP, czyli technologii AJAX. W przeciwieństwie do metody GET, której limit długości adresu ustalany jest przez przeglądarkę klienta. Limit długości tablicy POST ustalany jest przez konfigurację serwera. Metoda POST służy również do przesyłania plików w języku PHP.

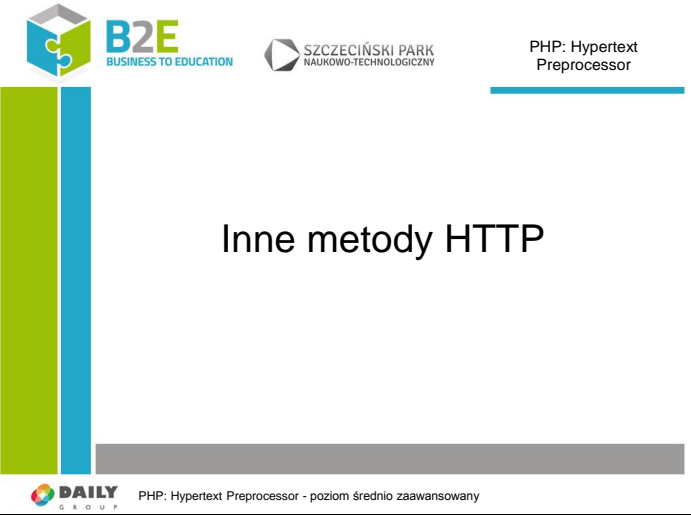
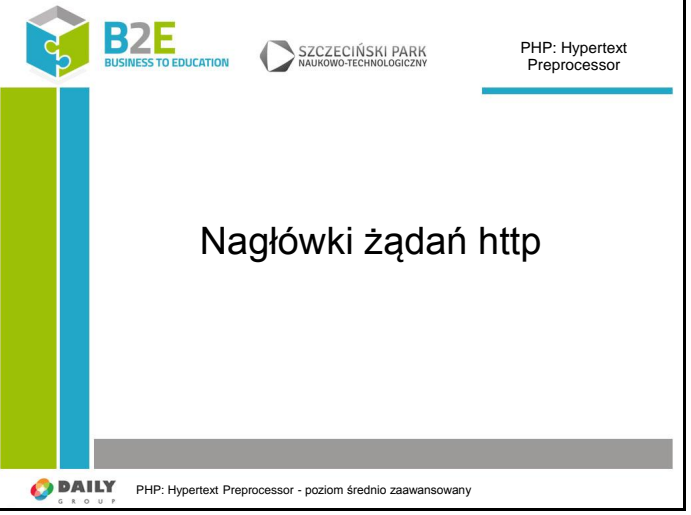
Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
 NARODOWA STRATEGIA SPÓJNOŚCI



UNIA EUROPEJSKA
 EUROPEJSKI
 FUNDUSZ SPOŁECZNY



<p>Slajd 22</p>	 <p>Inne metody HTTP</p>	<p>Poza standardowymi metodami jak GET i POST dostępnymi w samym języku PHP, możemy tworzyć własne zapytania HTTP korzystając przykładowo z rozszerzenia HTTP języka PHP lub zdefiniowanych przez nas obiektów, bądź funkcji. np. biblioteka cURL wspiera przesyłanie plików metodą PUT. Istnieją również architektury, które ułatwiają komunikację z serwerem, udostępniając odpowiednie API, jak architektura REST, SOAP czy Web Service.</p> <p>Biblioteka cURL w domyślnej konfiguracji serwera jest wyłączona, aby ją uruchomić należy w pliku php.ini odkomentować linijkę: extension=php_curl.dll Po wprowadzeniu zmiany należy zrestartować serwer apache.</p>
<p>Slajd 23</p>	 <p>Nagłówki żądań http</p>	<p>Żądanie HTTP realizuje się poprzez przesłanie odpowiednich nagłówków oraz parametrów do serwera, zakończonych znakami powrotu karetki oraz nowej linii. HTTP 1.1 pozwala na wysyłanie potokiem kilku żądań naraz. Nagłówki służą jako parametry żądania np. w jakim formacie wysyłamy żądanie i w jakim formacie odpowiedzi się spodziewamy.</p>


Człowiek - najlepsza inwestycja

Slajd
24

 **B2E**
BUSINESS TO EDUCATION
  SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY
 PHP: Hypertext
Preprocessor

Budowa żądania HTTP


1. Nazwa metody oraz standard np.
GET / HTTP/1.1
2. Nagłówki wymagane np.
Host: jakishost.pl
3. Nagłówki opcjonalne np.
Accept: text/xml
4. Dane np.
5. *CR + LF*

 **DAILY**
G R O U P
 PHP: Hypertext Preprocessor - poziom średnio zaawansowany


- W pierwszej kolejności wysyłamy nazwę metody, która w naszym przykładzie jest to metoda GET oraz nazwa standardu z jakim nasza prośba ma być przetwarzana.
- Następnie wysyłamy nagłówki wymagane przez standard w tym przypadku HOST, który jest wymagany ze względu na możliwość stworzenia Wirtualnych Hostów w obrębie jednego adresu IP.
- W dalszej części znajdują się nagłówki opcjonalne w przykładzie rodzaj akceptowalnych przez klienta danych.
- Jeżeli metoda przesyła dane np. POST umieszczamy je w tym właśnie miejscu.
- Kończymy żądanie znakiem powrotu karetki oraz znakiem przejścia do nowej linii.

Człowiek - najlepsza inwestycja

Slajd
25



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Budowa odpowiedzi HTTP

1. Kod odpowiedzi HTTP
HTTP/1.1 200 OK
2. Nagłówki wymagane np.
Date: Thu, 20 Dec 2001 12:04:30 GMT
3. *CR + LF*
4. Zwrócone dane




DAILY GROUP


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

- Pierwszym elementem jest kod odpowiedzi w tym przypadku wszystko jest prawidłowo, żądanie zostało zaakceptowane i zwrócone zostały dane.
- Nagłówki odpowiedzi w przykładzie czas serwera.
- Zakończenie nagłówków odpowiedzi.
- W tym miejscu przekazane są dane zwrócone z serwera HTTP.

Slajd
26



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne

Przy pomocy cURLa pobierz dowolną stronę internetową, następnie zmodyfikuj nagłówki odpowiedzi i wyświetl całość jako tekst (text/plain).



DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

curl.php

Człowiek - najlepsza inwestycja







KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
27





PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne



Stwórz skrypt, który wywoła z dowolnymi zmiennymi i wartościami w polu adresu, a następnie wyświetli całą tablicę \$_GET.
Przykładowy adres do wywołania:



- `http://localhost/Lekcja3/get.php?imie=Adam&nazwisko=slodowy`


 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

get.php

Slajd
28





PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne


Stwórz skrypt, który zawiera formularz HTML z polami imienia oraz nazwiskiem. Prześlij formularz przy pomocy metody POST a następnie wyświetl zawartość odpowiednich elementów tablicy \$_POST.


 PHP: Hypertext Preprocessor - poziom średnio zaawansowany


post.php

Człowiek - najlepsza inwestycja

Slajd
29



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne

Stwórz skrypt, który będzie wysyłał pliki na serwer i umieszczał je w folderze „upload”.




DAILY GROUP


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

upload.php

Slajd
30



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Ćwiczenia praktyczne

Do stworzonego wcześniej skryptu wysyłającego pliki na serwer, dodaj ograniczenie polegające na możliwości wysyłania:

- Tylko plików graficznych
- Plików nie większych niż 20000 bajtów
- Sprawdzanie plików dokonaj po rozszerzeniu (*.jpg, *.gif itd...) oraz mime-type pliku (image/gif, image/jpeg ...)



DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

upload.php

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
31


B2E
 BUSINESS TO EDUCATION
 
 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Dziękujemy za uwagę

Człowiek - najlepsza inwestycja


 KAPITAŁ LUDZKI
INICJATYWA EUROPEJSKA

 UNIA EUROPEJSKA
FUNDUSZ SPOŁECZNY

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego


 DAILY
 GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

10.3.3 Ćwiczenia

Wykonać ćwiczenia ze slajdu 11. Dodatkowo można zlecić uczniom napisanie przykładów użycia innych metod HTTP znajdujących się na wybranych slajdach.

10.3.4 Opis założonych osiągnięć ucznia

Uczeń powinien poznać i umieć opowiedzieć o metodach protokołu HTTP, potrafił wyjaśnić na czym polega komunikacja pomiędzy klientem, a serwerem. Powinien umieć opowiedzieć z czego składa się żądanie oraz odpowiedź HTTP.

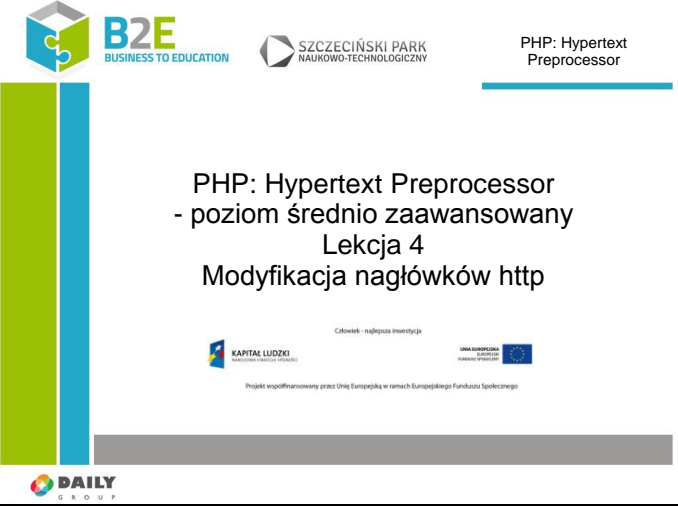
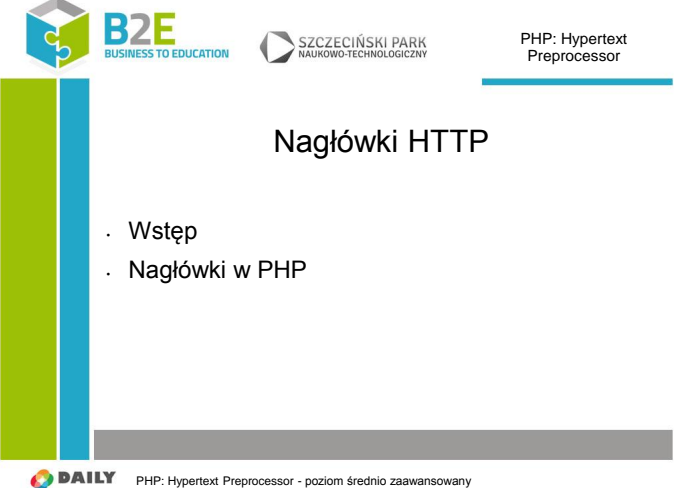
10.4 Lekcja 4 - Modyfikacja nagłówków http (Lista nagłówków, Wykorzystanie funkcji header, Zastosowania w skryptach PHP)

10.4.1 Cel lekcji

Celem lekcji jest nauczenie uczestników nagłówków http. Powinni umieć używać ich w języku php przy użyciu funkcji header.


Człowiek - najlepsza inwestycja

10.4.2 Treść - slajdy z opisem


<p>Slajd 1</p>	 <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany Lekcja 4 Modyfikacja nagłówków http</p> <p><small>Człowiek - najlepsza inwestycja</small></p> <p><small>KAPITAŁ LUDZKI NARODOWA STRATEGIA SPÓJNOŚCI</small></p> <p><small>UNIA EUROPEJSKA FUNDUSZ SPOŁECZNY</small></p> <p><small>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</small></p> <p><small>DAILY GROUP</small></p>	<p>Słowa wstępu.</p>
<p>Slajd 2</p>	 <p>Nagłówki HTTP</p> <ul style="list-style-type: none"> · Wstęp · Nagłówki w PHP <p><small>DAILY GROUP</small></p> <p><small>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</small></p>	<p>Nagłówki HTTP w języku PHP, wykorzystywane są do komunikacji pomiędzy serwerem, a stroną WWW. Do ich wysyłania najczęściej stosujemy funkcję 'header'. Używać ich może w różnych celach np. do autoryzacji HTTP, odświeżenia strony, bądź przekierowania przez serwer na inną stronę.</p>

Człowiek - najlepsza inwestycja

Slajd
3



B2E
BUSINESS TO EDUCATION

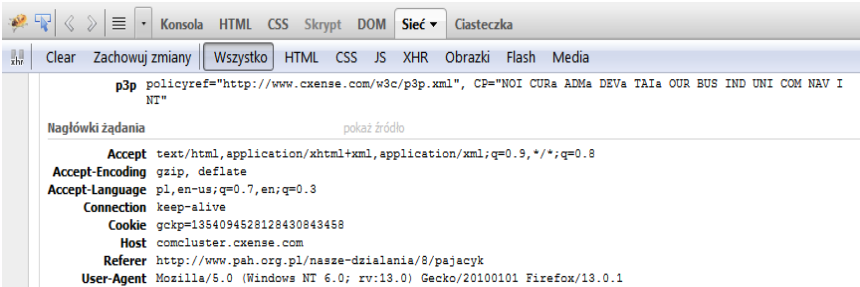



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Ćwiczenie

Otwórz dowolną stronę www i przeanalizuj nagłówki HTTP przy pomocy narzędzia firebug.





PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Slajd
4



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Dostępne nagłówki



- Standardowe
- Niestandardowe



PHP: Hypertext Preprocessor - poziom średnio zaawansowany


Nagłówki HTTP w języku PHP, dzielimy na standardowe i niestandardowe. Standardowe to te, które znajdują się w specyfikacji HTTP w wersji 1.1, a niestandardowe to takie nagłówki, które są dostępne, lecz nie znajdują się w specyfikacji.

Slajd
5



PHP: Hypertext
Preprocessor



Lista nagłówków



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Nagłówki HTTP w języku PHP, wykorzystywane są do komunikacji pomiędzy serwerem, a stroną WWW. Do ich wysyłania najczęściej stosujemy funkcję 'header'. Używać ich może w różnych celach np. do autoryzacji HTTP, odświeżenia strony, bądź przekierowania przez serwer na inną stronę.


Slajd
6



PHP: Hypertext
Preprocessor

Nagłówek Accept

Accept: application/xhtml+xml, application/xml;q=0.9, text/xml;q=0.7, text/html;q=0.5, text/plain;q=0.3

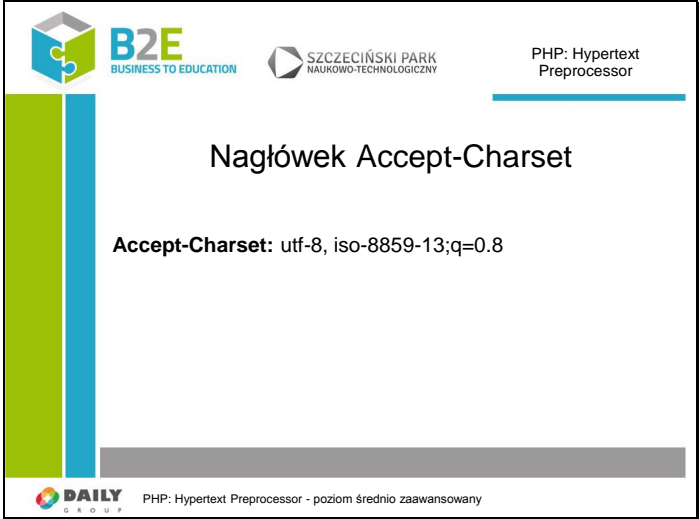
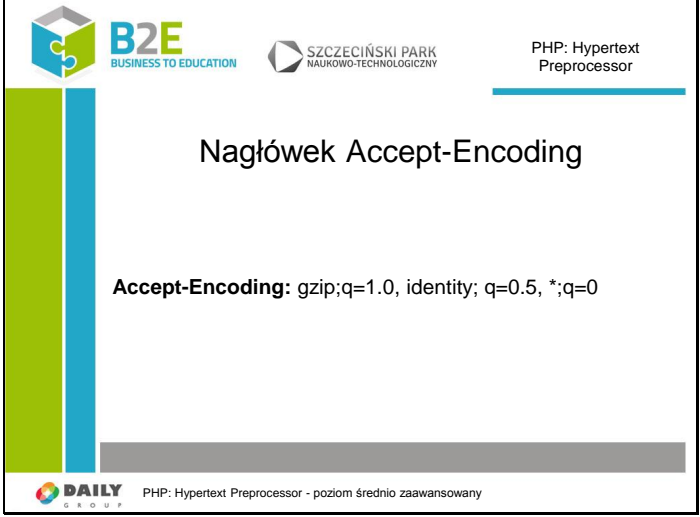


PHP: Hypertext Preprocessor - poziom średnio zaawansowany










Służy do określenia listy akceptowalnych przez przeglądarkę typów MIME dokumentu, oraz opcjonalnie hierarchii każdego typu. Liczby podane po ;q= powinny mieć wartości od 0 do 1, co 0.1. Jako separator miejsc dziesiętnych użyta musi być kropka. W przypadku braku zdefiniowanej wartości ;q= przyjmowana jest wartość: ;q=1 (równoznaczna z ;q=1.0).

Powyższy przykład ustala najwyższy priorytet typowi application/xhtml+xml. Jeżeli serwer nie posiada dokumentu w formacie application/xhtml+xml, to prześle plik w formacie application/xml – ogólnym dla podjęzyków XML, takich jak XHTML, SVG. Jeżeli na serwerze nie ma również dokumentu w tym formacie, dane zostaną przesłane jako text/xml – czyli ogólne dane w formacie XML, w ostateczności, jeżeli żadnego z tych typów nie posiada, serwer wyśle dane w formacie text/plain, czyli

Człowiek - najlepsza inwestycja


	<p>zwykłego tekstu. Typy MIME mogą być określone za pomocą symboli wieloznacznych np.: text/*, */*.</p>	
Slajd 7		<p>Określa preferowane przez przeglądarkę formaty kodowania. Wartości ;q= interpretowane są na takich samych zasadach, jak w przypadku typów MIME z nagłówka Accept.</p>
Slajd 8		<p>Ten nagłówek określa kodowanie, za pomocą którego zostanie przesłana zawartość. Stosowane zwykle do kompresji treści.</p>

Człowiek - najlepsza inwestycja


<p>Slajd 9</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Nagłówek Accept-Language</h2> <p>Accept-Language: pl, en-us;q=0.7</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Określa w jakim języku użytkownik przeglądarki życzy sobie czytać strony.</p>
<p>Slajd 10</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Nagłówek Accept-Ranges</h2> <p>Accept-Ranges: bytes <i>lub</i> none</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Ten nagłówek jest ustawiany przez przeglądarki i programy ułatwiające ściąganie plików. Określa czy klient potrafi (wartość bytes) lub nie (wartość none) odczytywać pliki przesyłane w częściach.</p>
<p>Slajd 11</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Nagłówek Allow</h2> <p>Allow: GET, POST, HEAD</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Lista metod na które serwer ma zezwolić.</p>

Człowiek - najlepsza inwestycja

Slajd
12



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Nagłówek Authorization


Authorization: Basic QWxhZGRpbjpvGVulHNlc2FtZQ==



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Przesyła nagłówek do autoryzacji, najczęściej zakodowane login i hasło.

Slajd
13



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Nagłówek Cache-Control

Cache-Control: no-cache/no-store/max-age=3600/min-freash=3600

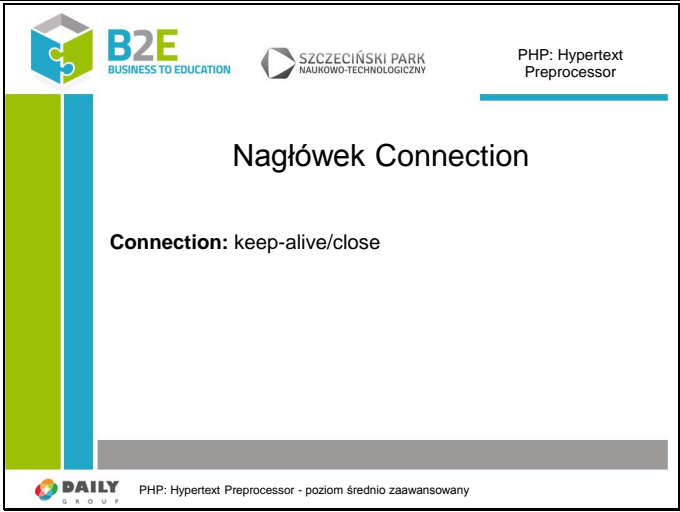
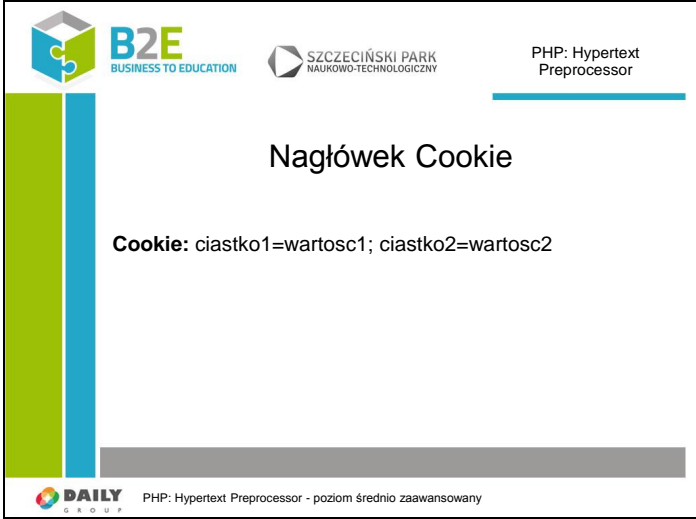


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

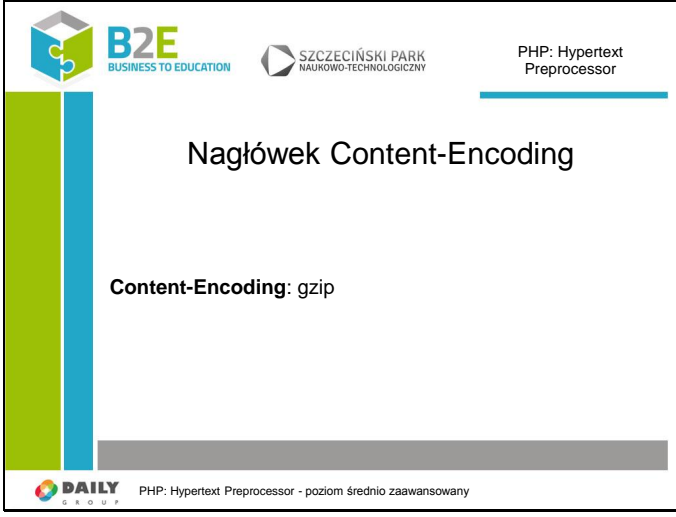
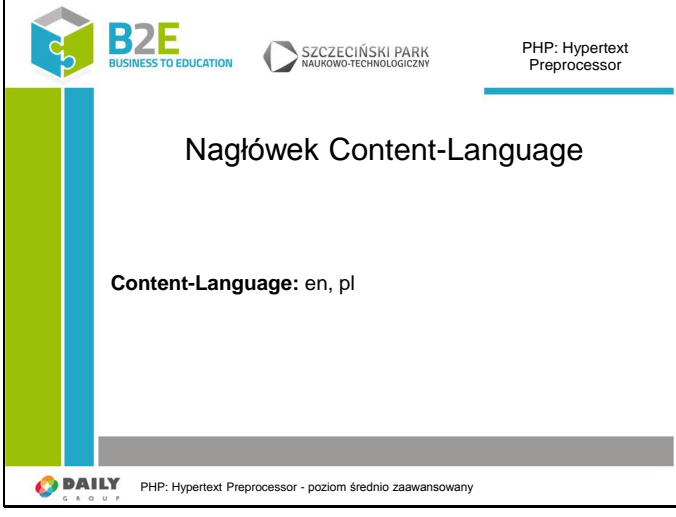

Określa regułę według której przeglądarka może (lub nie może) przechowywać stronę w pamięci podręcznej. Dostępne wartości to:

- no-store – strona nie może być przechowywana.
- no-cache – strona może być przechowywana pod warunkiem każdorazowego sprawdzania jej aktualności. Te dwa nagłówki mają szczególne znaczenie dla serwerów proxy. Pierwszy nakazuje przesyłać dokument bezwzględnie, natomiast drugi pozwala na skorzystanie z kopii w pamięci podręcznej, jeżeli dokument źródłowy się nie zmienił.
- max-age=liczba_sekund – oznacza, że można skorzystać z cache pod warunkiem iż nie jest przechowywane dłużej jak liczba_sekund. Jeżeli jest starsze przeglądarka powinna ściągnąć nowszą wersję.


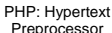



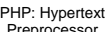
Człowiek - najlepsza inwestycja

		<ul style="list-style-type: none"> •min-fresh=liczba_sekund – nakazuje przeglądarce ściągnąć dokument jeżeli jego wiek nie przekracza liczby_sekund + aktualna_data
Slajd 14		Nagłówek określa, czy zamknąć połączenie po zapytaniu, czy je utrzymywać.
Slajd 15		W tym nagłówku przesyłane są wszystkie ciasteczka ustawione w przeglądarce.

Człowiek - najlepsza inwestycja

<p>Slajd 16</p>		<p>Określa typ kodowania pliku.</p>
<p>Slajd 17</p>		<p>Język dokumentu przesłanego przez serwer. Może być określony więcej niż jeden język. Wtedy język główny strony jest określony jako ostatni na tej liście.</p>
<p>Slajd 18</p>		<p>Określa rozmiar zawartości w bajtach</p>

Człowiek - najlepsza inwestycja

Slajd 19	   <h2 data-bbox="485 327 839 360">Nagłówek Content-Range</h2> <p data-bbox="387 418 732 441">Content-Range: bytes 123-456/5000</p>  <p data-bbox="416 672 751 687">PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	Określa zakres przesłanej części pliku w tym przypadku bajty od 123 do 456 z 5000(cały plik).
Slajd 20	   <h2 data-bbox="496 840 826 873">Nagłówek Content-Type</h2> <p data-bbox="387 931 860 954">Content-Type: application/xhtml+xml; charset=utf-8</p>  <p data-bbox="416 1184 751 1200">PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	Tym nagłówkiem serwer informuje przeglądarkę, w jakim formacie i stronie kodowej wysyłany jest dokument.
Slajd 21	   <h2 data-bbox="555 1373 762 1406">Nagłówek Date</h2> <p data-bbox="387 1514 743 1536">Date: Tue, 15 Nov 1994 08:12:31 GMT</p>  <p data-bbox="416 1722 751 1738">PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	Data serwera.

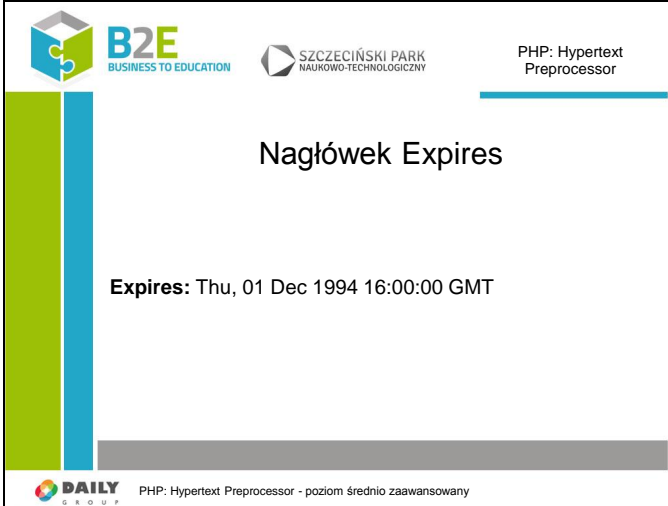

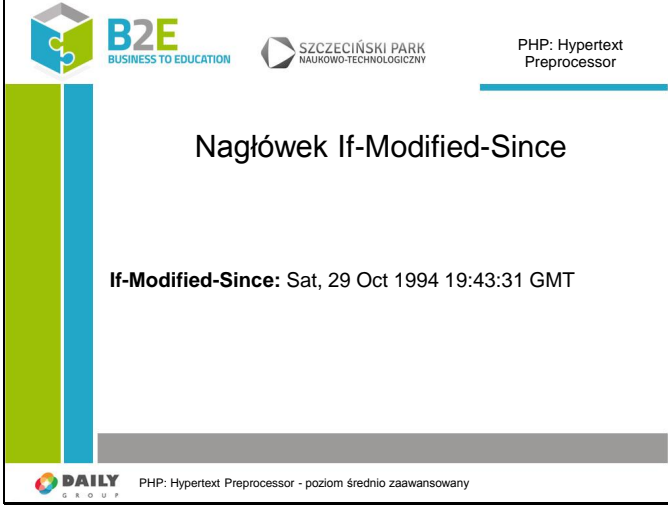
Człowiek - najlepsza inwestycja




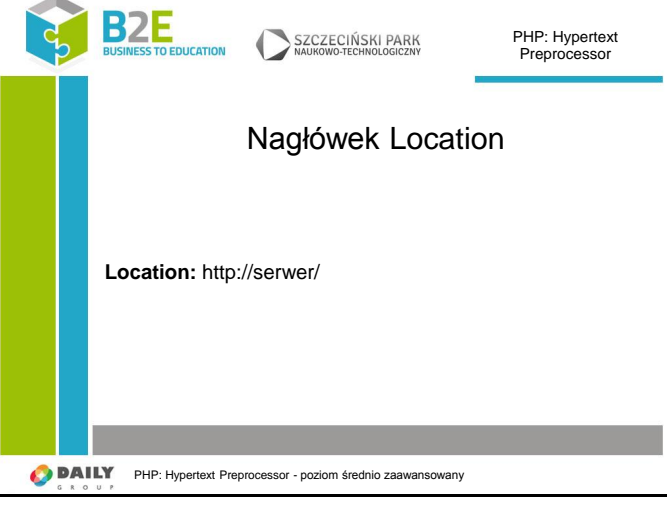
KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY






<p>Slajd 22</p>		<p>Data powyżej której dokument będzie nieaktualny.</p>
<p>Slajd 23</p>		<p>Nagłówek określający z/dla jakiej domeny określony został URI (do serwera bowiem nie dociera nazwa domenowa, ale jedynie adres IP). Nagłówek wymagany w HTTP 1.1. Dzięki niemu można uruchomić kilka serwerów wirtualnych na jednym adresie IP.</p>
<p>Slajd 24</p>		<p>Nagłówek nakazuje serwerowi przesłać dokument tylko jeżeli został zmodyfikowany od danej daty. Jeżeli dokument się nie zmienił serwer powinien wysłać przeglądarce kod odpowiedzi 304.</p>








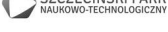

Człowiek - najlepsza inwestycja

<p>Slajd 25</p>		<p>Nagłówek serwera informujący o ostatniej aktualizacji dokumentu.</p>
<p>Slajd 26</p>		<p>Wymusza przekierowanie na określony adres. Dla kodu odpowiedzi 201 adres oznacza miejsce nowo utworzonego zasobu. Kod 301 - oznacza przekierowanie stałe. Od tej pory przeglądarka powinna korzystać z nowego adresu. Kod 304 - oznacza, że strona nie została zmodyfikowana od ostatniego zgłoszonego pobrania. Kod 307 – oznacza czasowe przekierowani</p> <p>Specyfikacja odpowiedzi HTTP: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html</p>

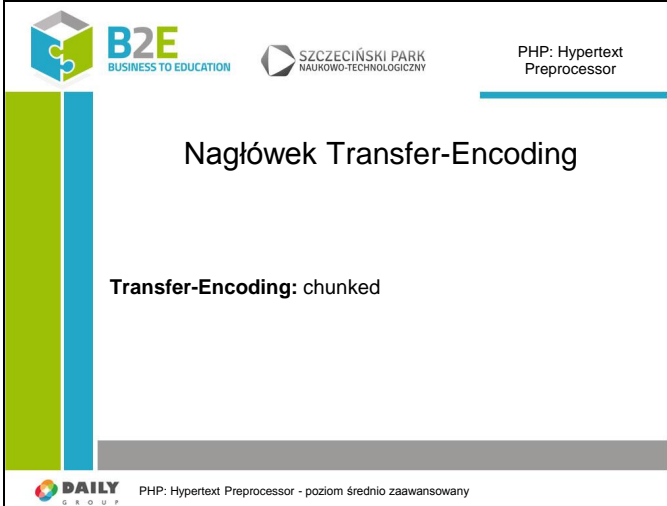
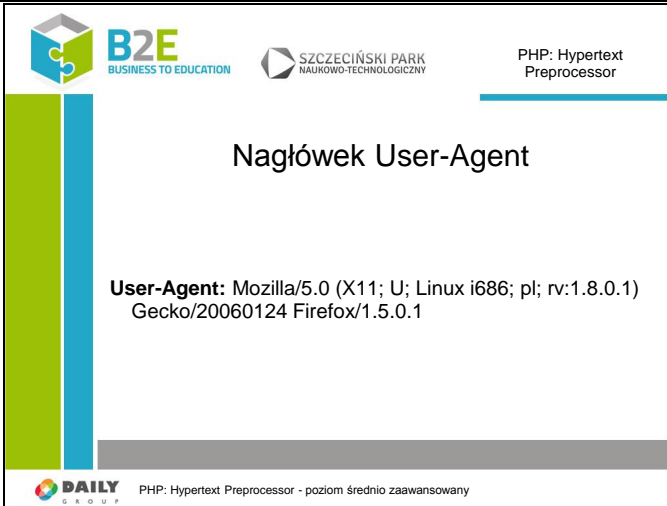
Człowiek - najlepsza inwestycja

<p>Slajd 27</p>		<p>Określa jaką część pliku oczekuje przeglądarka. Podawany w formacie bytes=pozycja_początkowa-pozycja_końcowa</p>
<p>Slajd 28</p>		<p>Określa adres strony internetowej, z której użytkownik został przekierowany za pomocą odnośnika. Przykład: film z YouTube został umieszczony na FaceBook'u, po kliknięciu w adres filmu serwer YouTube otrzymuje w nagłówku informację, że wejście na film odbyło się z serwisu FaceBook.</p>
<p>Slajd 29</p>		<p>Ustawia automatyczne przekierowanie w przeglądarce na podany adres po określonym czasie</p>

Człowiek - najlepsza inwestycja

<p>Slajd 30</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Nagłówek Retry-After</h2> <p>Retry-After: Fri, 31 Dec 1999 23:59:59 GMT</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Używany razem z kodem odpowiedzi 503 (Serwer zajęty). Określa czas po którym serwer będzie w stanie odpowiedzieć. Czas może być podany w formie konkretnej daty (jak w przykładzie) lub liczby sekund.</p>
<p>Slajd 31</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Nagłówek Server</h2> <p>Server: Apache/2.0.50</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Nagłówek identyfikujący serwer i użyte w nim oprogramowanie.</p>
<p>Slajd 32</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Nagłówek Set-Cookie</h2> <p>Set-Cookie: ciastko1=wartosc1; ciastko2=wartosc2</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Nagłówek wysłany od serwera nakazuje przeglądarce ustawienie określonych ciasteczek.</p>

Człowiek - najlepsza inwestycja

<p>Slajd 33</p>	 <p>Nagłówek Transfer-Encoding</p> <p>Transfer-Encoding: chunked</p> <p>PHP: Hypertext Preprocessor</p> <p>DAILY GROUP PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Określa w jaki sposób serwer przesłał zawartość dokumentu.</p> <p>Dostępne metody to:</p> <ul style="list-style-type: none"> •chunked <p>Oznacza przesyłanie zawartości kawałkami. Każdy z nich powinien mieć postać: wielkość treść wielkość – jest to długość treści w bajtach ,zapisane w formie szesnastkowej.</p> <p>Przykład: 1a abcdefghijklmnopqrstuvwxyz z 10 1234567890abcdef 0</p> <p>Zero w ostatniej linijce określa zerowej długości kawałek i kończy transmisję.</p> <ul style="list-style-type: none"> •identity <p>Oznacza tekst nieskompresowany</p> <ul style="list-style-type: none"> •gzip <p>Oznacza dane skompresowane algorytmem ZIP</p> <ul style="list-style-type: none"> •compress <p>Dane skompresowane algorytmem LZW</p>
<p>Slajd 34</p>	 <p>Nagłówek User-Agent</p> <p>User-Agent: Mozilla/5.0 (X11; U; Linux i686; pl; rv:1.8.0.1) Gecko/20060124 Firefox/1.5.0.1</p> <p>PHP: Hypertext Preprocessor</p> <p>DAILY GROUP PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Identyfikuje przeglądarkę, silnik html oraz System operacyjny.</p>

Człowiek - najlepsza inwestycja

<p>Slajd 35</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Nagłówek WWW-Authenticate</h2> <p>WWW-Authenticate: rodzaj_authentyfikacji</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Przesyłany razem z kodem 401. Określa sposób w jaki ma zostać przeprowadzona identyfikacja użytkownika.</p>
<p>Slajd 36</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Funkcja getallheaders() oraz apache_response_headers()</h2> <ul style="list-style-type: none"> • array getallheaders (void) • array apache_response_headers (void)  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Funkcja ‘getallheaders’ pobiera wszystkie nagłówki żądania http i parsuje je do tablicy PHP, a funkcja ‘apache_response_headers’, parsuje wszystkie nagłówki odpowiedzi http.</p>
<p>Slajd 37</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Funkcja getallheaders</h2> <ul style="list-style-type: none"> • array getallheaders (void) • array apache_response_headers (void) <p>Wykonaj skrypty wyświetlające rezultaty ww funkcji.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Wykonaj skrypty z lekcji 4, wyświetlające rezultaty funkcji getallheaders(); oraz apache_response_headers();</p>

Człowiek - najlepsza inwestycja





KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
38



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext Preprocessor

Funkcja header


```
void header ( string $string [, bool $replace = true [, int $http_response_code ]] )
```



 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Funkcja 'header' służy do przesyłania nagłówka http dokumentu. Pierwszym argumentem jest treść nagłówka w postaci ciągu znaków PHP. Drugim argumentem jest argument opcjonalny, mówi o tym, czy funkcja ma zastąpić nagłówek tego samego typu przygotowany przez serwer. Kolejnym argumentem opcjonalnym jest narzucony serwerowi kod odpowiedzi.

 Slajd
39

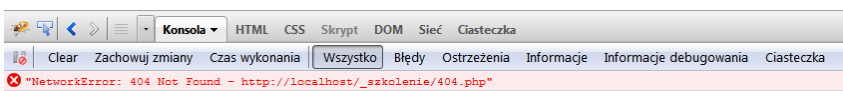

B2E
 BUSINESS TO EDUCATION



 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext Preprocessor

Przykłady zastosowania header()

```
<?php
    header("HTTP/1.0 404 Not Found");
?>
```




 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Przy odpowiedniej konfiguracji serwera HTTP, język PHP może zwracać komunikaty o braku zasobów np. strony o podanym adresie. W tym wypadku znany wszystkim błąd 404.

Zrzut ekranu przedstawia narzędzie Firebug. Jeśli przeglądarka Firefox nie zawiera wspomnianego narzędzia, trzeba je pobrać oraz zainstalować (Firefox -> Dodatki -> Rozszerzenia).

Człowiek - najlepsza inwestycja



Przykłady zastosowania header()

```
<?php
header("Location: http://przyklad.pl/");
exit();
?>
```



Proste przekierowanie HTTP na podany adres.



Przykłady zastosowania header()

```
<?php
$file = 'przyklad.php';
header('Content-
disposition: attachment; filename=' . basename($file));
header('Content-type: text/plain');
readfile($file);
?>
```



Przykład dotyczy wykorzystania nagłówków do pobrania pliku typu PHP jako tekst.



Slajd
42PHP: Hypertext
Preprocessor

Przykłady zastosowania header()

```
header('Cache-Control: no-cache, no-store, max-age=0,  
must-revalidate');
```

```
header('Expires: Mon, 26 Jul 1997 05:00:00 GMT');
```

```
header('Pragma: no-cache');
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Powyższy przykład spowoduje wyłączenie cache'u.

Człowiek - najlepsza inwestycja

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCIUNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego



Przykłady zastosowania header()

```
<?php
header('Content-Type: text/html; charset=utf-8');
echo 'ąęźźńś';
?>
```





Powyższy przykład ustawia typ kodowanie pliku na utf-8.

UWAGA: nagłówek informuje przeglądarkę, że będzie wysyłał treść w kodowaniu UTF-8. Aby polskie znaki diakrytyczne prawidłowo wyświetliły się w przeglądarce, plik zawierający polskie znaki również musi być zakodowany w UTF-8.



Slajd
 44


B2E
 BUSINESS TO EDUCATION



 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

Przykład zastosowania kodu HTML w PHP - przekierowanie

```

<?php
echo "<meta http-equiv='refresh' content='0;url=http://przyklad.pl'>";
?>
```


 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Przekierowanie lub odświeżenie strony wygenerowane przed kod php.


Po załadowaniu strony przeglądarka wykonuje kod HTML i przechodzi na adres <http://przyklad.php>


Proszę zwrócić uwagę na fakt, iż jest to inna metoda niż:

```
header('Location: http://przyklad.pl');
```

gdzie przeglądarka bez ładowania strony natychmiast wykonuje przekierowanie.

 Slajd
 45



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

Ćwiczenia praktyczne

Wykonaj skrypt, który zwróci do przeglądarki odpowiedź o braku znalezienia strony (404 – Not Found). Sprawdź działanie skryptu poprzez analizę nagłówka zwróconej odpowiedzi HTTP przy pomocy dowolnego narzędzia (Firebug w Firefox, Narzędzie dla programistów – Chrome)

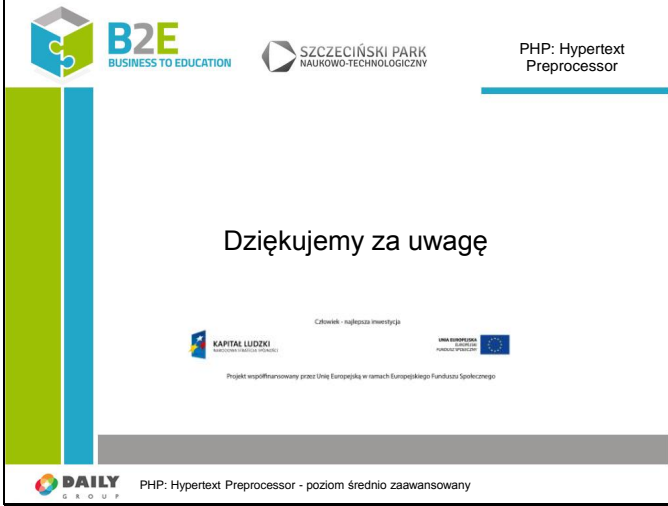

 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

404.php

Człowiek - najlepsza inwestycja

<p>Slajd 46</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenia praktyczne</h3> <p>Wykonaj skrypt, który najpierw odczeka 10 sekund a następnie zmodyfikuje element nagłówka „Location” i przekieruje użytkownika na dowolny inny adres.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>location.php</p>
<p>Slajd 47</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenia praktyczne</h3> <p>Wykonaj skrypt, który wymusi na przeglądarce rozpoczęcie pobieranie zdefiniowanego przez Ciebie pliku oraz ukryje przed pobierającym oryginalną lokalizację pliku.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>downloadFile.php</p>
<p>Slajd 48</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenia praktyczne</h3> <p>Wykonaj skrypt, który z poziomu PHP wyświetli nagłówki HTTP żądania oraz odpowiedzi.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>get.php</p>

Człowiek - najlepsza inwestycja

<p>Slajd 49</p>		<p>referer1.php, referer.php</p>
<p>Slajd 50</p>		

10.4.3 Ćwiczenia

Jako ćwiczenia należy przećwiczyć z uczniami przykłady podane na slajdach.

10.4.4 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie będą znali nagłówki protokołu HTTP oraz będą potrafili stosować je w swoich skryptach.

Człowiek - najlepsza inwestycja

10.5 Lekcja 5 – Cookie - informacje o użytkowniku (Budowa i działanie cookie, Używanie cookie w aplikacjach, Definiowanie i obsługa z poziomu PHP, Bezpieczeństwo)

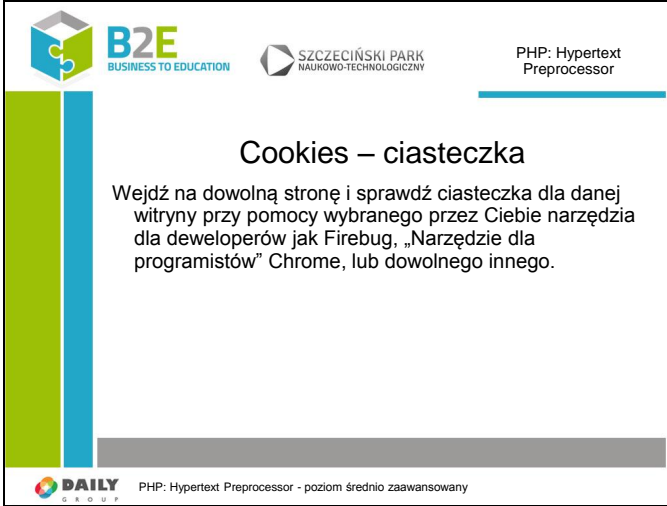
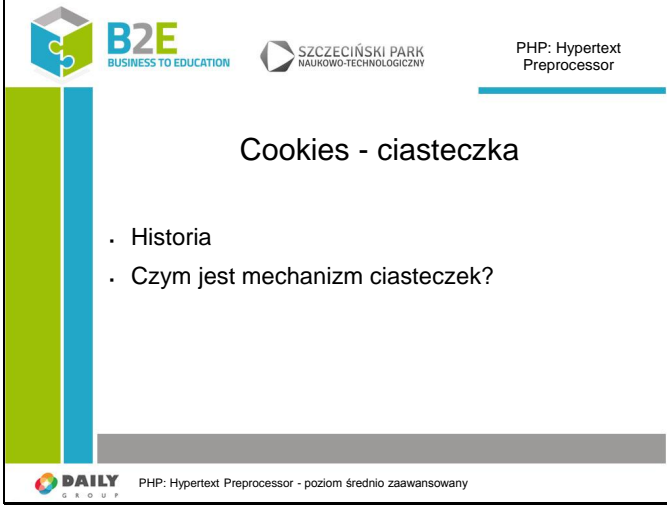
10.5.1 Cel lekcji

Celem lekcji jest wyjaśnienie uczniom na czym polega mechanizm ciasteczek i jak stosować go w języku PHP. Opisano również wady i zalety stosowania tego mechanizmu oraz do czego należy go stosować.

10.5.2 Treść - slajdy z opisem



<p>Slajd 1</p>		<p>Słowa wstępu.</p>																																																															
<p>Slajd 2</p>	 <table border="1"><thead><tr><th>Nazwa</th><th>Zawartość</th><th>Domena</th><th>Wielkość</th><th>Ścieżka</th><th>Wygasa</th><th>Tylko HTTP</th><th>Bezpieczeństwo</th></tr></thead><tbody><tr><td>PREF</td><td>ID=d0346cc9d4a3eee2:U=4...8846:S=D2EuIpoelzEeOByv</td><td>.google.pl</td><td>100 B</td><td>/</td><td>23 listopad 2014 20:27:27</td><td></td><td></td></tr><tr><td>SID</td><td>DQAAAMyAAAgd8G--9eHfM4...PzMTWPXqZ5CWha9_wTp0ZBF</td><td>.google.pl</td><td>291 B</td><td>/</td><td>16 luty 2023 17:16:29</td><td></td><td></td></tr><tr><td>HSID</td><td>ALJVoYPfKt14cSHpP</td><td>.google.pl</td><td>21 B</td><td>/</td><td>16 luty 2023 17:16:29</td><td>HttpOnly</td><td></td></tr><tr><td>SSID</td><td>Ao8ApCH-M63NyST1</td><td>.google.pl</td><td>21 B</td><td>/</td><td>16 luty 2023 17:16:29</td><td>HttpOnly</td><td>Bezpieczne</td></tr><tr><td>APISID</td><td>d0vzWOKCoNPSySM1jAVqkZFD2DPz-Ukhu</td><td>.google.pl</td><td>40 B</td><td>/</td><td>16 luty 2023 17:16:29</td><td></td><td></td></tr><tr><td>SAPISID</td><td>V7x_3Sp8m9nL_B9jA1FNZ4eF9YjKbdJH</td><td>.google.pl</td><td>41 B</td><td>/</td><td>16 luty 2023 17:16:29</td><td></td><td>Bezpieczne</td></tr><tr><td>NID</td><td>67=unReXU-WBQZqqIMf6uNd...azUC729uFEUD3ey_UmdaoUo</td><td>.google.pl</td><td>201 B</td><td>/</td><td>27 sierpień 2013 22:36:23</td><td>HttpOnly</td><td></td></tr></tbody></table>	Nazwa	Zawartość	Domena	Wielkość	Ścieżka	Wygasa	Tylko HTTP	Bezpieczeństwo	PREF	ID=d0346cc9d4a3eee2:U=4...8846:S=D2EuIpoelzEeOByv	.google.pl	100 B	/	23 listopad 2014 20:27:27			SID	DQAAAMyAAAgd8G--9eHfM4...PzMTWPXqZ5CWha9_wTp0ZBF	.google.pl	291 B	/	16 luty 2023 17:16:29			HSID	ALJVoYPfKt14cSHpP	.google.pl	21 B	/	16 luty 2023 17:16:29	HttpOnly		SSID	Ao8ApCH-M63NyST1	.google.pl	21 B	/	16 luty 2023 17:16:29	HttpOnly	Bezpieczne	APISID	d0vzWOKCoNPSySM1jAVqkZFD2DPz-Ukhu	.google.pl	40 B	/	16 luty 2023 17:16:29			SAPISID	V7x_3Sp8m9nL_B9jA1FNZ4eF9YjKbdJH	.google.pl	41 B	/	16 luty 2023 17:16:29		Bezpieczne	NID	67=unReXU-WBQZqqIMf6uNd...azUC729uFEUD3ey_UmdaoUo	.google.pl	201 B	/	27 sierpień 2013 22:36:23	HttpOnly	
Nazwa	Zawartość	Domena	Wielkość	Ścieżka	Wygasa	Tylko HTTP	Bezpieczeństwo																																																										
PREF	ID=d0346cc9d4a3eee2:U=4...8846:S=D2EuIpoelzEeOByv	.google.pl	100 B	/	23 listopad 2014 20:27:27																																																												
SID	DQAAAMyAAAgd8G--9eHfM4...PzMTWPXqZ5CWha9_wTp0ZBF	.google.pl	291 B	/	16 luty 2023 17:16:29																																																												
HSID	ALJVoYPfKt14cSHpP	.google.pl	21 B	/	16 luty 2023 17:16:29	HttpOnly																																																											
SSID	Ao8ApCH-M63NyST1	.google.pl	21 B	/	16 luty 2023 17:16:29	HttpOnly	Bezpieczne																																																										
APISID	d0vzWOKCoNPSySM1jAVqkZFD2DPz-Ukhu	.google.pl	40 B	/	16 luty 2023 17:16:29																																																												
SAPISID	V7x_3Sp8m9nL_B9jA1FNZ4eF9YjKbdJH	.google.pl	41 B	/	16 luty 2023 17:16:29		Bezpieczne																																																										
NID	67=unReXU-WBQZqqIMf6uNd...azUC729uFEUD3ey_UmdaoUo	.google.pl	201 B	/	27 sierpień 2013 22:36:23	HttpOnly																																																											

Człowiek - najlepsza inwestycja

<p>Slajd 3</p>		<p>Można tutaj użyć np. standardowo w budowanego w przeglądarkę Chrome „Narzędzia dla programistów” wchodząc w: Ustawienia->Narzędzia->Narzędzia dla programistów (kombinacja Ctrl+Shift+I)</p>
<p>Slajd 4</p>		<p>Mechanizm ciasteczek został wymyślony przez pracownika firmy Netscape Communications. Ciasteczko to niewielka porcja danych wysyłana przez serwer WWW i zapisywana w danych przeglądarki po stronie klienta. Dostęp do nich ma tylko serwer, który je wysłał, czyli mówiąc prosto każda domena ma swoje własne ciasteczka po stronie klienta, o ile mechanizm jest włączony i ciasteczka dla konkretnego adresu są akceptowane. Obecnie wykorzystuje się je głównie do prowadzenia statystyk, bądź zliczania wyświetleń reklam.</p>

Człowiek - najlepsza inwestycja


Slajd
5



PHP: Hypertext
Preprocessor

Cookies w PHP



- Gdzie można je znaleźć – tablica `$_COOKIE`?
- Jak z nich korzystać?



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Wszystkie ciasteczka dla danej domeny przechowywane są w superglobalnej tablicy `$_COOKIE`. PHP udostępnia kilka funkcji pomocniczych do mechanizmu ciasteczek np. `setcookie`.


Slajd
6



PHP: Hypertext
Preprocessor

Set Cookie – Tworzymy ciasteczko

- `bool setcookie (string $name [, string $value [, int $expire = 0 [, string $path [, string $domain [, bool $secure = false [, bool $httponly = false]]]]]])`
- `bool setrawcookie (string $name [, string $value [, int $expire = 0 [, string $path [, string $domain [, bool $secure = false [, bool $httponly = false]]]]]])`



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Ciasteczko ustawiamy (tworzymy) używając funkcji `setcookie` lub analogicznej funkcji `setrawcookie`, które różnią się tylko tym, że `setrawcookie` nie używa automatycznego enkodowania adresu url. Wymagany jest pierwszy parametr, czyli nazwa ciasteczka. Pozostałe parametry są opcjonalne, dlatego umieszczone w nawiasach kwadratowych. Drugim parametrem jest wartość ciasteczka. Kolejnym data do której ciasteczko jest ważne w formacie Unixowym, czyli daty i czasu skonwertowanego do liczby całkowitej. Czwartym parametrem jest ścieżka pod którą ciasteczka mają być dostępne np. `/`, pozwala na dostęp w dowolnej ścieżce. Piątym parametrem jest nazwa domeny, która ogranicza dostęp do ciasteczka dla domen

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



zewnętrznych, najbardziej zewnętrzną domeną, która pozwala na dostęp dla całej witryny jest np. nazwa_domeny.pl. Wszystkie domeny wewnętrzne np. www.nazwa_domeny.pl, będą miały dostęp do ciasteczek. Następne parametry określają dostęp do ciasteczka ze względu na używany protokół. Secure mówi, że ciasteczko będzie tylko dostępne jeżeli ustawione jest połączenie HTTPS., a httponly, że jest dostępne tylko za pomocą protokołu HTTP. Funkcja zwraca true (prawdę), jeżeli ciasteczko zostało ustawione, false (fałsz) w przeciwnym wypadku.


Należy dodatkowo zwrócić uwagę na istotny fakt, iż próba ustawienia ciasteczka musi nastąpić przed wysłaniem jakiegokolwiek treści do przeglądarki. W przypadku nie spełnienia tego warunku otrzymamy ostrzeżenie:

Warning: Cannot modify header information - headers already sent.


Jak można wywnioskować z ostrzeżenia, ustawianie ciasteczek jest realizowane przez wysłanie stosownych nagłówków.

Przykład nieprawidłowego użycia setCookie dostępny jest w pliku: wrong.php

Slajd
7



B2E
 BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Read Cookie – odczytujemy ciasteczko

- Do odczytu używamy tablicy oraz klucza(nazwy)


```
if(isset($_COOKIE['cookie_name']))
    $cookie = $_COOKIE['cookie_name'];
```





PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Odczyt ciasteczek jest o wiele prostszy, niż ich tworzenie. Posługujemy się tablicą \$_COOKIES oraz nazwą ciasteczka. Z racji możliwości kasowania ciasteczek przez użytkowników, należy się upewnić, że owe ciasteczko istnieje, możemy wykorzystać do tego funkcję isset.


Ze względu na fakt, iż skrypty php przetwarzane są po stronie serwera, a ciastka przechowywane są po stronie klienta (w przeglądarce) przy pierwszym żądaniu

Człowiek - najlepsza inwestycja


	<p>serwer otrzymuje stan ciasteczek z poprzedniego żądania, następnie wykonuje skrypt w którym ustawia ciastko. Wysyła odpowiedź HTTP z nowo ustawionym ciastkiem. Dopiero w drugim żądaniu otrzymuje ciastko ustawione w pierwszym żądaniu i może je przetworzyć.</p> <p>Przykład: wyobraźmy sobie skrypt cookie.php który ustawia oraz odczytuje ciastko - gdy ono istnieje.</p> <ul style="list-style-type: none"> • Pierwsze przetworzenie skryptu serwer wyśle do przeglądarki informację o stworzeniu ciastka, ale zawartość ciastka nie wyświetli się • Drugie przetworzenie skryptu – do serwera dociera ciastko utworzone w poprzednim wywołaniu, strona serwerowa przetwarza jego zawartość i generuje rezultat polegający na wyświetleniu zawartości ciastka, który przesyła do przeglądarki
<p>Slajd 8</p>	<div data-bbox="280 730 962 1238">  <h3>Ćwiczenie – tworzymy ciasteczko</h3> <ul style="list-style-type: none"> • Wykorzystując funkcję setcookie ustaw ciasteczko o nazwie 'name', zawierające wartość Twojego imienia oraz następnie sprawdź czy istnieje jeżeli tak, wyświetl komunikat: Hello, 'ciasteczko_z_imieniem'! (skrypt nazwij name.php) Jeżeli nie ma ciasteczka lub jest puste, wypisz tylko 'Hello'. Odśwież stronę dwa razy.  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p> </div> <div data-bbox="999 730 1402 909"> <p>Wykonać ćwiczenie, na następnej stronie znajduje się rozwiązanie ćwiczenia, nie pokazywać dopóki studenci nie skończą.</p> </div>

Człowiek - najlepsza inwestycja

Slajd
9



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Rozwiązanie i opis

```
<?php
if (!isset($_COOKIE['name']))
    setcookie('name', 'Stefan');
if (isset($_COOKIE['name']) and !empty($_COOKIE['name']))
    echo 'Hello, ' . $_COOKIE['name'] . '!';
else
    echo 'Hello!';
?>
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Rozwiązanie poprzedniego ćwiczenia.

W pierwszej kolejności sprawdzamy, czy ciasteczko istnieje, jeżeli nie to je tworzymy. Następnie jeżeli ciastko istnieje i nie jest puste wypisujemy 'Hello, imię', a jeżeli istnieje i jest puste wypisujemy tylko 'Hello'.

Slajd
10



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor














PHP: Hypertext Preprocessor - poziom średnio zaawansowany



Chwila rozluźnienia.

Człowiek - najlepsza inwestycja


<p>Slajd 11</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Overwrite Cookie – nadpisywanie ciasteczka</h2> <ul style="list-style-type: none"> • Ponowne wywołanie funkcji setcookie, bez sprawdzenia, czy ciasteczko już istnieje  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Sprawdzenie istnienie ciasteczka za pomocą funkcji 'isset', jest rzeczą istotną, ponieważ jeżeli ustawimy ciasteczko o nazwie istniejącego ciasteczka, zostanie ono nadpisane nowym. Mechanizm ten jest wykorzystywany w praktyce, z tym, że dobrze jest rozumieć jak on działa i używać go świadomie, ze względu na możliwe problemy w aplikacjach.</p>
<p>Slajd 12</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Delete Cookie – usuwanie ciasteczka</h2> <ul style="list-style-type: none"> • <code>setcookie("hello", "", time()-3600);</code>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Jedyną metodą usunięcia ciasteczka jest stworzenie ciastka, którego okres ważności się skończył. W powyższym przykładzie ustawiliśmy ważność ciasteczka na czas obecny minus 3600 sekund, czyli na godzinę wstecz.</p>
<p>Slajd 13</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h2>Ćwiczenie</h2> <ul style="list-style-type: none"> • Wykorzystując skrypt napisany w poprzednim ćwiczeniu, usunąć ciasteczko i zmodyfikować skrypt w taki sposób, aby nie ustawiać ponownie ciasteczka. Odświeżyć przeglądarkę dwa razy.  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Wykonać ćwiczenie, na następnej stronie znajduje się rozwiązanie ćwiczenia, nie pokazywać dopóki studenci nie skończą.</p>

Człowiek - najlepsza inwestycja

Slajd
14

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY



PHP: Hypertext
Preprocessor

Rozwiązanie

```
<?php
if (isset($_COOKIE['name']))
    setcookie('name', '', time() - 3600);
if (isset($_COOKIE['name']) and !empty($_COOKIE['name']))
    echo 'Hello, ' . $_COOKIE['name'] . '!';
else
    echo 'Hello!';
?>
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Rozwiązanie poprzedniego ćwiczenia.


W pierwszej kolejności sprawdzamy, czy ciasteczko istnieje, aby następnie ustawić jego datę ważności na godzinę wstecz.

Następnie jeżeli istnieje, wypisujemy 'Hello, ciastko', bądź jeżeli je usunęliśmy tylko 'Hello'.

Slajd
15

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY



PHP: Hypertext
Preprocessor

Dlaczego odświeżamy przeglądarkę dwa razy?


- Ustawiamy ciastko
- Pobieramy ciastko

PHP: Hypertext Preprocessor - poziom średnio zaawansowany


Skrypty PHP wykonywane są po stronie serwera w wyniku zgłoszenia żądania HTTP w którym wysyłamy ciasteczka dla danego serwera. Przy pierwszym żądaniu serwer otrzymuje stan ciasteczek z poprzedniego żądania, następnie wykonuje skrypt w którym ustawia ciastko. Wysyła odpowiedź HTTP z nowo ustawionym ciastkiem. Dopiero w drugim żądaniu otrzymuje ciastko ustawione w pierwszym żądaniu i może je przetworzyć.

Człowiek - najlepsza inwestycja

Slajd
16



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Ćwiczenie – przykładowe zastosowanie

- Wykorzystując wiedzę z poprzednich slajdów i ćwiczeń. Napisać licznik odwiedzin, wykorzystujący mechanizm ciasteczek.




DAILY GROUP


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Wykonać ćwiczenie, na następnej stronie znajduje się rozwiązanie ćwiczenia, nie pokazywać dopóki studenci nie skończą.

Slajd
17



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Rozwiązanie

```
<?php
if (!isset($_COOKIE['counter']))
    setcookie('counter', 2, time() + 60 * 60 * 24 * 365);
if (isset($_COOKIE['counter'])) {
    setcookie('counter', $_COOKIE['counter'] + 1, time() + 60 * 60 * 24 * 365);
    echo 'Odwiedzasz stronę ' . $_COOKIE['counter'] . ' raz!';
} else {
    echo 'Odwiedzasz stronę 1 raz!';
}

?>
```



DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Rozwiązanie poprzedniego ćwiczenia.

Jeżeli ciastko licznika nie istnieje ustawiamy je na dwa, ponieważ, zostanie wyświetlone dopiero przy następnej wizycie i wyświetlamy ciąg znaków z wyrażenia warunkowego po 'else'. W przypadku gdy ciastko jest już ustawione, wyświetlamy jego wartość i zwiększamy o jeden. Kolejność nie ma znaczenia, gdyż jak wcześniej wspomniano ustawione zostanie ono dopiero w odpowiedzi HTTP po stronie klienta, aby zostać wysłane przy kolejnym żądaniu HTTP.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
18



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Rozwiązanie podgląd

Dla zrealizowanego zadania wyświetl zawartość ciasteczka w firebug'u.

Ciasteczka • Filtr • Akceptuj ciasteczka (domyślne) •

Nazwa	Zawartość	Domena	Wielkość	Ścieżka	Wygasa	Tylko HTTP	Bezpieczeństwo
PHPSESSID	5f8b9f72a3cd3e95a3810ff5c27e69b4	localhost	41 B	/	Sesja		
counter	3	localhost	8 B	/_szkolenie/	Sesja		

Zawartość

3



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Rozwiązanie poprzedniego ćwiczenia.

Jeżeli ciastko licznika nie istnieje ustawiamy je na dwa, ponieważ, zostanie wyświetlone dopiero przy następnej wizycie i wyświetlamy ciąg znaków z wyrażenia warunkowego po 'else'. W przypadku gdy ciastko jest już ustawione, wyświetlamy jego wartość i zwiększamy o jeden. Kolejność nie ma znaczenia, gdyż jak wcześniej wspomniano ustawione zostanie ono dopiero w odpowiedzi HTTP po stronie klienta, aby zostać wysłane przy kolejnym żądaniu HTTP.

Człowiek - najlepsza inwestycja








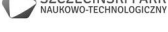



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

<p>Slajd 19</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Zalety ciasteczek</h3> <ul style="list-style-type: none"> • Prywatność - są dostępne tylko dla użytkownika w jego przeglądarce i nikt z zewnątrz nie ma do nich dostępu • Użytkownik ma na nie bezpośredni wpływ, co może być również wadą, ale w niektórych przypadkach jest zdecydowanie zaletą np. możliwość ich blokowania dla konkretnych stron  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Przedyskutować wypunktowane kwestie.</p>
<p>Slajd 20</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Zalety ciasteczek</h3> <ul style="list-style-type: none"> • Łatwość w ich obsłudze z poziomu języka PHP • Ułatwienia dla użytkownika, np. nie musi wypełniać formularza, kolejny raz • Możliwość tworzenia efektywnych kampanii reklamowych oraz prowadzenia statystyk (zaleta dla właściciela portalu)  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Przedyskutować wypunktowane kwestie.</p>
<p>Slajd 21</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Wady ciasteczek</h3> <ul style="list-style-type: none"> • Ciasteczka muszą być dołożone do każdego żądania HTTP • Mają ograniczony rozmiar z góry przez przeglądarkę • Bez użycia HTTPS są widoczne i niezaszyfrowane w przeglądarce • Użytkownik może zmieniać dane w nich zawarte  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Przedyskutować wypunktowane kwestie.</p>










Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



<p>Slajd 22</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Wady ciasteczek</h3> <ul style="list-style-type: none"> · Użytkownik może je usunąć · Użytkownik może zablokować ciasteczka · Zależą od przeglądarki, czyli ich bezpieczeństwo zależy od konkretnej implementacji, czasami wadliwej, wykazano luki w niektórych przeglądarkach  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Przedyskutować wypunktowane kwestie.</p>
<p>Slajd 23</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Podsumowanie</h3> <ul style="list-style-type: none"> · Czy warto je stosować? · Do czego warto je stosować?  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Ciasteczka ułatwiają życie zarówno użytkownikowi, programiście oraz właścicielowi portalu. W niektórych kwestiach warto je stosować np. w statystykach, autouzupełnianiu formularzy, kampaniach reklamowych czy jeżeli używamy szyfrowania np. SSL, również do autoryzacji.</p>
<p>Slajd 24</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Ćwiczenie praktyczne</h3> <ul style="list-style-type: none"> · Wykonaj skrypt, który ustawi czas życia ciastka na 10 sekund oraz wyświetli zawartość ciastka. Sprawdź co stanie się gdy ponownie uruchomisz skrypt przed upływem 10s oraz po upływie tego czasu. Co stanie się gdy ponownie odświeżymy stronę po upłynięciu czasu życia ciastka?  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>cookieLifeTime.php</p>

Człowiek - najlepsza inwestycja

Slajd
25



Slide 25 content: The slide features a header with logos for B2E (Business to Education), Szczeciński Park Naukowo-Technologiczny, and PHP: Hypertext Preprocessor. The main text reads 'Dziękujemy za uwagę' (Thank you for your attention). Below this, there are logos for 'KAPITAŁ LUDZKI' (Human Capital) and the European Union flag, with the text 'Człowiek - najlepsza inwestycja' (Human - the best investment). At the bottom, there is a logo for 'DAILY' and the text 'PHP: Hypertext Preprocessor - poziom średnio zaawansowany' (PHP: Hypertext Preprocessor - intermediate level).

10.5.3 Ćwiczenia

Ćwiczenia zostały przedstawione na slajdach. Podczas prowadzenia zajęć należy wykonywać je razem z uczniami.

10.5.4 Opis założonych osiągnięć ucznia

Uczeń powinien posiadać wiedzę na temat mechanizmu ciasteczek, poznać schemat jego działania oraz potrafić korzystać z niego w swoich skryptach.

10.6 Lekcja 6 - Sesje - współdzielenie informacji (Struktura i sposób działania, Kontrola sesji w PHP, Wykorzystanie sesji w aplikacjach internetowych)

10.6.1 Cel lekcji

Celem lekcji jest zaprezentowanie mechanizmu sesji, opis jego wykorzystania w skryptach i nauczanie uczestników kursu wykorzystywać powyższy mechanizm w ich skryptach.

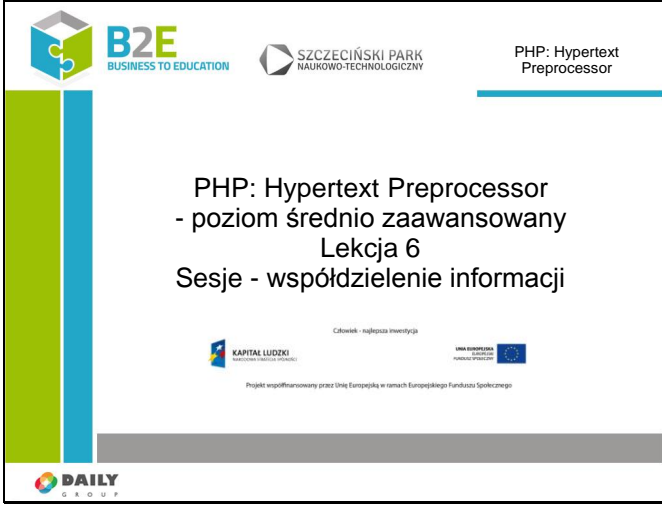
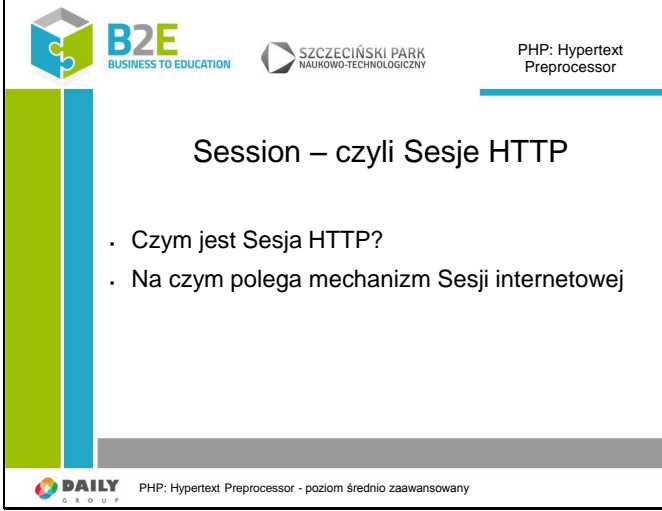
10.6.2 Ćwiczenia

Ćwiczenia zostały przedstawione na slajdach. Podczas prowadzenia zajęć należy wykonywać je razem z uczniami.

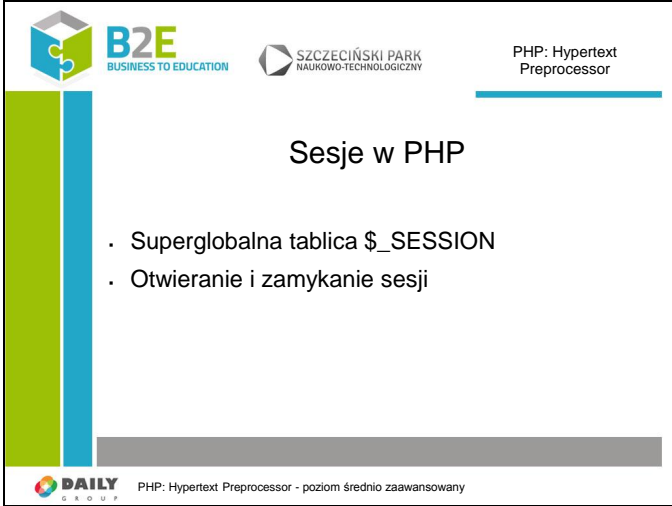

Człowiek - najlepsza inwestycja












10.6.3 Treść - slajdy z opisem

<p>Slajd 1</p>		<p>Słowa wstępu.</p>
<p>Slajd 2</p>		<p>Sesja HTTP to mechanizm przechowywania danych po stronie serwera. Parametry sesji takie jak np. czas trwania sesji, znajdują się w pliku konfiguracyjnym serwera. W przeciwieństwie do mechanizmu ciasteczek znanego z poprzedniej lekcji. Sesje są niezależne od klienta, dzięki czemu dają programiście większą kontrolę nad danymi w nich przechowywanymi. Każda sesja posiada swój identyfikator, który jest unikatowy i jeden dla każdego klienta. Klientem w rozumieniu sesji jest np. przeglądarka, a nie adres komputera osoby z niej korzystającej. Konsekwencją tego jest tworzenie kilku sesji, dla jednego adresu komputera, w przypadku np. zalogowania się z różnych przeglądarek, czyli możliwość wielokrotnego logowania się na np. jedno konto, w</p>

Człowiek - najlepsza inwestycja

		przypadku nie zaimplementowanej możliwości jednokrotnego dostępu z poziomu jednego konta.
Slajd 3		Sesje w przeciwieństwie do ciasteczek, są mechanizmem bardziej rozbudowanym. Podobnie jak ciasteczka posiadają tablicę superglobalną, dostępną w każdym miejscu aplikacji PHP. Dane z sesji są dostępne w skrypcie od razu po ich stworzeniu (ustawieniu), nie ma potrzeby odświeżania strony, żeby korzystać z sesji trzeba ją jednak najpierw otworzyć lub stworzyć, możemy ją również zamknąć wedle naszej potrzeby.
Slajd 4		Aby zainicjalizować sesję, używamy funkcji 'session_start', zwraca ona zawsze wartość true i nie przyjmuje żadnych argumentów. Aby pobrać stan sesji stosujemy funkcję 'session_status', zwraca 0 jeżeli sesje są wyłączone po stronie serwera, 1 jeżeli sesja nie istnieje lub jest nieaktywna bądź, 2 jeżeli istnieje.










Człowiek - najlepsza inwestycja

<p>Slajd 5</p>	  <div>PHP: Hypertext Preprocessor</div> <h3>Identyfikator sesji</h3> <ul style="list-style-type: none"> • string session_id([string \$id]) • bool session_regenerate_id ([bool \$usunięcie_starej_sesji = false])  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>Funkcja session_id(), zwraca nam identyfikator obecnej sesji lub jeżeli dostanie identyfikator jako parametr, ustawia identyfikator sesji, w tym przypadku należy ją wywołać przed funkcją session_start. Kolejna funkcja zastępuje obecny identyfikator sesji, nowym, a argument decyduje o tym, czy usunąć zmienianą sesję, czy nie.</p>
<p>Slajd 6</p>	  <div>PHP: Hypertext Preprocessor</div> <h3>Ćwiczenie – tworzenie sesji</h3> <ul style="list-style-type: none"> • Napisz skrypt, który stworzy nową sesję oraz wyświetli jej identyfikator, którym będzie Twoje imię (nie stosuj polskich znaków).  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>Wykonać ćwiczenie, na następnej stronie znajduje się rozwiązanie ćwiczenia, nie pokazywać dopóki studenci nie skończą.</p>
<p>Slajd 7</p>	  <div>PHP: Hypertext Preprocessor</div> <h3>Rozwiązanie</h3> <pre><?php session_id('Ewa'); if (session_id() == '') session_start(); echo session_id(); ?></pre>  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>W pierwszej kolejności ustalamy nazwę sesji na nasze imię. Następnie sprawdzamy czy sesja istnieje, jeżeli nie istnieje tworzymy nową lub startujemy wcześniej utworzoną. Na koniec wypisujemy identyfikator sesji.</p>

Człowiek - najlepsza inwestycja

<p>Slajd 8</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Ćwiczenie – tworzenie sesji</h3> <p>W pliku konfiguracyjnym php.ini znajdź lokalizację w której zapisywane są sesje serwera, otwórz plik sesji i przeanalizuj jego zawartość.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Wykonać ćwiczenie.</p>
<p>Slajd 9</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Ćwiczenie – tworzenie sesji</h3> <p>Otwórz skrypt tworzący sesję w dwóch przeglądarkach, sprawdzając jednocześnie co powstaje w folderze z sesjami.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Wykonać ćwiczenie.</p>
<p>Slajd 10</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Dodawanie danych do sesji</h3> <ul style="list-style-type: none"> • session_register (stara metoda) • \$_SESSION['key'] = \$value;  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Aby dodać dane do wcześniej rozpoczętej sesji, należy użyć drugiej metody ze slajdu. Pierwsza metoda jest przestarzała i niepoprawna w najnowszej implementacji PHP, działa w wersji PHP < 5.4. Wartości zapisujemy operując bezpośrednio na superglobalnej tablicy sesji.</p>

Człowiek - najlepsza inwestycja



<p>Slajd 11</p>	  <div>PHP: Hypertext Preprocessor</div> <hr/> <h3>Nadpisywanie danych w sesji</h3> <ul style="list-style-type: none"> • <code>\$_SESSION['key'] = \$value;</code>  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>Jeżeli klucz w sesji istnieje, a my dodamy od niego nowe dane, zostanie on nadpisany. Tablica sesji działa dokładnie tak jak inne tablice w języku PHP.</p>
<p>Slajd 12</p>	  <div>PHP: Hypertext Preprocessor</div> <hr/> <h3>Usuwanie bieżących danych sesji</h3> <ul style="list-style-type: none"> • <code>session_destroy();</code>  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>Funkcja <code>session_destroy</code> usuwa tylko bieżące dane skojarzone z sesją. Nie usuwa jednak sesji, której identyfikator przechowywany jest w ciasteczku sesji.</p>
<p>Slajd 13</p>	  <div>PHP: Hypertext Preprocessor</div> <hr/> <h3>Nazwa sesji</h3> <ul style="list-style-type: none"> • <code>string session_name ([string \$nazwa])</code>  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>Nazwa sesji - obok identyfikatora - jest używana do identyfikacji sesji. Funkcja <code>session_name</code> podobnie jak funkcja <code>session_id</code>, służy albo do zwracania nazwy sesji albo do zmieniania jej nazwy. Podobnie jak <code>session_id</code> musi być wywołana przed funkcją <code>session_start</code>.</p>

Człowiek - najlepsza inwestycja

<p>Slajd 14</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Usuwanie sesji</h2> <pre>if (isset(\$_COOKIE[session_name()])) { setcookie(session_name(), "", time()-42000, '/'); }</pre>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Aby usunąć sesję całkowicie, należy usunąć ciasteczko z jej nazwą, które jest przechowywane po stronie użytkownika. Wpierw należy sprawdzić, czy takowe istnieje, a następnie należy je usunąć, metodą podaną w poprzedniej lekcji.</p>
<p>Slajd 15</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Usuwanie danych z sesji</h2> <ul style="list-style-type: none"> • session_unset(); • unset(\$_SESSION['key']);  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Aby usunąć wszystkie dane ustawione w sesji kiedykolwiek, należy zastosować funkcję session_unset. Aby usunąć konkretną zmienną należy użyć drugiej konstrukcji.</p>
<p>Slajd 16</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Ćwiczenie – tworzenie sesji</h2> <ul style="list-style-type: none"> • Należy przygotować skrypt, który zapisze w sesji nasze imię oraz wyświetli 'Hello, nasze_imie' z sesji. Następnie zapisze do poprzedniej zmiennej nazwisko i wyświetli 'Hello, nasze_nazwisko'. Na końcu usunie całą sesję.  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Wykonać ćwiczenie, na następnej stronie znajduje się rozwiązanie ćwiczenia, nie pokazywać dopóki studenci nie skończą.</p>

Człowiek - najlepsza inwestycja

Slajd
17





**PHP: Hypertext
Preprocessor**

Rozwiązanie cz.1

```


<?php
    if (session_id() == '')
        session_start();
    if (!isset($_SESSION['name']))
        $_SESSION['name'] = 'Ewa';
    echo 'Hello, ' . $_SESSION['name'];
    $_SESSION['name'] = 'Kowalska';
    echo '<br />Hello, ' . $_SESSION['name'];
        
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

W pierwszej kolejności inicjalizujemy sesję. Następnie ustawiamy zmienną `$_SESSION['name']` na nasze imię oraz wyświetlamy ją. Kolejnym krokiem jest nadpisanie zmiennej naszym nazwiskiem oraz wyświetlenie jej.

Człowiek - najlepsza inwestycja

Slajd
18

**B2E**
BUSINESS TO EDUCATION

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP: Hypertext
Preprocessor


DAILY
GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część końcową naszego skryptu rozpoczyna usunięcie zmiennych z sesji oraz zniszczenie ciasteczka z jej identyfikatorem.

Slajd
19

**B2E**
BUSINESS TO EDUCATION

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

DAILY
GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Wykonać ćwiczenie, na następnej stronie znajduje się rozwiązanie ćwiczenia, nie pokazywać dopóki studenci nie skończą.

Człowiek - najlepsza inwestycja




KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
20





SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Rozwiązanie

```
<?php
$url = (@$_SERVER["HTTPS"] == "on") ? "https://" : "http://";
if ($_SERVER["SERVER_PORT"] != "80") {
    $url .= $_SERVER["SERVER_NAME"] . ":" .
        $_SERVER["SERVER_PORT"] . $_SERVER["REQUEST_URI"];
} else {
    $url .= $_SERVER["SERVER_NAME"] . $_SERVER["REQUEST_URI"];
}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

W pierwszej kolejności potrzebujemy adres url naszej strony, będzie nam on potrzebny do zerowania licznika.

Człowiek - najlepsza inwestycja



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Rozwiązanie

```
$url = str_replace('?clear=1', '', $url);
if (session_id() == '')
    session_start();
if (!isset($_SESSION['count']))
    $_SESSION['count'] = 0;
else
    $_SESSION['count'] = $_SESSION['count'] + 1;
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Następnie czyścimy z niego zmienne z tablicy \$_GET, w przeciwnym wypadku zerowalibyśmy licznik za każdym razem. Kolejną rzeczą jest implementacja samego mechanizmu zliczającego i przechowującego dane w sesji.



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Rozwiązanie

```
if (isset($_GET['clear']) and $_GET['clear']) {
    $_SESSION['count'] = 0;
    header('Location: ' . $url);
}
echo 'Odwiedzasz nas ' . $_SESSION['count'] . ' raz!';
echo '<br /><a href="?clear=1">Zeruj</a>';
?>
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Na koniec potrzebna jest nam akcja, która w związku ze zmienną \$_GET wyczyści

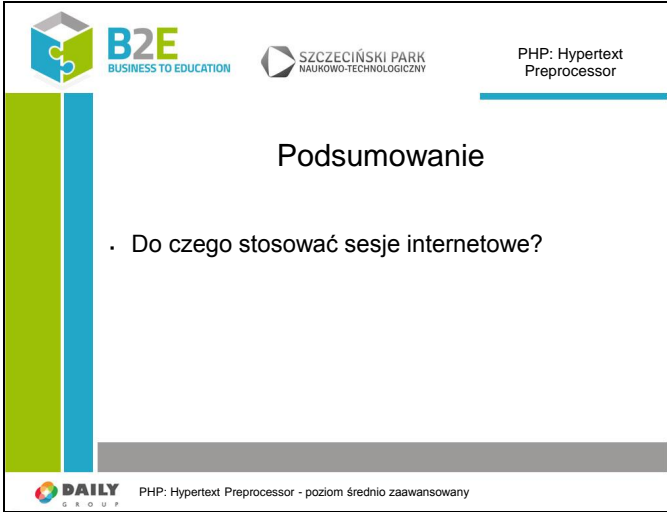
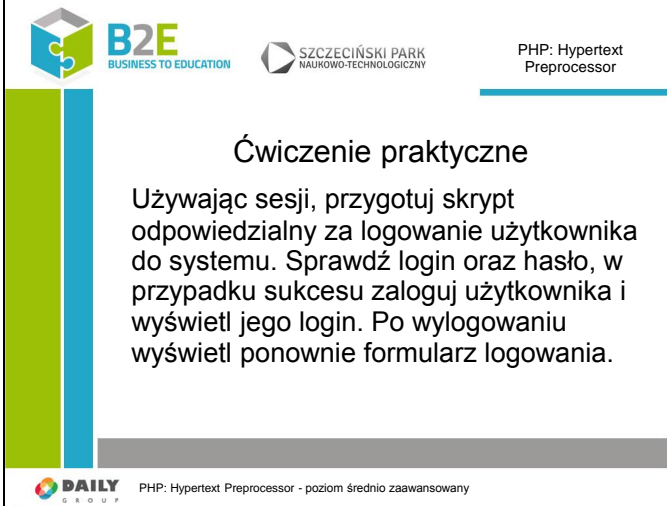
Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



	nam licznik i odświeży stronę. Kolejną rzeczą jest wyświetlenie naszego licznika oraz linku do jego zerowania.	
Slajd 23	 <p>Podsumowanie</p> <ul style="list-style-type: none"> Do czego stosować sesje internetowe? <p>PHP: Hypertext Preprocessor</p> <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Sesje HTTP służą głównie do systemów logowania oraz przechowywania danych globalnych lub konfiguracyjnych. Są bezpieczniejsze od ciasteczek jeśli chodzi o przechowywanie danych ze względu na ich umiejscowienie (serwer). Opierają się jednak o system ciasteczek, co może spowodować brak możliwości dostania się do odpowiedniej sesji.</p>
Slajd 24	 <p>Ćwiczenie praktyczne</p> <p>Używając sesji, przygotuj skrypt odpowiedzialny za logowanie użytkownika do systemu. Sprawdź login oraz hasło, w przypadku sukcesu zaloguj użytkownika i wyświetl jego login. Po wylogowaniu wyświetl ponownie formularz logowania.</p> <p>PHP: Hypertext Preprocessor</p> <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Login.php, logout.php</p>

Człowiek - najlepsza inwestycja

<p>Slajd 25</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenie praktyczne</h3> <p>Do stworzonego wcześniej systemu logowania dodaj sprawdzanie liczby nieudanych prób zalogowania (błędny login lub hasła). Po 5 nieudanych próbach zablokuj możliwość zalogowania do aplikacji.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Login.php, logout.php</p>
<p>Slajd 26</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenie praktyczne</h3> <p>Do stworzonego wcześniej ćwiczenia, dodaj sprawdzanie czasu, który upłynął od momentu zablokowania użytkownika na skutek podania nieprawidłowego loginu lub hasła. Po upływie minuty odblokuj formularz logowania i pozwól użytkownikowi na ponowną próbę logowania.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Login.php, logout.php</p>
<p>Slajd 27</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Dziękujemy za uwagę</h3> <div data-bbox="464 1585 555 1619">  <p>KAPITAŁ LUDZKI NARODOWA STRATEGIA SPÓJNOŚCI</p> </div> <div data-bbox="587 1574 678 1585"> <p>Człowiek - najlepsza inwestycja</p> </div> <div data-bbox="710 1597 786 1619">  <p>UNIA EUROPEJSKA EUROPEJSKI FUNDUSZ SPOŁECZNY</p> </div> <p><small>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</small></p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	

Człowiek - najlepsza inwestycja

10.6.4 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie będą posiadać wiedzę o mechanizmie sesji oraz będą potrafili wykorzystać go w swoich skryptach, bądź aplikacjach internetowych.

10.7 Lekcja 7 - Poczta - systemy SMTP (Systemy pocztowe w Internecie, Budowa protokołu SMTP, Komunikacja SMTP, Struktura nagłówków wiadomości, Obsługa poczty w aplikacjach PHP)

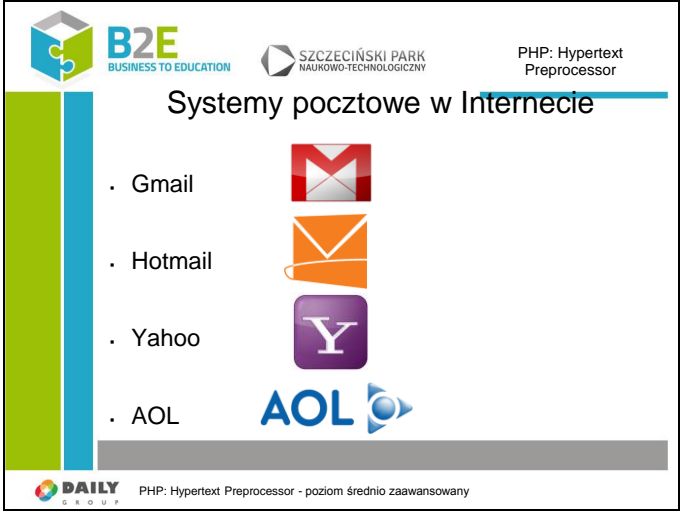

10.7.1 Cel lekcji

Celem lekcji jest zapoznanie z protokołem SMTP. Poznanie jego działania oraz nauczenie się, jak wykorzystać go w języku php, celem wysyłania poczty elektronicznej.

10.7.2 Treść - slajdy z opisem



Slajd 1	  <div style="text-align: right;">PHP: Hypertext Preprocessor</div> <div style="text-align: center;"> <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p> <p>Lekcja 7</p> <p>Poczta - systemy SMTP</p> </div> <div style="text-align: center;">   </div> <div style="text-align: center;"> <small>Człowiek - najlepsza inwestycja</small> <small>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</small> </div> 	Słowa wstępu
Slajd 2	  <div style="text-align: right;">PHP: Hypertext Preprocessor</div> <div style="text-align: center;"> <p>Simple Mail Transfer Protocol – czyli SMTP</p> <ul style="list-style-type: none"> • Co to jest? • Historia • Zastosowanie w języku PHP </div> <div style="text-align: center;">   </div> <div style="text-align: center;"> <small>Człowiek - najlepsza inwestycja</small> <small>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</small> </div> 	<p>Protokół SMTP (ang. Simple Mail Transfer Protocol) – to protokół komunikacyjny, który służy jako opis standardu przesyłania poczty elektronicznej. PO raz pierwszy został opisany w roku 1982 przez Jonathana B. Postela. Często jest wymieniany razem z protokołem POP3, który służy do pobierania poczty z serwera. SMTP jest wykorzystywany w języku PHP jako warstwa do obsługi poczty elektronicznej. Możemy z niego korzystać</p>

Człowiek - najlepsza inwestycja

		<p>bezpośrednio lub za pomocą np. obiektu phpmailer. Istnieje również wiele innych komponentów wspomagających obsługę poczty elektronicznej na serwerach WWW.</p>
Slajd 3		<p>Do najpopularniejszych systemów pocztowych w internecie należą te wyżej wymienione. Są dostępne z poziomu przeglądarki najczęściej po protokole HTTPS. Działają najczęściej w oparciu o protokoły IMAP (następca POP3) oraz SMTP. W niektórych istnieje możliwość dostępu po zdefiniowanych przez nas protokołach.</p>
Slajd 4		<p>Protokół SMTP wykorzystuje warstwę transportową TCP, domyślnie po porcie 25. Adres pocztowy zdefiniowany jest jako część związana z domeną, czyli adres serwera na serwerze DNS (Domain Name System), który zwraca adres IP danej domeny oraz nazwę użytkownika systemu pocztowego na danym serwerze.</p>

Człowiek - najlepsza inwestycja

Slajd
5


 **B2E**
BUSINESS TO EDUCATION
  SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Składnia wiadomości SMTP



```

S: 220 domena.pl ESMTP Postfix
C: HELO filip.domena.pl
S: 250 domena.pl
C: MAIL FROM:<ewa@domena.pl >
S: 250 2.1.0 Ok
C: RCPT TO:<ewa@domena.pl >
S: 250 2.1.5 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: This is the message body.
C: .
S: 250 2.0.0 Ok: queued as C2BEC5B0143
C: QUIT
S: 221 2.0.0 Bye
    
```

 **DAILY**
G R O U P
 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Składnia wiadomości SMTP wygląda całkiem podobnie do składni żądania HTTP. Można ją podzielić na część w której znajdują się nagłówki, część z treścią oraz część kończącą wiadomość. Treść zawsze musi być zakończona znakiem kropki (.) oraz znakami powrotu karetki (CR) i znakiem przejścia do nowej linii (LF).


Slajd
6

 **B2E**
BUSINESS TO EDUCATION
  SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

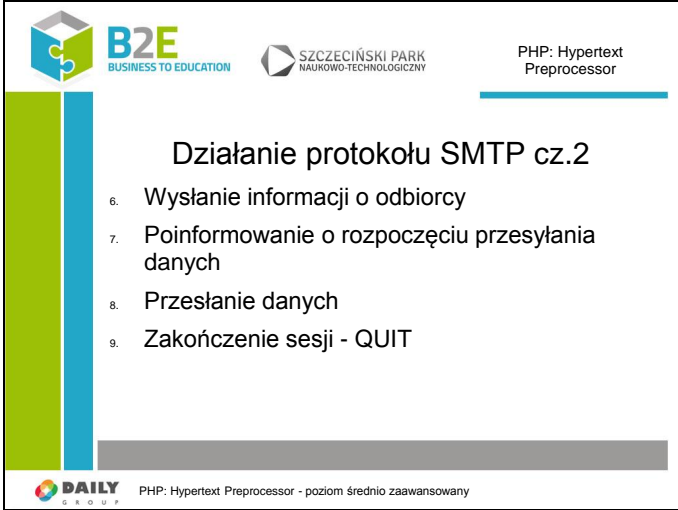
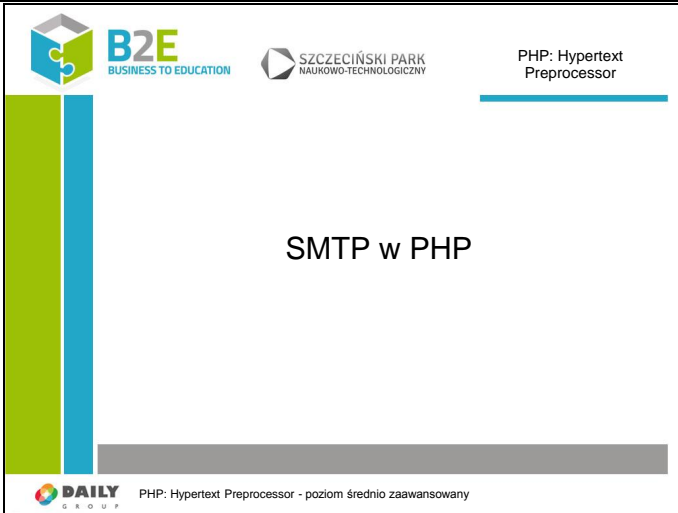
Działanie protokołu SMTP cz.1

1. Połączenie.
2. Potwierdzenie otwarcia sesji
3. Identyfikacja hosta
4. Identyfikacja nadawcy
5. Potwierdzenie gotowości na odebranie wiadomości e-mail

 **DAILY**
G R O U P
 PHP: Hypertext Preprocessor - poziom średnio zaawansowany


W pierwszej kolejności nawiązywane jest połączenie TCP hosta z serwerem, do którego ma zostać wysłana wiadomość (wysyłane są trzy wiadomości, pierwsza z hosta na serwer, druga to odpowiedź serwera, trzecia prośba o otwarcie sesji). Następnie serwer SMTP wysyła wiadomość, która potwierdza otwarcie sesji. Kolejną rzeczą jest identyfikacja hosta (wiadomość z komendą HELO, lub EHLO dla rozszerzonej wersji SMTP – ESMTP, Extended SMTP). Po


Człowiek - najlepsza inwestycja

		<p>zidentyfikowaniu hosta, klient SMTP wysyła wiadomość MAIL służącą do identyfikacji nadawcy. W dalszej części serwer potwierdza gotowość na odebranie maila (wysyła wiadomość zawierającą komendę 250 OK).</p>
Slajd 7		<p>Klient SMTP wysyła wiadomość z informacją o odbiorcy maila (RCPT TO). Serwer akceptuje dane odpowiadając komendą: 250 OK.</p> <p>Następnie klient SMTP wysyła wiadomość z komendą DATA <CRLF> informując o początku przesyłania danych maila. Serwer akceptuje komendę, odpowiadając komendą: 354.</p> <p>W końcu dane zostają przesłane, zakończone kropką i przejściem do nowej linii w formacie systemu Windows. Klient kończy sesję, wysyła wiadomość z komendą QUIT. Serwer potwierdza koniec sesji wysyłając wiadomość z kodem: 221.</p>
Slajd 8		<p>Aby PHP mogło wysyłać wiadomości via SMTP, musi mieć dostęp do serwera pocztowego albo do serwera, na którym jest zainstalowany i skonfigurowany protokół SMTP.</p>

Człowiek - najlepsza inwestycja

Slajd
 9



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

PHP funkcja mail


- `bool mail (string $to , string $subject , string $message [, string $additional_headers [, string $additional_parameters]])`



 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Do wysyłania wiadomości e-mail bezpośrednio z serwera, na którym SMTP jest skonfigurowane, można użyć funkcji mail. Funkcja zwraca wartość true, jeżeli wysyłanie wiadomości się powiodło, false w przeciwnym wypadku. Wymaga trzech argumentów: pierwszym jest adres e-mail osoby do której wysyłamy wiadomość, drugi to temat wiadomości, a trzeci jej treść. Posiada również dwa argumenty opcjonalne: pierwszy to dodatkowe nagłówki oraz drugi to dodatkowe parametry dla poleceń linii komend.

 Slajd
 10



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

Ćwiczenie – wysyłania maila

- Używając skonfigurowanego serwera SMTP, wysłać maila na swoje konto. Użyć funkcji PHP `mail`.


 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Wykonać ćwiczenie.

Człowiek - najlepsza inwestycja



Rozwiązanie ćwiczenia



```
<?php
    $email = 'adres@domena.pl';
    $subject = 'Temat maila';
    $content = 'Hello Filip';
    try{
        mail($email, $subject, $content)
        echo 'Wysłano wiadomość email!';
    } catch(Exception $e) {
        echo $e->message;
    }
?>
```



W pierwszej kolejności definiujemy adres pod który chcemy wysłać maila, treść maila oraz jego temat.

Następnie próbujemy wysłać maila, jeżeli nam się uda otrzymujemy komunikat o wysłaniu, w przeciwnym wypadku treść błędu, który mówi nam o tym dlaczego się nie powiodło.







**PHP: Hypertext
Preprocessor**

Przykład obiektu klienta SMTP cz. 1

```

<?php
    $cfg = new stdClass();
    $cfg ->SMTP->SmtpServer = 'localhost';
    $cfg ->SMTP->SmtpPort = '25';
    $cfg ->SMTP->SmtpUser = 'me@localhost.com';
    $cfg ->SMTP->SmtpPass = 'aaaaa';
?>
    
```


 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Przykład zobrazuje nam całą komunikację pomiędzy serwerem SMTP, a klientem i wyświetli w postaci tablicy. Został skompresowany na potrzeby zobrazowania niskopoziomowego mechanizmu komunikacji z serwerem oraz uszczuplony o mechanizmy zabezpieczające aplikacje. Nie należy korzystać z niego jako z gotowej aplikacji. W pierwszej kolejności tworzymy plik konfiguracyjny dla naszego obiektu umieszczamy go w osobnym pliku i wypełniamy odpowiednimi danymi., które będą nam potrzebne do wysłania maila.

Użyta w przykładzie klasa stdClass jest podstawową klasą dostępną w PHP (podobnie jak Object w Javie). Jest obiektem odpowiednikiem tablicy, często używana do ustawiania parametrów konfiguracyjnych.

Np:




```

$book = new stdClass;
$book->title = "Harry Potter and the Prisoner of Azkaban";
$book->author = "J. K. Rowling";
$book->publisher = "Arthur A. Levine Books";
$book->amazon_link = "http://www.amazon.com/dp/0439136369/";
    
```

Następnie uzyskać można dostęp do odpowiednich pól poprzez: \$book->title
 Można również rzutować tablicę na obiekt stdClass, np:

```

$array = array(
    "title" => "Harry Potter and the Prisoner of Azkaban",
    
```

	<pre>"author" => "J. K. Rowling", "publisher" => "Arthur A. Levine Books", "amazon_link" => "http://www.amazon.com/dp/0439136369/"); \$books = (object) \$array;</pre> <p>Źródło: http://www.webmaster-source.com/2009/08/20/php-stdclass-storing-data-object-instead-array/</p>
<p>Slajd 13</p>	<div data-bbox="284 584 1374 1397">   <p>PHP: Hypertext Preprocessor</p> <h2>Przykład obiektu do klienta SMTP cz. 2</h2> <pre><?php class MailForm{ public static function getForm(){ \$form = '<form method="post" action=""><table> <tr><td>To: </td><td><input type="text" name="to" /></td></tr> <tr><td>From: </td><td><input type="text" name="from" /></td></tr> <tr><td>Subject: </td><td><input type="text" name="sub" /></td></tr> <tr><td>Message: </td><td><textarea name="message"></textarea></td></tr> <tr><td colspan="2"><input type="submit" value=" Send " /></td></tr> </table></form>'; return \$form; } } ?></pre>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p> </div> <p>Potrzebujemy obiekt formularza, który będzie przetwarzany na wiadomości e-mail. Powyższy obiekt posiada jedną metodę, która zwraca nam formularz HTML w postaci ciągu znaków PHP. Umieszczamy go w osobnym pliku.</p>

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
14



PHP: Hypertext
Preprocessor

Przykład obiektu klienta SMTP cz. 3

```
<?php  
class SMTPClient  
{
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część pierwsza klasy

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Slajd
15



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

```
function setConfig($config) {
    $this->SmtpServer = $config->SMTP->SmtpServer;
    $this->SmtpUser = base64_encode ($config->SMTP-
>SmtpUser);
    $this->SmtpPass = base64_encode ($config->SMTP-
>SmtpPass);
    if ($config->SMTP->SmtpPort == ""){
        $this->PortSMTP = 25;
    } else {
        $this->PortSMTP = $config->SMTP->SmtpPort;
    }
}

function setFrom($from) {
    $this->from = $from;
}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część druga klasy, procedura, która ustawia odpowiedni parametry i dane, potrzebne do wysłania naszego maila.

Slajd
16



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

```
function setTo($to) {
    $this->to = $to;
}

function setSubject($subject) {
    $this->subject = $subject;
}

function setMailContent($body) {
    $this->body = $body;
}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część druga klasy, procedura, która ustawia odpowiedni parametry i dane, potrzebne do wysłania naszego maila.

Człowiek - najlepsza inwestycja




KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
17



B2E
 BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

**PHP: Hypertext
Preprocessor**

```
function SendMail () {
    $talk = array();
    $HTTP_HOST = $_SERVER['HTTP_HOST'];
    try {
        if($this->SmtpServer == null || strlen($this->SmtpServer) == 0){
            throw new Exception('SmtpServer is missing, please use
            setConfig() method.');
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część trzecia klasy. Próbujemy otworzyć połączenie z serwerem SMTP za pomocą funkcji fsockopen, a jeżeli nam się powiedzie zaczynamy się z nim komunikować. Funkcja fwrite wysyła komendy z parametrami, a funkcja fgets pobiera odpowiedź.

Człowiek - najlepsza inwestycja

Slajd
18



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

```
if ($SMTPPIN = fsockopen ($this->SmtServer, $this->PortSMTP, $errno,
    $errstr, 10)) {
    fwrite($SMTPPIN, "EHLO ".$HTTP_HOST."\r\n");
    $talk["hello"] = fgets ($SMTPPIN, 1024 );
    fwrite($SMTPPIN, "auth login\r\n");
    $talk["res"]=fgets($SMTPPIN,1024);
    fwrite($SMTPPIN, $this->SmtUser."\r\n");
    $talk["user"]=fgets($SMTPPIN,1024);
    fwrite($SMTPPIN, $this->SmtPass."\r\n");
    $talk["pass"]=fgets($SMTPPIN,256);
    fwrite ($SMTPPIN, "MAIL FROM: <".$this->from.">\r\n");
    $talk["From"] = fgets ($SMTPPIN, 1024 );
    fwrite ($SMTPPIN, "RCPT TO: <".$this->to.">\r\n");
    $talk["To"] = fgets ($SMTPPIN, 1024);
    fwrite($SMTPPIN, "DATA\r\n");
    $talk["data"]=fgets($SMTPPIN,1024 );
    fwrite($SMTPPIN, "To: <".$this->to.">\r\nFrom: <".$this->from.">\r\nSubject: ".$this->subject."\r\n\r\n\r\n".$this->body."\r\n\r\n");
    $talk["send"]=fgets($SMTPPIN,256);
    fwrite ($SMTPPIN, "QUIT\r\n");
    fclose($SMTPPIN);
}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część trzecia klasy. Próbuje otworzyć połączenie z serwerem SMTP za pomocą funkcji fsockopen, a jeżeli nam się powiedzie zaczynamy się z nim komunikować. Funkcja fwrite wysyła komendy z parametrami, a funkcja fgets pobiera odpowiedź.

Slajd
19



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

```
else {
    $talk["error"] = "$errstr ($errno) \n";
}
catch (Exception $e) {
    $talk["exception"] = $e->getMessage()." \n";
}
return $talk;
}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część trzecia klasy. Próbuje otworzyć połączenie z serwerem SMTP za pomocą







Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY




	funkcji fsockopen, a jeżeli nam się powiedzie zaczynamy się z nim komunikować. Funkcja fwrite wysyła komendy z parametrami, a funkcja fgets pobiera odpowiedź.	
Slajd 20	<div><div></div><div>PHP: Hypertext Preprocessor</div></div> <div><div></div><div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div></div> <h3>Przykład obiektu klienta SMTP cz. 4</h3> <pre><?php require_once('config.php'); require_once('App.php'); \$app = new App(\$cfg); ?></pre>	<p>Implementujemy główne wywołanie naszego klienta SMTP. W przykładzie można ten kod umieścić w pliku index.php w katalogu naszej aplikacji.</p> <p>Importujemy plik konfiguracyjny oraz klasę naszej aplikacji, którą zaraz zaimplementujemy.</p> <p>Tworzymy jej egzemplarz i wywołujemy metodę inicjalizującą.</p>
Slajd 21	<div><div></div><div>PHP: Hypertext Preprocessor</div></div> <div><div></div><div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div></div> <h3>Przykład obiektu klienta SMTP cz. 5</h3> <pre><?php class App{ private \$config; private \$form; private \$data; public function __construct(\$config){ \$this->config = \$config; \$this->data = new stdClass(); \$this->init(); } }</pre>	
<p>Pierwsza część klasy. Rozpoczynamy implementację obiektu App. W przykładzie powyżej znajdują się zmienne wewnętrzne jak dane konfiguracyjne, formularz i dane do wysłania. Plik konfiguracyjny przesyłamy za pomocą konstruktora obiektu, który wywoływany jest w momencie jego tworzenia. Funkcja aplikacji mogła by być obiektem typu singleton, lecz ze względu na brak potrzeby, mogło by to tylko zaciemnić obraz przykładu.</p>		

Człowiek - najlepsza inwestycja

Slajd
22

**B2E**
BUSINESS TO EDUCATION

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP: Hypertext
Preprocessor


DAILY
GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część druga implementacji klasy App. Funkcja inicjalizująca aplikację. W pierwszej kolejności ładuje odpowiednie klasy, przekazane w argumencie jako tablica. Następnie pobiera formularz potrzebny do wysłania maila. Metoda sprawdza czy zostało coś wysłane metodą POST, jeżeli tak to pobiera te dane, przetwarza je i wysyła maila. W przeciwnym wypadku wyświetla formularz.

Slajd
23

**B2E**
BUSINESS TO EDUCATION

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

DAILY
GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część trzecia implementacji klasy App. Pierwsza metoda ładuje klasy podane w tablicy jako argumenty. Druga metoda wyświetla formularz, wcześniej wczytany do obiektu.

Człowiek - najlepsza inwestycja




KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
24



B2E
 BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

**PHP: Hypertext
Preprocessor**


```
private function sendMail(){

    $SMTPMail = new SMTPClient ($this->config->SMTP-
    >SmtpServer, $this->config->SMTP->SmtpPort, $this-
    >config->SMTP->SmtpUser, $this->config->SMTP->SmtpPass,
    $this->data->from, $this->data->to, $this->data-
    >subject, $this->data->body);

    $SMTPChat = $SMTPMail->SendMail();

    print_r($SMTPChat);

}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Część czwarta implementacji klasy App. Tworzymy egzemplarz wcześniej utworzonej klasy i wywołujemy jej metodę sendmail. Następnie wypisujemy tablicę z odpowiedziami od serwera SMTP.

Człowiek - najlepsza inwestycja



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

```
private function getPostData() {
    if($_SERVER["REQUEST_METHOD"] == "POST")
    {
        $this->data->to = $this->test_input($_POST['to']);
        $this->data->from = $this->test_input($_POST['from']);
        $this->data->subject = $this->test_input($_POST['sub']);
        $this->data->body = $this->test_input($_POST['message']);
    }
}

private function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
}
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Ostatnia część trzecia implementacji klasy App. Funkcja przetwarzająca dane z formularza. Plik zapisujemy jako np. App.php. Nareszcie możemy zobaczyć efekt działania naszej aplikacji, pod warunkiem, że serwer SMTP jest poprawnie skonfigurowany oraz wprowadziliśmy poprawne dane w pliku konfiguracyjnym i formularzu.

Człowiek - najlepsza inwestycja





KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
 26


B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY


PHP: Hypertext
 Preprocessor

Przykład użycia PHPMailera cz. 1

```

<?php
require_once('../class.phpmailer.php');


$mail = new PHPMailer();
$body =
file_get_contents('szablon_wiadomosci.html');
    
```



 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

W pierwszej kolejności dołączamy klasę phpmailera do naszego skryptu, używamy require_once – jeśli plik był już wcześniej dołączony, nie dołączamy go ponownie. Następnie tworzymy egzemplarz klasy i pobieramy szablon html, bądź txt naszego maila w zależności od potrzeb.

 Slajd
 27


B2E
 BUSINESS TO EDUCATION



 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor

Przykład użycia PHPMailera cz. 2

```

$mail->IsSMTP();
$mail->Host = "mail.nazwa_domeny.pl";
$mail->SMTPAuth = true;
$mail->Port = 25;
$mail->Username = "mail.nazwa_domeny.pl";
$mail->Password = "haslo";
    
```


 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Ustalamy protokół na SMTP, włączamy autoryzację oraz konfigurujemy podstawowe parametry dla konta SMTP, takie jak host, port, nazwę użytkownika i hasło.

Człowiek - najlepsza inwestycja



Przykład użycia PHPMailera cz. 3

```
$mail->From('filip@nazwa_domeny.pl');
$mail->AddReplyTo('ewa@nazwa_domeny.pl', 'Ewa');
$mail->Subject    = "Temat";
$mail->MsgHTML($body);
```



W tym miejscu ustalamy adres nadawcy, adres pod który należy wysłać odpowiedź, temat wiadomości oraz treść wiadomości, w tym przypadku wiadomość w formacie HTML.




Przykład użycia PHPMailera cz. 4

```
$address = "adres_odbiiorcy";
// dodawanie adresów i załączników
$mail->AddAddress($address, "Ewa L");
$mail->AddAttachment("sciezka_do_zalacznika");
```




Możemy dodawać wiele adresów, bądź załączników do wiadomości e-mail. Należy jednak pamiętać, że wielkość załącznika jest konfigurowalna po stronie serwera, co może spowodować niemożliwość przyjęcia go przez serwer SMTP. Dodatkowo podczas tworzenia skryptu walidującego rozmiar załączanego pliku, należy zwrócić uwagę na fakt, że ze względu na specyfikę kodowania załączników w mailach (base64) jego rozmiar rośnie o ok. 30%.

Slajd
30



B2E
 BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład użycia PHPMailera cz. 5

```

if(!$mail->Send()) {
    echo "Błąd phpmailera: " . $mail->ErrorInfo;
} else {
    echo "Wysłano!";
}
?>
```




PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Po odpowiednim skonfigurowaniu naszego obiektu próbujemy wysłać wiadomość, w przypadku niepowodzenia otrzymuje błąd zdefiniowany w Klasie PHPMailer, w przeciwnym razie otrzymujemy komunikat o wysłaniu wiadomości.

<p>Slajd 31</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenie praktyczne</h3> <p>Przygotuj i skonfiguruj skrypt, który wyśle wiadomość email przy pomocy funkcji „mail”.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>mail.php</p>
<p>Slajd 32</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenie praktyczne</h3> <p>Przygotuj i skonfiguruj skrypt, który wyśle wiadomość email przy pomocy omówionej w lekcji klasy SMTPClient.php</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>smtp/index.php smtp/App.php smtp/config.php smtp/MailForm.php smtp/SMTPClient.php</p>
<p>Slajd 33</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <h3>Ćwiczenie praktyczne</h3> <p>Pobierz ze strony http://goo.gl/Jov3T5 i skonfiguruj framework phpMailer. Na podstawie istniejącej specyfikacji lub przykładów, przygotuj skrypt, który wyśle wiadomość email w postaci pliku HTML.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>phpMailer/index.php phpMailer/contents.html</p>


Człowiek - najlepsza inwestycja


Slajd
 34


B2E
 BUSINESS TO EDUCATION
 
SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY


PHP: Hypertext
Preprocessor

Dziękujemy za uwagę



Człowiek - najlepsza inwestycja


Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

10.7.3 Ćwiczenia

Przykłady i ćwiczenia zawarte są na slajdach tej lekcji.

10.7.4 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie nauczą się jak działa protokół SMTP oraz będą potrafili wykorzystać go w swoich skryptach. Nauczą się również jak zaimplementować obiektowo małą aplikację, w tym przypadku prostego klienta do wysyłania poczty elektronicznej.



10.8 Lekcja 8 - MIME - rozszerzenia multimedialne (Struktura nagłówek MIME, Kodowanie Base64 i Quoted-Printable, Wiadomości wieloczęściowe, Obsługa poczty multimedialnej w PHP)

10.8.1 Cel lekcji

Celem lekcji jest przedstawienie protokołu MIME oraz jego rozszerzenia multimedialnego. Uczniowie poznają kodowania Base64 i Quoted-Printable. Nauczą się wysyłać pocztę multimedialną oraz poznają strukturę wiadomości wieloczęściowych.


Człowiek - najlepsza inwestycja


10.8.2 Treść - slajdy z opisem

<p>Slajd 1</p>	 <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany Lekcja 8 MIME - rozszerzenia multimedialne</p> <p><small>Człowiek - najlepsza inwestycja</small></p> <p><small>KAPITAŁ LUDZKI</small></p> <p><small>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</small></p>	<p>Słowa wstępu.</p>
<p>Slajd 2</p>	 <p>MIME</p> <ul style="list-style-type: none"> • Co to jest? • Wprowadzenie <p><small>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</small></p>	<p>MIME (ang. Multipurpose Internet Mail Extensions) jest to standard do przesyłania poczty elektronicznej, który definiuje budowę komunikatu. Składa się z nagłówek i treści. Standardowo mail wysyła treść jako tekst zakodowany w 7-bitowym kodzie ASCII. MIME posiada kodowanie 8-bitowe, dlatego zwiększa objętość maila. Źródło zdjęcia: http://www.costelloentertainments.co.uk/3770/174/artists/street-performers---walkabout-entertainers.aspx</p>

Człowiek - najlepsza inwestycja

Slajd
3


 **B2E**
BUSINESS TO EDUCATION

 **SZCZECIŃSKI PARK**
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Typ MIME

- Co to jest?
- Do czego służy?


 **DAILY**
G R O U P

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Typ MIME (nazwany od standardu MIME) – internet media type lub Content-Type (po nazwie nagłówka kilku protokołów, którego wartość jest tego typu) jest to dwuczęściowy identyfikator typu pliku w sieci, pierwsza część opisuje jego typ, a druga konkretny podtyp. Np. Image/gif. Image mówi, że jest to typ obrazkowy o podtypie gif, który jest zdefiniowany jako standard dla obrazów o tym rozszerzeniu. Początkowo typ ten był używany w protokole SMTP, a później zaczęto z niego korzystać również przy protokole HTTP. Służy do określania standardu przesyłanego pliku w sieci Internet.

Slajd
4

 **B2E**
BUSINESS TO EDUCATION

 **SZCZECIŃSKI PARK**
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Struktura nagłówków MIME










MIME-version: 1.0

 **DAILY**
G R O U P

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Struktura nagłówków MIME jest identyczna ze strukturą nagłówków HTTP. Powyższy nagłówek określa wersję stosowanego standardu, w tym przypadku wersja 1.0.

Człowiek - najlepsza inwestycja

<p>Slajd 5</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Nagłówki MIME</h2> <p>Wymagane:</p> <ul style="list-style-type: none"> • MIME-version • Content-Type <p>Opcjonalne:</p> <ul style="list-style-type: none"> • Content-Disposition • Content-Transfer-Encoding • Content-ID • Content-Description  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>W najnowszym standardzie wymagane nagłówki to wersja MIME oraz typ przesyłanej treści. Opcjonalne: Kodowanie MIME zawiera przeważnie jeden z dwóch parametrów base64 lub quoted-printable. Jest jeszcze jeden mniej popularny za to najprostszy: 8bit, nie wymagający żadnej konwersji. Nagłówek Content-ID wyznacza wiadomość właściwą, a Content-Description komentarz do wysyłanego komunikatu.</p>
<p>Slajd 6</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Składnia nagłówka Content-Type</h2> <ol style="list-style-type: none"> 1. Nazwa typu mediów 2. Nazwa podtypu mediów 3. Wymagane parametry (nie każdy typ tego wymaga) 4. Opcjonalne parametry (nie każdy typ tego wymaga)  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Nagłówek Content-Type wymaga typu i podtypu mediów. Niektóre z nich wymagają parametrów, które podajemy w następnej kolejności. Na ostatnim miejscu umieszczamy parametry opcjonalne.</p>
<p>Slajd 7</p>	  <p>PHP: Hypertext Preprocessor</p> <h2>Możliwe wartości Content-Type cz. 1</h2> <ul style="list-style-type: none"> • text • image • audio • video • application • multipart • message  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Część odpowiedzialna za typ, nagłówek Content-Type może przyjmować jedną z powyższych wartości. Czasami dopuszczalne są dwie ostatnie opcje multipart oraz message. Przez użycie typu multipart, MIME pozwala, by wiadomość posiadała wiele części, z których każda może mieć określony swój własny typ MIME.</p>

Człowiek - najlepsza inwestycja

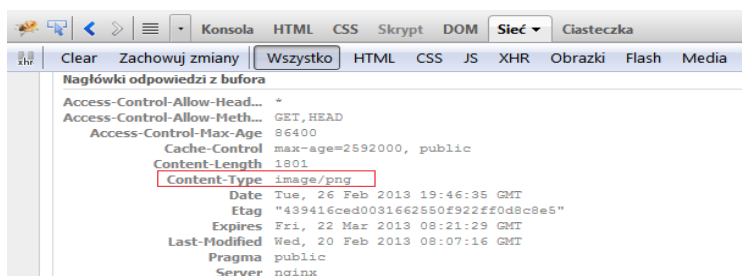
Slajd
8



PHP: Hypertext
Preprocessor

Ćwiczenie

- Otwórz dowolną stronę www i przy pomocy narzędzia firebug znajdź content-type dla różnych typów plików, np. grafiki, plików tekstowych itd.



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

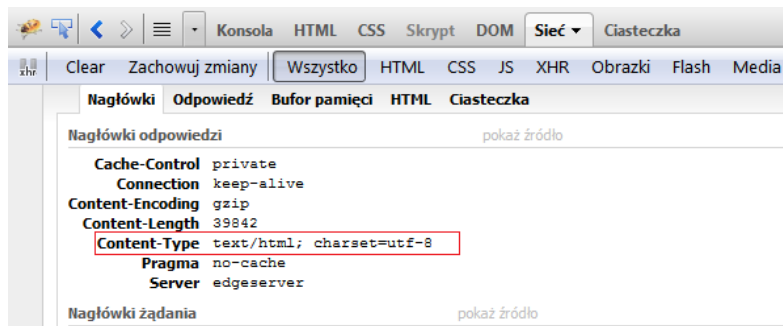


Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego



Ćwiczenie cd

Przykład dla pliku tekstowego html:





Możliwe wartości Content-Type cz. 2

- Podtypów MIME jest bardzo dużo, podobnie jak możliwych rozszerzeń plików, cały wykaz znajduje się w dokumentacji:

<http://goo.gl/P5iCtw>




Slajd
11

PHP: Hypertext Preprocessor

Możliwe wartości Content-Type cz. 3



- charset="us-ascii"



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Powyższy przykład obrazuje zastosowanie parametrów w nagłówku Content-Type, w tym przypadku jest to kodowanie znaków.


Slajd
12

PHP: Hypertext Preprocessor

Kodowanie Base64

- Co to jest?
- Do czego służy?
- Jak działa?



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Kodowanie base64 to rodzaj kodowania transportowego, oznacza to, że służy do bezpiecznego przesyłania danych. Zabezpiecza dane szczególnie ze względu na uszkodzenia spowodowane niemożnością przesyłania 8 bitowych danych przez starsze maszyny, które przysyłały znaki 7-bitowe. Kodowanie to dzieli dowolne 64 znaki na grupy po trzy bajty. Bajt posiada 8 bitów, dlatego grupa ma 24 bity. Co po podzieleniu na cztery skutkuje 6 bitowymi jednostkami. Na sześciu bitach możemy zapisać 64 różne znaki, czyli każda jednostka może przyjąć 64 wartości, stąd też nazwa kodowania. Wszystkim takim jednostkom przypisywane są znaki, na podstawie początkowo ustalonych liczb. Jeśli rozmiar wejściowego ciągu bajtów nie jest podzielny przez 3, stosowane jest dopełnienie, aż do jego podzielności przez 4.

Człowiek - najlepsza inwestycja

<p>Slajd 13</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Przykład kodowania Base64</h3> <ul style="list-style-type: none"> · Hello World! <p>W kodowaniu Base64:</p> <ul style="list-style-type: none"> · SGVsbG8gV29ybGQh  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Powyższy przykład obrazuje wynik kodowania ciągu znaków przy pomocy kodowania Base64.</p>
<p>Slajd 14</p>	  <p>PHP: Hypertext Preprocessor</p> <h3>Zastosowania kodowania Base64</h3> <ul style="list-style-type: none"> · Przesyłanie załączników binarnych w mailach · Kodowanie haseł przy wysyłaniu protokołem SMTP  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>Przesyłanie załączników binarnych w mailach – np. grafiki, muzyki mp3 itp</p>

Człowiek - najlepsza inwestycja





KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
15

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY




PHP: Hypertext
Preprocessor

Ćwiczenie

Otwórz plik mail.eml zawierający przykładowy e-mail z programu pocztowego. Znajdź fragment w którym znajduje się zakodowany w Base64 załącznik.



```
--_004_4EBD81F4E859284EBDC02845B7EF01A2453FD6A0AM2PRD0311MB421_
Content-Type: application/octet-stream; name="Week meeting 06022013.pdf"
Content-Description: Week meeting 06022013.pdf
Content-Disposition: attachment; filename="Week meeting 06022013.pdf";
size=85382; creation-date="Mon, 11 Feb 2013 06:13:22 GMT";
modification-date="Mon, 11 Feb 2013 06:13:22 GMT"
Content-Transfer-Encoding: base64

JVBERi0xLjQKJelz9MKMyAwIG9iago8PAovUHJvZHVjZXIgaKFBERi1YQ2hhbmdlIDQuMC4yMDEu
MCBcKfDpbmRvd3MgU2V2ZW4gVWx0aW1hdGUgeDY0IFNlcnZpY2UgUGFjayAxIFwoQnVpbGQgNzYw
MVVpXCkpCi9DcmVndG1vbkrhdGUgKEQ6MjAxMzAwMTEwODEwMDEwMDEwMDEwMDEwMDEwMDEw
NSAwIG9iago8PAovVHlwZSAvWE9iamVjdAovU3VidHlwZSAvSW1hZ2UKL1dpZHRoIDk2OQovSGVp
Z2h0IDYzMQovQm10c1BlckNvbXBvbmVudCA4Ci9Db2xvc1NwYWN1IC9EZXBpY2VSR0IKL0xlbmd0
aCA2ODgzMAovRmlsdGVyIFsvRm9hdGVEZWNvZGUgL0R0DVER1Y29kZV0KFj4Kc3RyZWFrCnJa7P0H
VFTLti+ML0RFQUARKnWqIFkkZ1pEstgSJCdBRUBjyq2ioOQgObRIoIKTYWYwSASAKInnnOmw+mv2
Pufcvfc5997z3rv/8X3jjT+0hay1ZlfVnFU1S2/OmLWN/4mfBi6rKqooAkREAEBE+AfqgwB54ML5
8yTnz10gISG5ePEC6SVq8ktkZJfoqa5SUjMzQF1YGZiYrrPd5rh+k5eViYLTjIv3joCwsDCEQ0JG
XFD6tpCw4GkhRBcvXrxEdomOnJx08AbTDcH/SR/8N+DKhTPuxO3ERDeBM1eIiK8Q4ZsACKGd54h+
```




PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Slajd
16


SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY



PHP: Hypertext
Preprocessor

Kodowanie Quoted-Printable



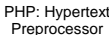



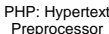

- Co to jest?
- Wprowadzenie



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Quoted-Printable – jest kolejnym kodowaniem transportowym zgodnym ze standardem MIME. Polega ono na tym, że pozostawia wszystkie bajty mniejsze od 128 (7-bitów), które nie są znakami sterującymi ASCII (oprócz spacji), a resztę zamienia na zakodowany szesnastkowo napis ASCII poprzedzony znakiem równości. Znak równości też jest kodowany, w celu uniknięcia wieloznaczności jako =3D. Kodowanie to jest głównie wykorzystywane w poczcie elektronicznej do kodowania przesyłanych wiadomości, które zawierają znaki narodowe z poza ASCII.

Człowiek - najlepsza inwestycja

		Nie nadaje się do przesyłania plików binarnych ze względu na jego narzut około 225%, gdzie np. Base64 posiada tylko 33% narzutu.
Slajd 17	   <p>Przykład kodowania Quoted-Printable</p> <ul style="list-style-type: none"> • Hello= world! • Hello=3D world!  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	Przykład zakodowania hello world w kodowaniu Quoted – Printable, znaki alfabetu oraz znak ‘!’, mają kod ASCII poniżej 128, dlatego dla lepszego zobrazowania, że coś się dzieje, dodano znak równości.
Slajd 18	   <p>Wiadomości wieloczęściowe</p> <ul style="list-style-type: none"> • Jak je definiować?  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	Wiadomości wieloczęściowe określone jako typ multipart w nagłówku Content-Type. Dodatkowo w tym samym nagłówku muszą posiadać parametr określający separator ‘boundary’, którego nie wolno umieszczać w treści wiadomości. Służy on do oddzielania od siebie wiadomości.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
19



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wiadomości wieloczęściowej cz.1

From: zuraw@domena.pl

To: czapla@domena.pl Subject: =?iso-8859 -
2?Q?Czaplo=2C_czy_um=F3wiesz_si=EA_ze_mn=B1=3F?=
MIME Version: 1.0

MIME Version: 1.0

Content-Type: multipart/mixed;
boundary="xxxToJestSeparator0000xxx"

This is a message with multiple parts in MIME format.

--xxxToJestSeparator0000xxx

Content-Type: multipart/alternative;
boundary="xxxToJestSeparatorZagniezdzony1111xxx"



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Na początku określamy adresata, odbiorcę oraz wersję MIME z której skorzystamy. Typ wiadomości ustawiony jest na multipart oraz podtyp na mixed (mieszany), który określa, że części wiadomości mogą być różnych typów. W sekcji typu umieszczamy parametr boundary, który definiuje separator w tym przypadku główny, gdyż wiadomościach można zagnieżdżać również wiadomości.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wiadomości wieloczęściowej cz. 2

```
--xxxToJestSeparatorZagniezdzony1111xxx
```

```
Content-Type: text/plain; charset="iso-8859-2"
```

```
Content-Transfer-Encoding: quoted-printable
```

```
To jest tre=B6=E6 wiadomo=B6ci.
```

```
--xxxToJestSeparatorZagniezdzony1111xxx
```

```
Content-Type: text/html; charset="iso-8859-2"
```

```
Content-Transfer-Encoding: quoted-printable
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wiadomości wieloczęściowej cz. 3

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0  
Transitional//EN">
```

```
<HTML><HEAD>
```

```
<META http-equiv=3DContent-Type content=3D"text/html;  
charset=3Diso-8859-2"></HEAD>
```

```
<BODY><FONT face=3DArial size=3D2>To jest tre=B6=E6  
wiadomo=B6ci.</FONT></BODY></HTML>
```

```
--xxxToJestSeparatorZagniezdzony1111xxx--
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Tutaj widzimy przykład zagnieżdżonej wiadomości typu HTML.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
22



PHP: Hypertext
Preprocessor

Przykład wiadomości wieloczęściowej cz. 4

```
--xxxToJestSeparator0000xxx
Content-Type: image/gif; name="obrazek.gif"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="obrazek.gif"
PGh0bWw+CiAgPGhlYWQ+CiAgPC9oZWFKPgogIDxib2R5PgogICAgP
HA+VGhpcyBpcyB0aGUg
Ym9keSBvZiB0aGUgbWVzc2FnZS48L3A+CiAgPC9ib2R5Pgo8L2h0b
Ww+Cg==
--xxxToJestSeparator0000xxx--
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Jak widać w powyższym przykładzie po raz pierwszy użyto kodowania base64, zamiast quoted-printable. Użyto go do zakodowania obrazu typu gif.
Ćwiczenie – zdekoduj zawartość pliku decodeBase64.txt do postaci pliku graficznego gif.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Slajd
23



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wysyłania wiadomości HTML w PHP cz. 1

```
<?php  
  
$to = 'stefan@domena.pl' . ', ';  
$to .= 'marian@domena.pl';  
$subject = 'Birthday Reminders for August';
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Na początek definiujemy odbiorców oraz ustalamy temat maila, ponieważ do przesłania wiadomości multimedialnej w tym przypadku w formacie html, użyjemy funkcji mail w PHP.

Slajd
24



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wysyłania wiadomości HTML w PHP cz. 2

```
$message = '  
  <html>  
  <head><title>Tytuł</title></head>  
  <body>  
    <p>Hello world!</p>  
  </body>  
</html>  
';
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Jako treść wiadomości definiujemy sobie kod w formacie html.

Człowiek - najlepsza inwestycja




KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI


UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
25



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wysyłania wiadomości HTML w PHP cz. 3

```

$headers = 'MIME-Version: 1.0' . "\r\n";
$headers .= 'Content-type: text/html; charset=utf8' .
"\r\n";
$headers .= 'From: Marek<marek@domena.pl>' . "\r\n";
    
```




DAILY
G R O U P


PHP: Hypertext Preprocessor - poziom średnio zaawansowany

W nagłówkach ustalamy wersję MIME, wymagany w tym przypadku typ text/html z kodowaniem w utf8 oraz nadawcę maila.

Slajd
26



B2E
BUSINESS TO EDUCATION




SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Przykład wysyłania wiadomości HTML w PHP cz. 4

```

if (mail($to, $subject, $message, $headers))
    echo 'Wysłano wiadomość!';
else
    echo 'Wysyłanie wiadomości nie powiodło się!';
?>
    
```







DAILY
G R O U P

PHP: Hypertext Preprocessor - poziom średnio zaawansowany



Na końcu wysyłamy wiadomość.

Człowiek - najlepsza inwestycja

<p>Slajd 27</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <p>Ćwiczenie praktyczne</p> <p>Znajdź dowolny, nieduży plik graficzny GIF i przygotuj skrypt, który zakoduje grafikę do ciągu znaków Base64.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>base64Encode.php</p>
<p>Slajd 28</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <p>Ćwiczenie praktyczne</p> <p>Przygotuj skrypt, który zdekoduje podany w pliku decodeBase64.txt ciąg znaków do pliku graficznego GIF. Skrypt powinien wyświetlić grafikę w przeglądarce lub uruchomić pobieranie pliku.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>decodeToGif.php</p>
<p>Slajd 29</p>	  <p>PHP: Hypertext Preprocessor</p> <hr/> <p>Ćwiczenie praktyczne</p> <p>Przygotuj skrypt, który zakoduje w Base64 dowolny plik binarny (grafika, muzyka) o rozmiarze ok 1MB. Wygenerowany w Base64 ciąg znaków zapisz do pliku TXT. Porównaj różnicę w rozmiarach obu plików.</p>  <p>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</p>	<p>decodeToGif.php</p>


Człowiek - najlepsza inwestycja

Slajd
30



B2E
 BUSINESS TO EDUCATION
 
SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor


Dziękujemy za uwagę



Człowiek - najlepsza inwestycja



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

10.8.3 Ćwiczenia

Jako ćwiczenia do lekcji, należy wykonać razem z uczniami przykłady znajdujące się na slajdach.

10.8.4 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie będą potrafili wysyłać pocztę multimedialną np. w postaci plików html. Uczeń powinien poznać możliwe rozszerzenia plików, podstawowe nagłówki MIME oraz podstawowe kodowania transportowe. Umieć rozróżnić wiadomość wieloczęściową, od jednoczęściową.










10.9 Lekcja 9 - UML 2.1 (Diagramy czynności, Projektowanie złożonych aplikacji)

10.9.1 Cel lekcji


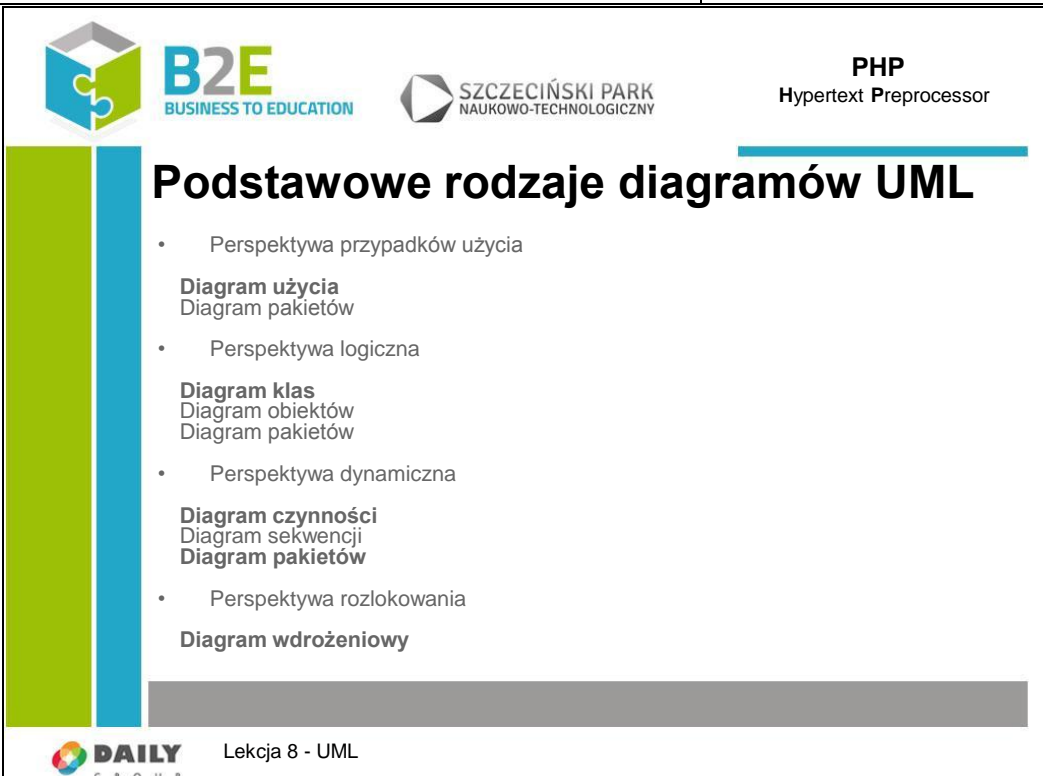
Celem lekcji jest zapoznanie ucznia z językiem do projektowania UML. Nauczenie go na czym polegają podstawowe diagramy oraz jak zaprojektować złożone aplikacje.

Człowiek - najlepsza inwestycja

10.9.2 Treść - slajdy z opisem

Slajd 1	<div><div></div><div><h1>PHP</h1><h2>Hypertext Preprocessor</h2></div><div><p>Lekcja 9</p><p>UML 2.1</p><p><small>Człowiek - najlepsza inwestycja</small></p><p><small>KAPITAŁ LUDZKI</small></p><p><small>UNIA EUROPEJSKA</small></p><p><small>Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego</small></p></div><div> PHP Hypertext Preprocessor Lekcja 8 UML 2.1</div></div>	<p>UML jest notacją za pomocą której możliwe jest graficzne modelowanie oprogramowania. Notacja to sposób zapisu symboli, liter i znaków za pomocą której można wyrazić treść. Treścią którą będziemy omawiać jest budowa oprogramowania.</p>
Slajd 2	<div><div></div><div><h1>Czym jest UML?</h1><ul style="list-style-type: none">• Unified Modeling Language• Modelowanie• Historia UML</div><div></div><div> Lekcja 8 - UML</div></div>	<p>Rozwinięciem skrótu UML jest Unified Modeling Language czyli.... Zunifikowany Język Modelowania.</p> <p>Czym jest modelowanie? Opis struktur, zachowań, interakcji, zjawisk, układu, budowy za pomocą narzędzi (notacji, diagramów..) pozwalających zobrazowanie budowy układu, sprawdzenie zachowania badanego przedmiotu. Przykładem modelu jest np. model układu słonecznego.</p> <p>UML jest notacją za pomocą której możliwe jest zamodelowanie różnego rodzaju systemy. Bardzo często wykorzystywany jest do zamodelowania tworzonego</p>

Człowiek - najlepsza inwestycja

	<p>oprogramowania. Pozwala na zaprojektowanie budowy oprogramowania z wykorzystaniem podejścia obiektowego, jego modułów, integracji z użytkownikiem, z systemami zewnętrznymi.</p> <p>Twórcami notacji UML są G. Booch i I. Jacobson oraz J. Rumbaugh. Notacja powstała w latach 90 i jest rozwijana do dzisiaj – obecna wersja notacji to 2.2. UML jest standardem rozwijanym przez organizację OMG (Object Management Group).</p>	
Slajd 3		<ul style="list-style-type: none"> • przypadków użycia - definiuje funkcjonalność systemu, • dynamiczna - prezentuje dynamikę systemu, w jaki sposób zachowuje się system, • logiczna - dokumentuje statyczną część systemu • komponentów - grupuje powyższe perspektywy w logicznie powiązane komponenty • rozlokowania - specyfikuje sprzęt niezbędny do funkcjonowania określonych komponentów
Slajd 4		

Człowiek - najlepsza inwestycja


Aktualnie notacja składa się z około 16 rodzajów diagramów za pomocą których możliwe jest modelowanie statycznych i dynamicznych obszarów budowy i działania systemów. Diagramy pozwalają na zaprojektowanie budowy oprogramowania, baz danych, zaprojektowanie algorytmów, określenie modułów oprogramowania, rozmieszczenie modułów oprogramowania.


Diagramy można podzielić na statyczne - przedstawiające budowę systemu, oraz dynamiczne - przedstawiające np. komunikację pomiędzy elementami statycznymi

Na tej lekcji omówione zostaną podstawowe i najbardziej potrzebne rodzaje diagramów:

- Diagram użycia
- Diagram klas
- Diagram czynności
- Diagram pakietów
- Diagram wdrożeniowy

Slajd
5

**B2E**
BUSINESS TO EDUCATION

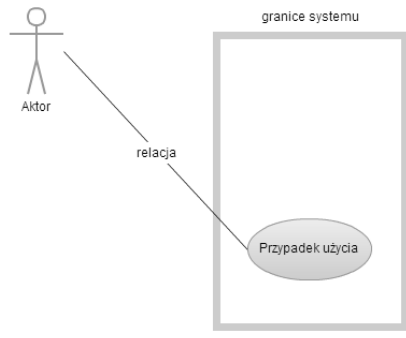
SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor


Diagram przypadków użycia

W skład diagramu wchodzi następujące elementy:

- aktor - wykonawca akcji, funkcji
- przypadek użycia - akcja, funkcja
- granice systemu
- relacje (związki)



The diagram illustrates a use case diagram. On the left, a stick figure labeled 'Aktor' is connected by a line labeled 'relacja' to an oval labeled 'Przypadek użycia'. This oval is contained within a rectangular box labeled 'granice systemu'.

DAILY
GROUP

Lekcja 8 - UML

Diagram przypadków użycia (DPU) pozwala na zaprezentowanie funkcjonalności projektowanego systemu, jego aktorów oraz komunikację między aktorami a przypadkami użycia. Pozwala także na zaprezentowanie relacji pomiędzy aktorami i przypadkami użycia.

Diagramy nazywane w języku ang. use case są stosowane do dokumentowania wymagań wobec projektowanego systemu. Projektant ma możliwość dekompozycji wymagania na podwymagania i w procesie iteracyjnego projektowania systemu na stopniowe ich uszczegóławianie. Odpowiednio opisany zestaw przypadków użycia może stanowić dokumentację wymagań która jest podstawą do projektowania i implementacji systemu a finalnie stanowi materiał wejściowy potrzebny przy testach aplikacji.

•Przypadek użycia - konkretna funkcjonalność, akcja którą system może wykonać w interakcji z aktorem. Przypadek użycia powinien mieć zwięzłą i jednoznaczną nazwę np. zadzwonić

•Aktor - użytkownik systemu, klient, inny system korzystający z projektowanego

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

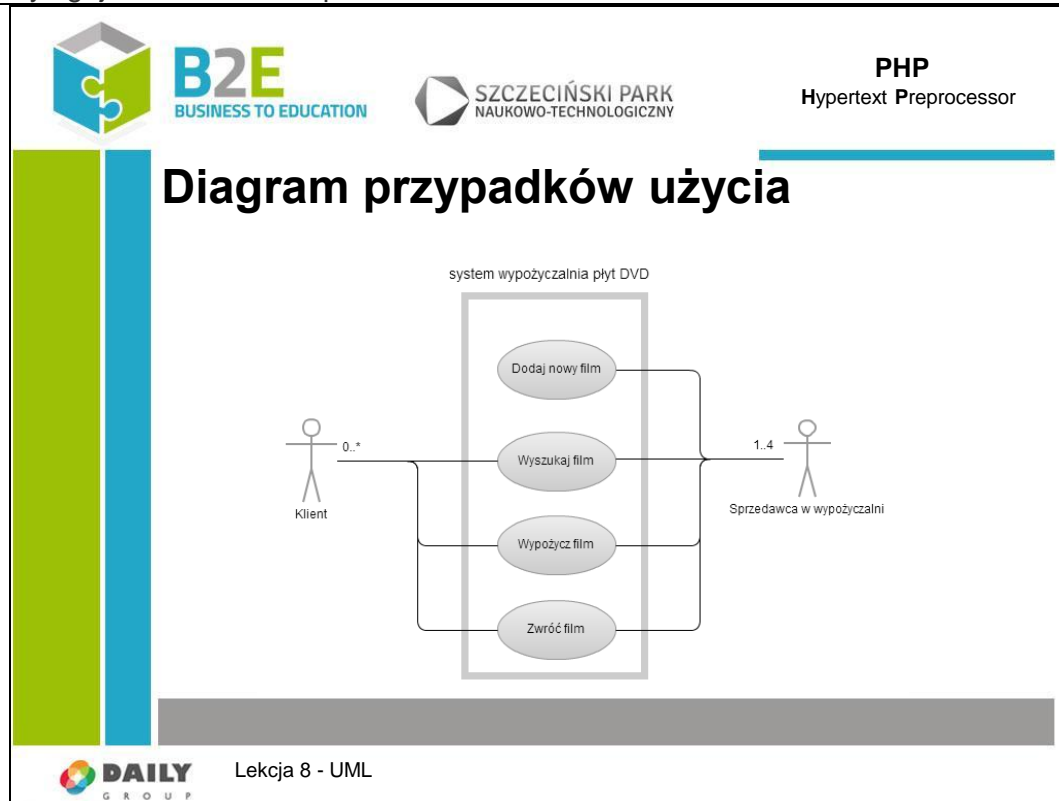


przez nas systemu. Aktor określany jest rzeczownikiem. Aktor reprezentuje typ użytkownika systemu, użytkownik może korzystać z wielu przypadków użycia.

- Relacja - każdy aktor na diagramie przypadków użycia powinien być związany przynajmniej z jednym przypadkiem użycia. Każdy przypadek użycia powinien być użytkowany przez co najmniej jednego aktora.
- Granice systemu - prezentują granice projektowanych systemów (na diagramie mogą wystąpić 2 lub więcej systemów których przypadki użycia należy ze sobą powiązać)

Przykłady: Przycisk dzwoni w telefonie, start w pralce, włącz silnik w samochodzie, wyloguj na facebook-u itp.

Slajd
6



Na rysunku przedstawiono przykładowy diagram przypadków użycia systemu zarządzającego wypożyczalnią płyt DVD (np. osiedlowa wypożyczalnia). Projektant założył bardzo prostą funkcjonalność:

Klient ma możliwość wyszukiwania filmów, wypożyczenia filmu, zwrócenia filmu
Sprzedawca ma także dostęp wyszukiwania filmów ich wypożyczenia i zwracania, ale dodatkowo może dodać nową pozycję w wypożyczalni.

Należy zauważyć że funkcjonalności są bardzo ogólne i w dalszym etapie projektowania należy je uszczegółowić - np. funkcja wypożycz film powinna sprawdzać czy dana płyta jest dostępna (nie jest wypożyczona), przydadzą się szczegóły każdej z funkcji zwłaszcza pola służące np. do wyszukiwania, zakres danych wyświetlanych jako rezultat itp.



Diagram przypadków użycia

Ćwiczenie:

Przygotuj szczegółowy diagram przypadków użycia dla biblioteki osiedlowej.

Zapewnij kompletność funkcjonalności związanej z obsługą wypożyczeń i zwrotów a także z zarządzaniem zbiorami bibliotecznymi i rezerwacjami.

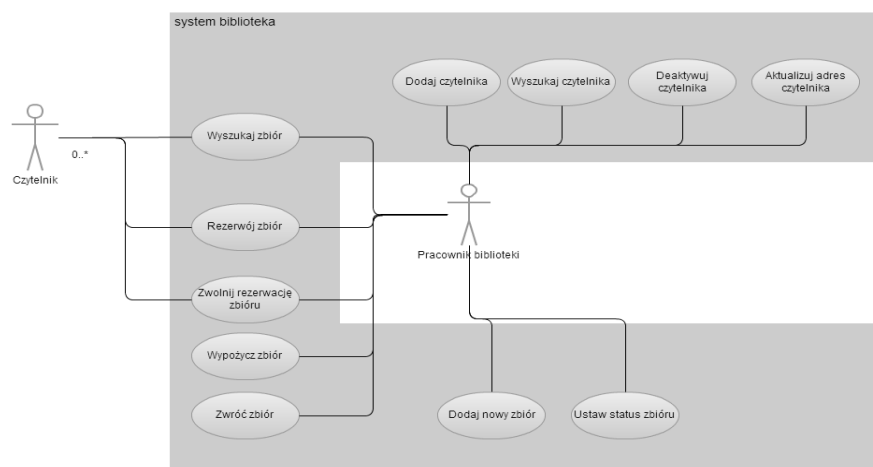


Zadanie można wykonać na kartkach.

Opcjonalnie można użyć istniejących darmowych narzędzi jak:
<http://www.umlet.com/>



Diagram przypadków użycia



Założono obsługę obszarów
Wyszukania i Rezerwacji (dostępne dla czytelnika)
Wypożyczenia i zwrotu – dla pracownika
Dodania zwrotu i edycji statusu zbioru
Zarządzania użytkownikiem (dodanie, wyszukanie, deaktywację, aktualizację).

Slajd
9

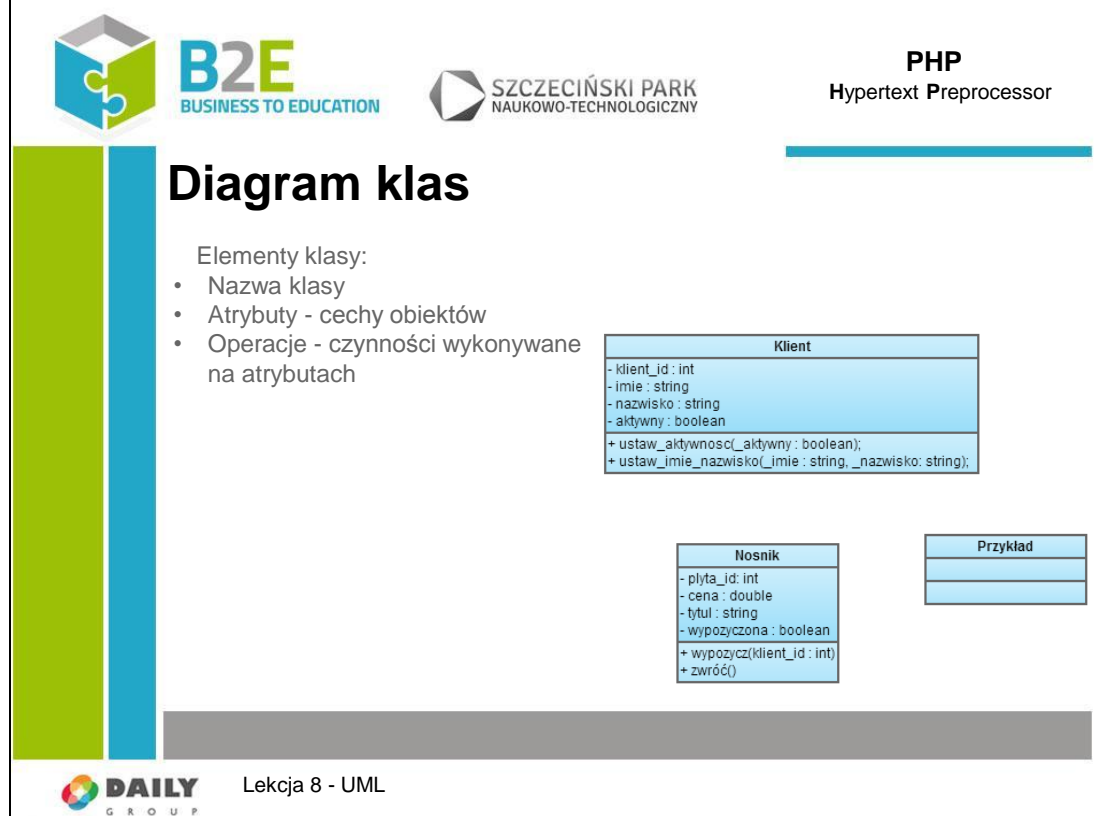


Diagram klas jest obok diagramów przypadków użycia najczęściej stosowanym diagramem w notacji UML.

Diagram klas najlepiej odwzorowuje programowanie obiektowe. Klasa w oprogramowaniu jest definicją obiektów. Obiekt zawiera wycinek modelowanej przez oprogramowanie rzeczywistości, jest czarną skrzynką która posiada atrybuty (dane) oraz metody (funkcje) które działają na atrybutach. Obiekt może udostępniać określone metody na zewnątrz dla innych obiektów - są to metody publiczne (w programowaniu obiektowym zadeklarowane jako public na diagramie oznaczone przez +). Za pomocą udostępnionych metod świat zewnętrzny może korzystać z funkcjonalności obiektu.

Widoczność atrybutów i metod oznaczana jest symbolami:

"-" - prywatne, niewidoczne na zewnątrz obiektu, dostępne tylko dla kodu obiektu

"+" - publiczne, widoczne na zewnątrz obiektu,

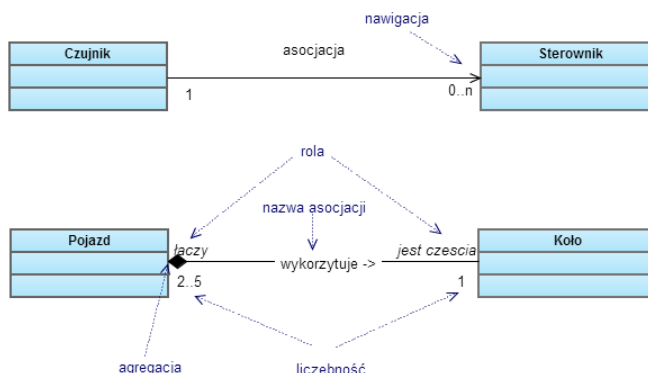
"#" - chronione, prywatne i dodatkowo widoczne dla klas dziedziczących

"~" - pakietowe, widoczne w pakiecie do którego należy klasa



Diagram klas

Asocjacja

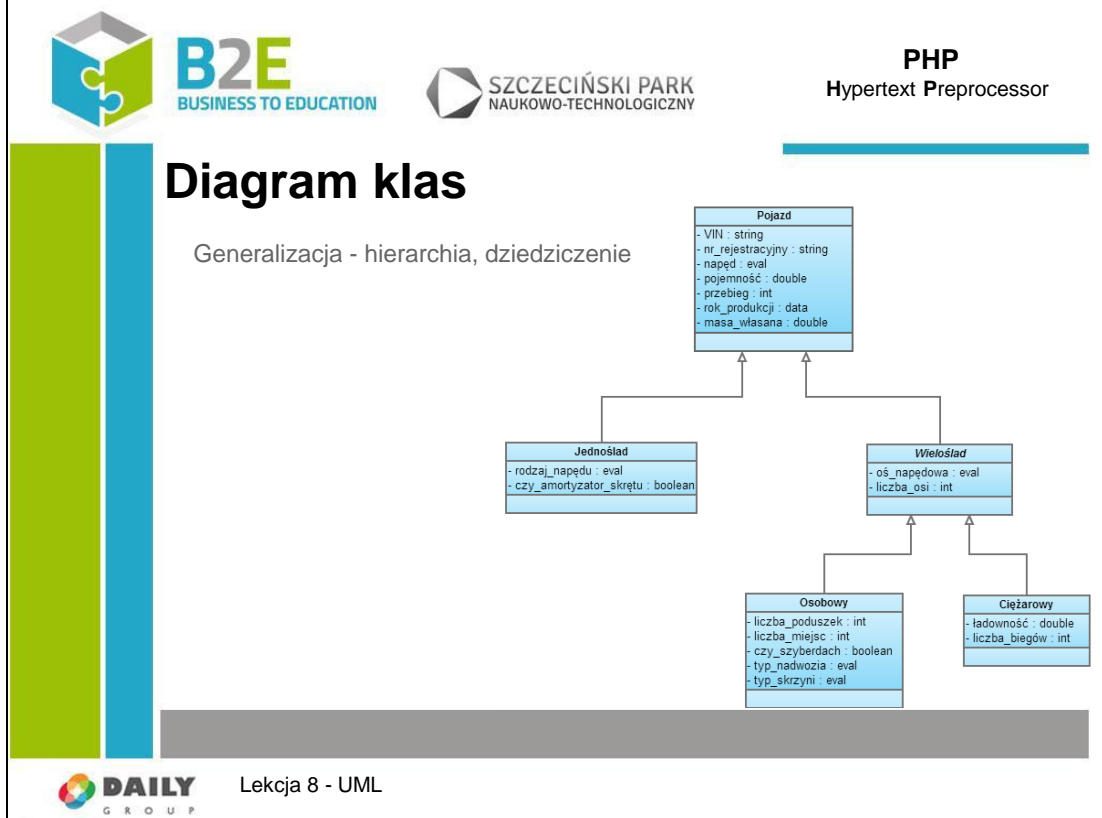


Podobnie jak na diagramie przypadków użycia, diagram klas wykorzystuje związki (relacje) do obrazowania zależności między klasami. Pierwszą relacją jest asocjacja która reprezentuje dowolny związek między obiektami klasy. Asocjacja jest reprezentowana linią ciągłą i może mieć następujące cechy:

- nazwę - podkreślająca istotę związku, posiadająca strzałkę prezentującą kierunek interpretacji (np. pojazd wykorzystuje koło),
- rolę - definiującą powinność obiektu względem drugiego (np. pojazd łączy koła, koło jest częścią pojazdu),
- nawigacja - precyzująca kierunek przepływu komunikatów (np. czujnik przekazuje sygnały do sterownika),
- liczebność - definiuje zakres dopuszczalnej liczby obiektów występujących w związku (np. pojazd ma od 2 do 5 kół, koło ma 1 pojazd, czujnik jest podłączony do jednego sterownika, sterownik posiada od zera do n czujników)
- agregacja - opisuje związek całość - część. Wyróżnia się
 - agregację silną (kompozycja) (zapęnlony romb) oznaczającą, że po usunięciu agregatu (obektu grupującego) usuwane są także wszystkie segmenty (obiekty składowe),
 - agregację częściową (pusty romb) oznaczającą, że po usunięciu agregatu segmenty nadal mogą istnieć.

Wykorzystywanie cech jest opcjonalne i zależy od przyjętego dla danego projektu szczegółowości dokumentacji.

Slajd
11



Generalizacja jest związkiem reprezentującym dziedziczenie klas. Dziedziczenie skutkuje powstaniem hierarchii klas w której najbardziej ogólna klasa jest umieszczona na samym szczycie hierarchii (korzeń). Na samym dole występują klasy nazywane liśćmi. W hierarchii występują też klasy abstrakcyjne (nazwa pisana kursywą). Klasy abstrakcyjne nie mają swoich instancji obiektów, służą natomiast do zdefiniowania wspólnych atrybutów i metod które będą dziedziczone przez klasy potomne (np. Wieloślad).

Człowiek - najlepsza inwestycja



Diagram klas

Ćwiczenie:

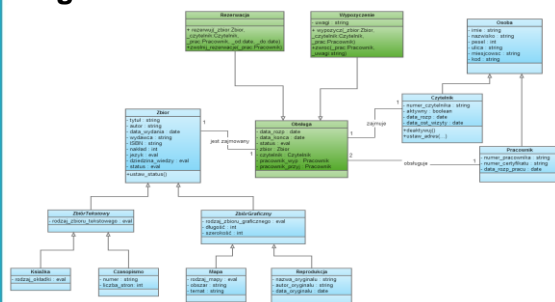
Zaproponuj diagram klas dla systemu zarządzania biblioteką publiczną.

Przyjmij przynajmniej 3 poziomą hierarchię opisującą zbiory biblioteczne i min 2 poziomą hierarchię klas opisującą użytkowników i pracowników biblioteki.

Wykorzystaj wcześniej przygotowany diagram przypadków użycia w celu weryfikacji kompletności funkcjonalności oprogramowania.



Diagram klas







W przykładowym rozwiązaniu zamodelowano tylko część struktury zbiorów bibliotecznych (tekstowe i graficzne – klasy abstrakcyjne). Rozróżniono osoby jako czytelników i pracowników biblioteki.

Centralną klasą wiążącą jest klasa Obsługa która jest powiązana


- ze zbiorem który jest wypożyczany (założono że na 1 wypożyczenie 1 zbiór)
- z czytelnikiem który zbiór wypożycza/rezerwuje
- z pracownikiem który zbiór przekazuje i przyjmuje/dokonuje rezerwacji

Potomkowie klasy Obsługa (Wypożyczenie, Rezerwacja) implementują metody Przedstawiono metody wypożycz i zwróć dla klasy

		<p>albo rezerwuj lub zwolnij</p> <p>Wypożyczenie oraz ustaw_status dla Zbiór i deaktywuj i ustaw_adres dla czytelnika</p>
Slajd 14	<div>   <p>SZCZECIŃSKI PARK NAUKOWO-TECHNOLOGICZNY</p> </div> <div> <p>Diagram czynności</p> <p>Diagramy czynności (aktywności)</p> <ul style="list-style-type: none"> • czynności • akcje • przepływ • start, koniec </div> <div> <p>PHP Hypertext Preprocessor</p> <pre> graph TD Start(()) --> Wyszukaj[Wyszukaj film] Wyszukaj --> Sprawdz[Sprawdź dostępność] Sprawdz --> Dec1{ } Dec1 -- dostępny --> Wypozycz[Wypożycz film] Dec1 -- niedostępny --> Rezerwuj[Rezerwuj] Wypozycz --> Dec2{ } Rezerwuj --> Dec2 Dec2 --> End((())) </pre> </div> <div>  <p>Lekcja 8 - UML</p> </div>	<p>Diagram czynności służy do zobrazowania algorytmów, procesów biznesowych w projektowanym oprogramowaniu lub scenariuszy przypadków użycia. Składają się z następujących elementów:</p> <ul style="list-style-type: none"> • czynności - reprezentująca czynność złożoną, możliwą do dalszej dekompozycji na podczynności i akcje i uszczegółowienia, (np. wyszukaj film), • akcji - elementarna jednostka zachowania, nie podlega dalszej dekompozycji, reprezentuje algorytm, metodę, • przepływów sterowania - strzałki łączące czynności, akcje i punkty startu i zakończenia, • punkt startu - czarne kółko, • punkt końca - okrąg z czarną kropką w środku



B2E
 BUSINESS TO EDUCATION

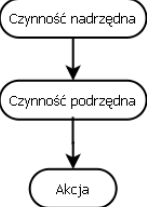


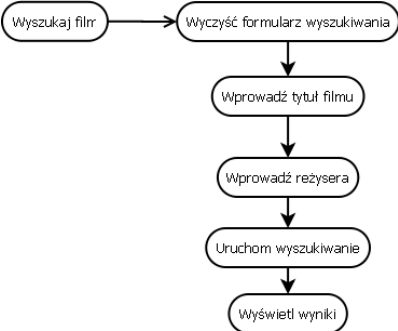
SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY


PHP
 Hypertext Preprocessor

Diagram czynności

Dekompozycja czynności









Lekcja 8 - UML

Czynności mogą być dekomponowane. Czynność nadrzędna po uszczegółowieniu może być procesem czynności podrzędnych albo sekwencją akcji. Przykładowo, czynność "Wyszukaj film" może składać się z sekwencji czynności.



B2E
 BUSINESS TO EDUCATION



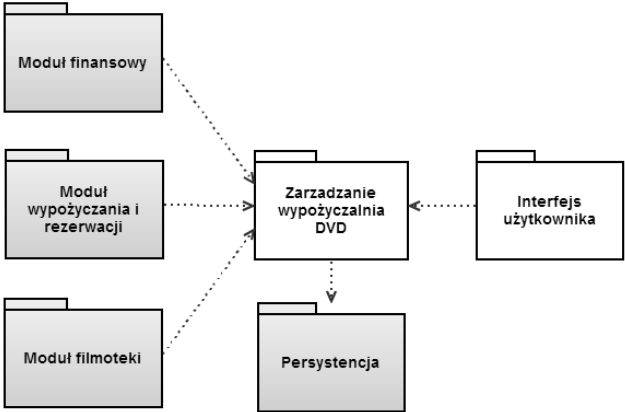
SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY


PHP
 Hypertext Preprocessor

Diagram pakietów

Diagram pakietów

- pakiet
- zależność
- zagnieżdżanie






Lekcja 8 - UML

Diagram składa się z następujących elementów:


Człowiek - najlepsza inwestycja

- pakiet - element grupujący inne diagramy, (ramka otaczająca elementy innych diagramów), praktyka zaleca objęcie pakietem diagramów
- zależność - łączy pakiety w logiczną strukturę, linia przerywana której kierunek wykorzystania pakietu,
- zagnieżdżanie - grupowanie innych pakietów.

Slajd
17



B2E
BUSINESS TO EDUCATION



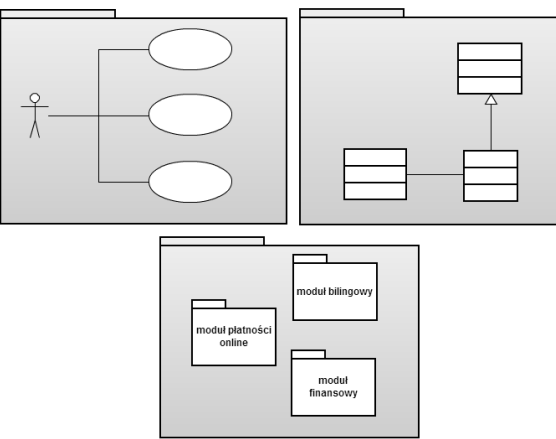
SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP
Hypertext Preprocessor

Diagram pakietów

Diagram pakietów

- zagnieżdżanie





Lekcja 8 - UML

Diagram pakietów pozwala na porządkowanie i grupowanie dokumentacji. Wszystkie diagramy mogą być organizowane w pakiety (np. pakiet grupujący przypadki użycia lub diagram klas). Pakiety mogą także grupować inne pakiety (np. pakiet dolny).

Człowiek - najlepsza inwestycja



Diagram pakietów

Ćwiczenie:

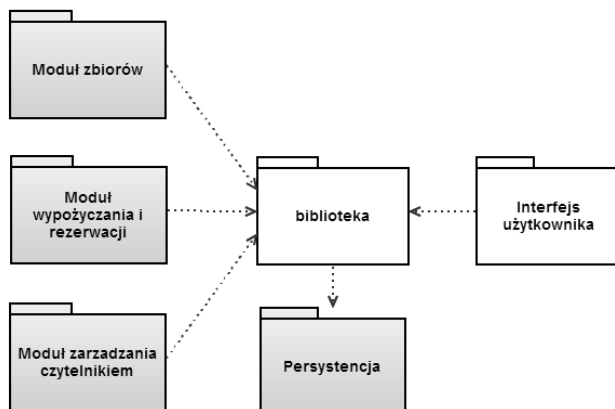
Zaproponuj podział oprogramowania do zarządzania biblioteką na pakiety.

Postaraj się pogrupować tematycznie klasy, przypadki użycia i diagramy czynności.

Następnie dokonaj zagnieżdżenia powstałych pakietów w pakiety nadrzędne - grupujące moduły między sobą np. zarządzanie czytelnikami, zarządzanie zbiorami bibliotecznymi.



Diagram pakietów - rozwiązanie



Założono rozdzielenie modułów:

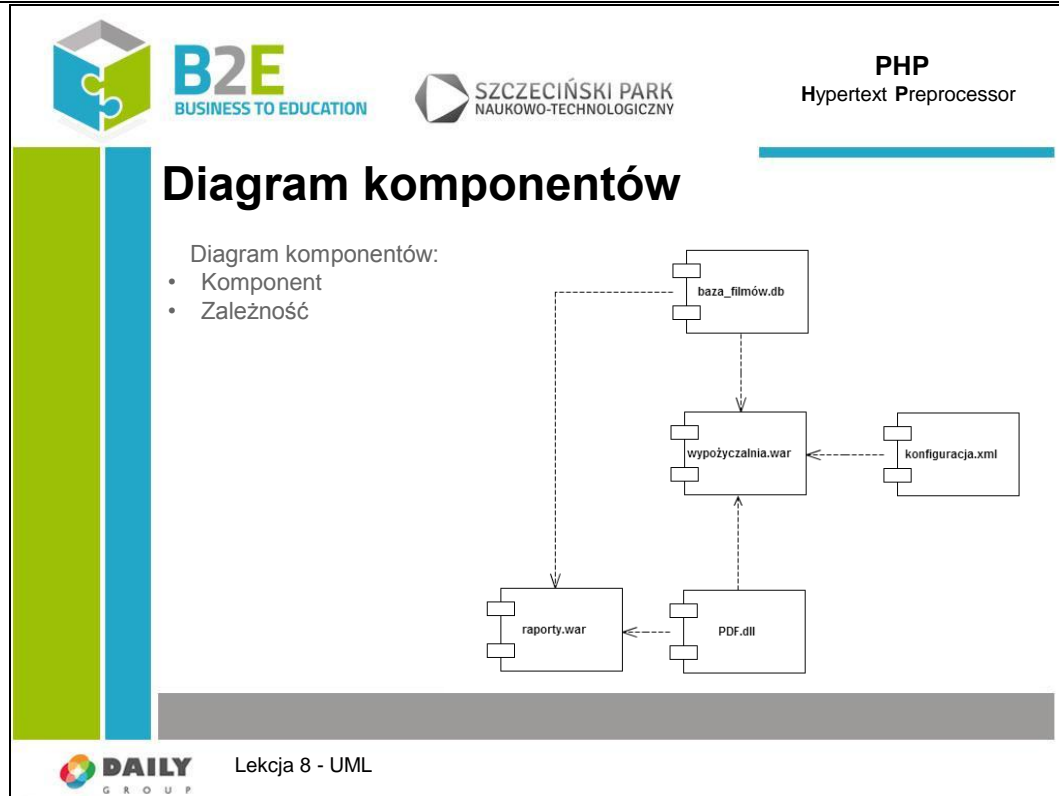
Zarządzania zbiorami

Wypożyczania i rezerwacji

Zarządzania czytelnikiem

Zaprojektowano, że stan systemu będzie zapisywany w bazie danych (persystencja)
Założono prosty interfejs użytkownika (GUI) dla pracowników i czytelników

Slajd
20



Komponentem jest izolowany, wymienny moduł oprogramowania stanowiący część systemu. Komponentami może być każdy element składowy systemu przykładowo:

- bazy danych (np. plik bazy danych, system zarządzania bazami danych),
- pliki bibliotek programowych (np. moduł do tworzenia plików PDF),
- moduły oprogramowania (np. wypożyczalnia, raporty),
- pliki (np. konfiguracyjne, graficzne).

Komponent jest reprezentowany przez symbol prostokąta z dwoma bolcami (symbolizuje interfejsu). W nowszej notacji UML dodano dodatkowe ikony komponentów pozwalających na definiowanie i nazywanie interfejsów.

Zależności komponentów są reprezentowane przez linię przerywaną z grotem. Kierunek wskazuje na komponent wykorzystujący.

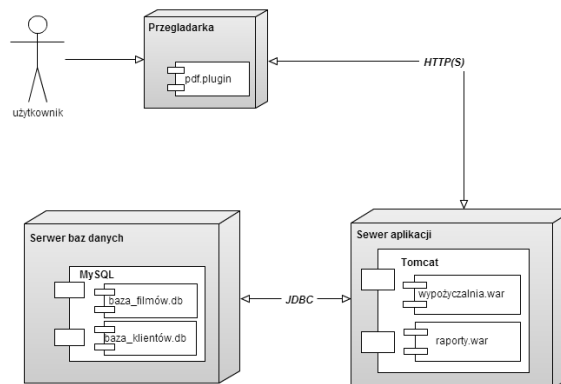
Człowiek - najlepsza inwestycja



Diagram wdrożeniowy

Diagram rozlokowania:

- artefakt,
- węzeł,
- ścieżki komunikowania.



W dokumentacji często występuje potrzeba określenia fizycznego rozmieszczenia artefaktów (sztucznie wytworzonych elementów, np. komponentów) oprogramowania. Artefaktem mogą być bazy danych (np. baza_filmów.db), klasy, diagramy, biblioteki, komponenty (np. wypożyczalnia.war)

Węzłem jest fizyczny lub logiczny zasób przetwarzający na którym zainstalowano i skonfigurowano do działania artefakty (np. serwer baz danych, przeglądarka).

Ścieżki komunikowania reprezentują przepływ i kierunek przepływu informacji. Nazwa ścieżki może sugerować protokół komunikacyjny który projektuje się wykorzystać.

Slajd
22



PHP
Hypertext Preprocessor

Diagram wdrożeniowy

Ćwiczenie:

Zaproponuj diagram wdrożeniowy dla oprogramowania zarządzającego biblioteką.

Postaraj się rozdzielić funkcje serwera aplikacji od serwera baz danych, umieść w węzłach odpowiednie komponenty.

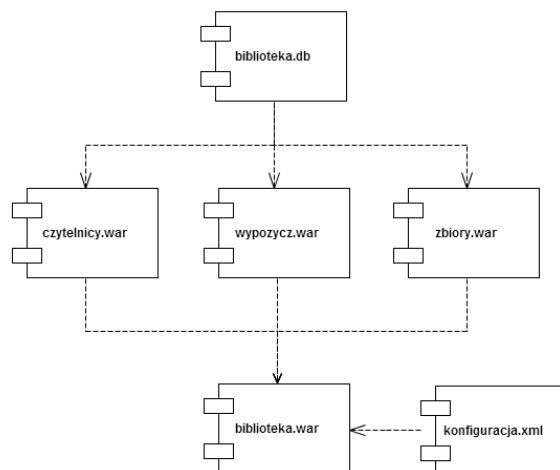


Lekcja 8 - UML

Slajd
23



PHP
Hypertext Preprocessor



Lekcja 8 - UML

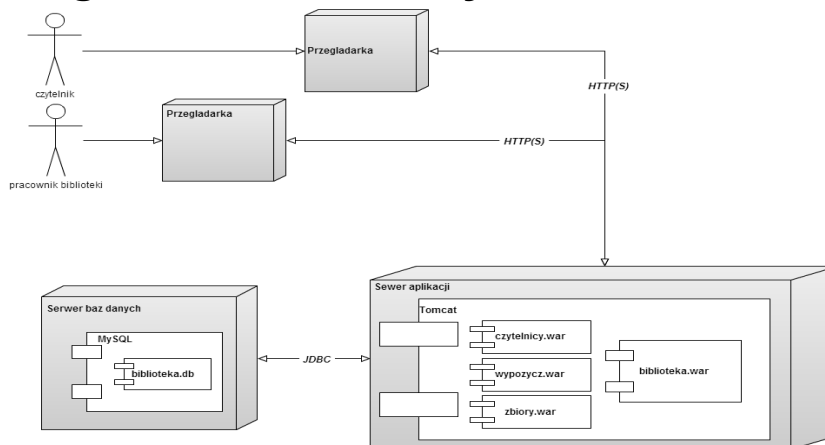
Założono realizację wspólnej bazy danych i rozdzielenie modułów funkcjonalnych wg pakietów
Całość zintegrowana interfejsem użytkownika biblioteka.war oraz moduł konfiguracji technicznej.

Człowiek - najlepsza inwestycja





Diagram wdrożeniowy



Założono instalację rozwiązania na dwóch serwerach – serwer baz danych oraz serwer aplikacyjny

Przyjęto zastosowanie darmowych rozwiązań MySQL i implementacji baz danych

Przyjęto zastosowanie serwera aplikacyjnego tomcat i uruchamianie rozwiązania z poziomu tego serwera

Założono że dostęp do systemu odbywa się wyłącznie przez przeglądarkę.



Slajd
25



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Ćwiczenia

1. Co to jest UML? Co oznacza skrót? Do czego służy?
2. Wymień min 5 diagramów UML. Jakie znasz perspektywy projektowania oprogramowania?
3. Jakim diagramem zobrazujesz wymagania na system informatyczny? Narysuj przykład diagramu i omów go.
4. Jakim diagramem zobrazujesz model obiektowy? Narysuj przykład diagramu prezentującego dziedziczenie i omów jego elementy.
5. Jakim diagramem zaprezentujesz algorytm lub proces który należy zaimplementować w systemie? Narysuj przykład diagramu i omów go.
6. Jakim diagramem zorganizujesz i pogrupujesz inne diagramy?
7. Jaki diagram służy do przedstawienia fizycznego rozmieszczenia elementów oprogramowania?
8. Do czego służy diagram komponentów?

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego



Lekcja 8 - UML

10.9.3 Ćwiczenia

Ćwiczenia dla tej jednostki lekcyjnej zostały umieszczone na ostatnim slajdzie.

10.9.4 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie będą potrafili samodzielnie zaprojektować aplikację w języku UML oraz stworzyć podstawowe diagramy aplikacji.

10.10 Lekcja 10 - Dobre praktyki programowania w PHP

10.10.1 Cel lekcji

Celem lekcji jest wprowadzenie ucznia w świat praktycznego podejścia do aplikacji oraz wytyczenie im dalszej drogi na rozwój w przyszłości. Pokazanie im dobrych praktyk oraz wielu przydatnych fragmentów kodu, szkieletów oraz narzędzi, które przydatne są każdemu programiście. Zaprezentowano również wzorce projektowe i obecnie najwyższy abstrakcyjnie paradygmat programowania, zwany programowaniem zorientowanym na aspekty.

Człowiek - najlepsza inwestycja



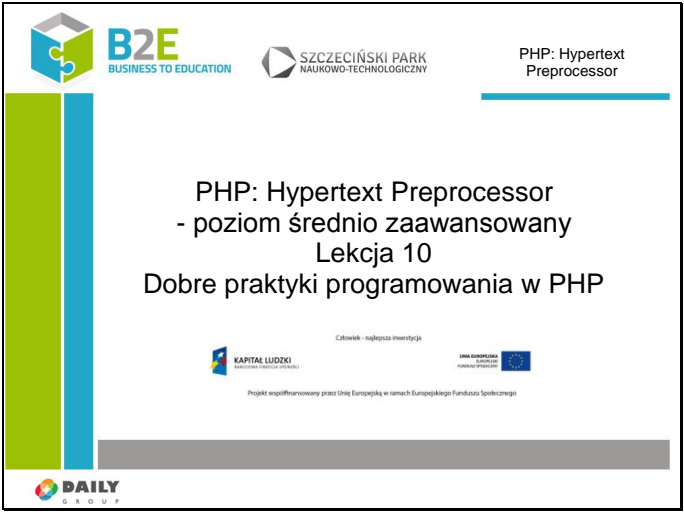
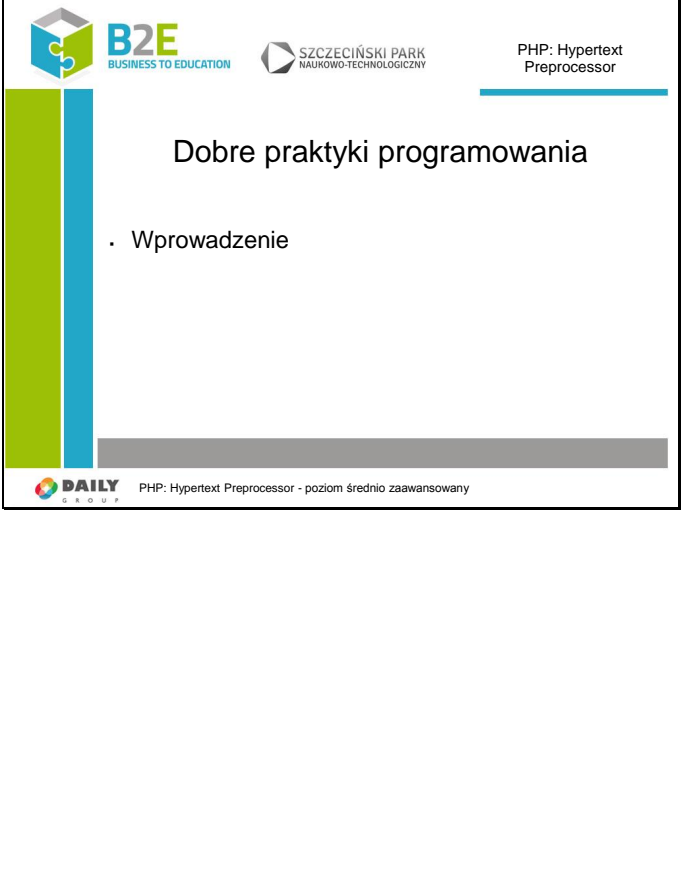
KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

10.10.2 Treść - slajdy z opisem

<p>Slajd 1</p>		<p>Słowa wstępu</p>
<p>Slajd 2</p>		<p>PHP jest językiem, w którym łatwo rozwiązuje się większość problemów. Jego prostota niesie jednak ze sobą poważną konsekwencję, kod który zaimplementowaliśmy, a który wymaga zmian, jest niezrozumiały i wymaga przepisania od nowa, aby zawrzeć odpowiednie zmiany. Często też kod, który rozwiązuje podobny problem w innym projekcie, nie nadaje się do zastosowania. Należy zatem zadać sobie pytanie, czy powinniśmy po raz kolejny rozwiązywać ten sam problem, bądź też na nowo odkrywać koło, czyli pisać rzeczy, które mam w najlepszy i sprawdzony sposób zaimplementowane przez innych programistów.</p>

Człowiek - najlepsza inwestycja



PHP: Hypertext
Preprocessor

Rozszerzenia PHP

- PEAR
- PECL
- Composer



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

PEAR jest to system dystrybucji opensource'owych rozszerzeń języka PHP. Powstał od roku 1999 i posiada wielką ilość zasobów, które ułatwiają nam pracę z językiem PHP. PECL to pierwotnie część PEAR, w przeciwieństwie do swojego rodzica, jest to repozytorium społeczności, a pakiety w nim zawarte, są pakietami nieoficjalnymi. Zanim napiszesz coś co już zostało napisane, zajrzyj do repozytorium, gdyż może to Ci pomóc zaoszczędzić mnóstwo czasu i dostarczyć źródło ze sprawdzonym rozwiązaniem.

Używając wielu frameworków oraz dodatków, do zarządzania zależnościami w projekcie przydatny może okazać się Composer (<https://getcomposer.org/>). Composer jest genialnym systemem wspomagającym zarządzanie zależnościami aplikacji w PHP. Wystarczy jeżeli wylistujesz swoje zależności w pliku composer.json, a narzędzie to, przy użyciu kilku prostych poleceń, automatycznie ściągnie odpowiednie wersje bibliotek i ich zależności oraz odpowiednio skonfiguruje autoloadera. Za pomocą Composera można zainstalować wiele znanych bibliotek. Ich listę znajdziesz na stronie projektu Packagist - oficjalnego repozytorium bibliotek kompatybilnych z tym systemem.

Człowiek - najlepsza inwestycja





B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Frameworki

- Wprowadzenie
- ZF
- Symfony
- Slim



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Implementując aplikację internetową wykonujemy szereg operacji, które związane są z jej szkieletem oraz często powtarzalnymi rzeczami, jak autoryzacja czy systemy formularzy. Z pomocą przychodzą nam gotowe szkielety aplikacji takie jak Zend Framework czy Symfony. Jeżeli piszemy aplikację, która jest większa niż formularz kontaktowy, warto użyć gotowych rozwiązań. Szkielety te posiadają architekturę MVC oraz całą masę gotowych bibliotek np. pośredniczących w komunikacji z bazą danych, związanych z cache'owaniem, czy chociażby obsługą plików pdf. Wersje 2 pierwszych frameworków, są zaimplementowane w pełni obiektowo i wykorzystują wiele wzorców poprawnego programowania, z których można czerpać świetne lekcje. Na ich stronach znajdziemy mnóstwo materiałów związanych z ich użyciem, video-tutoriale oraz wyczerpującą dokumentację np. ZF około 1300 stron formatu A4. Oprócz wyżej wymienionych frameworków, jest jeszcze wiele innych jak np. Yii, cakePHP, codeigniter czy fuel. Warto poświęcić trochę czasu na eksplorację internetu w poszukiwaniu tych zasobów, ponieważ można zaoszczędzić masę pracy oraz otrzymać sporo profesjonalnie napisanego kodu.

Slim (<http://www.slimframework.com/>) jest natomiast przykładem frameworka, przy pomocy którego w szybki i prosty sposób będziemy mogli tworzyć własne API (np REST API dla aplikacji na Androidzie – Lekcja 11).

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
5


B2E
 BUSINESS TO EDUCATION
 
 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Biblioteki



- Przykłady
- Composer


 DAILY GROUP

PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Jeżeli nie czujesz się jeszcze na siłach, aby korzystać z wyrafinowanych narzędzi, ponieważ ich ogrom Cię przeraża lub piszesz tylko małe aplikacje dla których szkoda tracić zasoby i wydajność, możesz korzystać z frameworków jako bibliotek lub ze stricte bibliotek, których jest wiele np. TCPDF do obsługi plików pdf, czy biblioteki SMARTY do obsługi szablonów. Użytecznym narzędziem okazuje się również Composer, czyli program do zarządzania pakietami PHP.


Slajd
6


B2E
 BUSINESS TO EDUCATION
 
 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Programowanie obiektowe (OOP), a aspektowe (AOP)

- Wprowadzenie
- Czy warto z niego korzystać?


 DAILY GROUP



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

PHP w wersji 5, zapewnia pełną obsługę programowania obiektowego, czyli najpopularniejszego obecnie programowania. Kontynuacją drogi ku wydzieleniu

Człowiek - najlepsza inwestycja

abstrakcji w językach programowania jest programowanie aspektowe. Implementuje się je na raz w sposób obiektowy, gdyż PHP nie zapewnia jeszcze wbudowanej obsługi aspektów. Polega ono na wydzieleniu abstrakcyjnych czynności na poziomie rozumowania człowieka np. system logów, i uniezależnienie ich od klas. Grupowanie takie zapewnia nam lepsze możliwości projektowania i bardziej zrozumiałą implementację. Zobrazujemy nasz przykład, założymy że nasz zbiór klas przy wykonywaniu jakiejś czynności, zapisuje te akcje w tzw. Logach. Czy nie łatwiej by było wyodrębnić ten mechanizm z klas i przenieść go do poziomu kontenera naszej aplikacji? Zyskujemy na tym wiele czasu, nie musimy implementować go za każdym razem. Poza tym wszelkie zmiany dokonujemy tylko raz. Co w przyszłości i przy częstych zmianach w aplikacji staje się uciążliwe.


Slajd
7

PHP: Hypertext Preprocessor

Wzorce projektowe

- Wprowadzenie
- Do czego służą?



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Kolejną ważną rzeczą, która ułatwia życie programiście są wzorce projektowe. Są to rozwiązania problemów projektowych, które stosowane w odpowiednich przypadkach, potrafią zaoszczędzić nam wiele czasu i stworzyć łatwe do dalszego rozwijania środowisko, czy też uniknąć wielu błędów.

Slajd
8




PHP: Hypertext Preprocessor

Wzorce projektowe ćwiczenie

Analiza przykładu wzorca projektowego singleton lub MVC.

Singleton implementation






Conventional implementation





PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Człowiek - najlepsza inwestycja

	<p>Celem wzorca projektowego singleton jest wprowadzenie ograniczenia możliwości tworzenia obiektów danej klasy do tylko jednej instancji. Tak stworzony obiekt dostępny jest globalnie dla całej aplikacji.</p> <p>Wzorca singleton powinniśmy używać do implementacji:</p> <ul style="list-style-type: none"> - obiektów, przechowujących informacje o stanie aplikacji (np konfiguracji). - obiektów które powinny istnieć przez cały czas trwania aplikacji (np logowanie zdarzeń z wielu miejsc w kodzie do jednego pliku Log) <p>Przykład znajduje się w materiałach ze skryptami PHP.</p> <p>Zmiany w stosunku do skryptu nie używającego wzorca singleton:</p> <ul style="list-style-type: none"> - konstruktor został zmieniony z publicznego na prywatny (to chroni przed tworzeniem obiektu bez wywołania metody getInstance) - zmienna \$dbh zmieniona z publicznej na prywatną (chroni to przed nadpisaniem zmiennej \$dbh) - tablica db_info zamieniona z publicznej na prywatną (z tego samego powodu co \$dbh) <p>Grafika pobrana ze strony: http://yavkata.co.uk/weblog/php/php-singleton-design-pattern/</p> <p>Kolejnym bardzo popularnym wzorcem projektowym jest MVC (Model-Widok-Kontroler). Używając tego modelu, aplikacje zawierające interfejs użytkownika są zorganizowane w 3 struktury:</p> <ul style="list-style-type: none"> - Model – zawiera logikę aplikacji - Widok – odpowiedzialny jest za wyświetlenie rezultatów modelu w interfejsie użytkownika - Kontroler - przyjmuje dane wejściowe od użytkownika i na ich podstawie dostarcza oraz pobiera dane z/do modelu oraz odświeża widok
Slajd 9	<div data-bbox="284 1232 973 1747">   <div data-bbox="805 1265 933 1310"> PHP: Hypertext Preprocessor </div> <hr/> <div data-bbox="391 1355 949 1411"> Dobre praktyki samego programowania </div> <div data-bbox="391 1444 646 1489"> . http://goo.gl/suWCFu </div> <div data-bbox="311 1702 406 1736">  </div> <div data-bbox="422 1713 766 1736"> PHP: Hypertext Preprocessor - poziom średnio zaawansowany </div> </div> <div data-bbox="989 1220 1402 1590"> <p>Jeśli chodzi o samo pisanie w języku PHP, w internecie możemy znaleźć wiele stron poświęconych dobrym praktykom programowania. Warto polecić „php the right way”, gdyż jest ona podstawą wielu publikacji na temat dobrych praktyk w języku PHP.</p> </div>



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Dokumentowanie kodu

- Wprowadzenie
- phpDocumentor

```
/**
 * Displays a new message box, or reinitializes an existing message box, based on the config options passed in.
 *
 * @param config the message box config
 */
public static native void show(MessageBoxConfig config) /*-{
    $wnd.Ext.MessageBox.show(config.$com.gwtExt.client.core.JsObject::jsObj);
}-*/;
```



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Jeżeli chcemy używać naszego kodu wielokrotnie, a nie pracujemy samemu, bądź też chcemy udostępniać nasz kod dla innych programistów. Warto zadbać o dobre udokumentowanie kodu. Ułatwia ono i przyspiesza pracę nie tylko innym, ale też nam samym, kiedy po dłuższym okresie sięgamy po nasz wcześniej napisany kod. Polecanym tutaj narzędziem jest phpDocumentator, który tworzy dokumentację automatycznie na podstawie specjalnych komentarzy zamieszczanych w skrypcie.

PhpDocumentor jest narzędziem, dzięki któremu możliwe jest generowanie dokumentacji z kodu źródłowego PHP. Dzięki takiej dokumentacji, można dostarczyć odbiorcy więcej informacji związanych z funkcjonalnością osadzoną w kodzie źródłowym, a nie tylko to, co jest wykorzystywane w interfejsie użytkownika.

Dokumentacja taka jest bardzo przydatna w następujących sytuacjach:

- Gdy tworzymy zestaw aplikacji udostępniające API
- Gdy tworzymy aplikacje opartej na architekturze pluginowej, umożliwiającej tworzenie własnych dodatków

Więcej informacji dostępnych na stronie: <http://www.phpdoc.org/>

Człowiek - najlepsza inwestycja





KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
11



B2E
 BUSINESS TO EDUCATION


 SZCZECIŃSKI PARK
 NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
 Preprocessor



Kontrola wersji

- Co to jest i do czego służy?
- SVN
- GIT


 PHP: Hypertext Preprocessor - poziom średnio zaawansowany

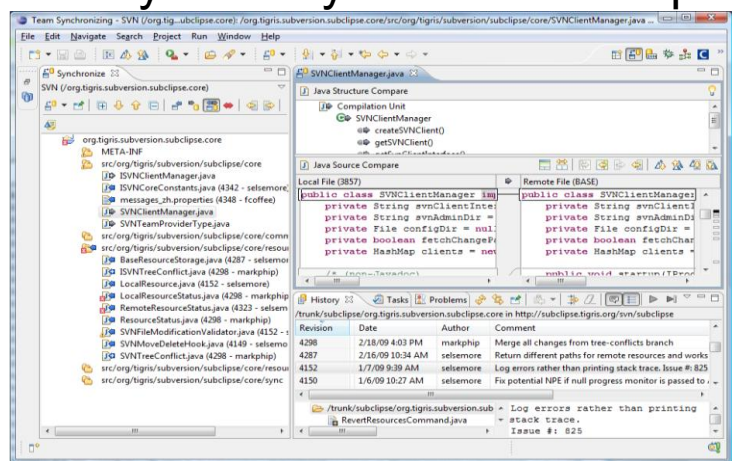
Systemy kontroli wersji lub w skrócie systemy wersjonowania, są to programy, które przechowują historię naszej pracy i ułatwiają pracę w większej grupie programistów. Do najbardziej popularnych należą subversion (SVN) oraz git. Po napisaniu danego fragmentu aplikacji, możemy swobodnie wysłać ją na serwer i dołączyć do całości większej aplikacji, możemy także pobierać fragmenty napisane w przeszłości, czy przeglądać zmiany zrobione przez innych w naszym kodzie. Systemy te pozwalają także na tworzenie repozytoriów na serwerach i udostępnianie najnowszych wersji naszych bibliotek, czy też aplikacji innym. Do komunikacji z serwerami zawierającymi usługi wersjonowania często używamy pluginów (np do ulubionego IDE) lub samodzielnych aplikacji jak np. TortoiseSVN.


Człowiek - najlepsza inwestycja

PHP: Hypertext
Preprocessor

Przykład użycia SVN w Eclipse







PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Na przykładzie widzimy zmiany przychodzące i wychodzące. Zmiany wychodzące to praca wykonana przez nas. Zmiany przychodzące to zmiany wykonane przez innych deweloperów. Z prawej strony widzimy konkretne zmiany dokonane na wybranym pliku.


(screen pobrany ze strony: <http://www.java-forums.org/blogs/eclipse/975-working-svn-eclipse.html>).

PHP: Hypertext
Preprocessor

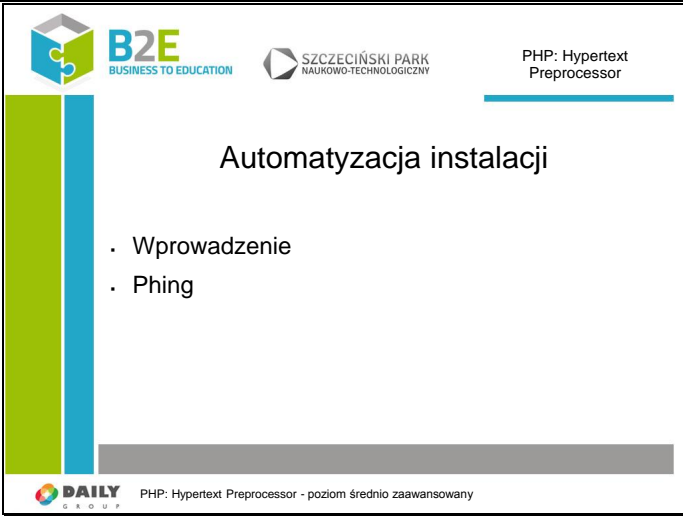
Testowanie aplikacji

- Wprowadzenie
- Testy jednostkowe
- phpUnit



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Testowanie aplikacji to jedna z ważniejszych części programowania, mamy różne rodzaje testów np. integracyjne, wydajnościowe. Jeśli chodzi jednak o najczęściej przeprowadzane testy przez samych programistów, są to testy jednostkowe. Testy jednostkowe to testy poszczególnych metod, procedur, bądź obiektów. Z pomocą przy testowaniu przychodzi nam biblioteka automatyzująca testy jednostkowe phpUnit. Warto z

		niej korzystać, ponieważ zaoszczędzimy sporo pracy.
Slajd 14		<p>PHP jest językiem skryptowym, stąd też nie posiada instalatorów znanych nam z systemów operacyjnych w formie pliku wykonywalnego. Są jednak narzędzia np. Phing, które pomogą nam stworzyć instalator aplikacji. Phing jest narzędziem niezależnym od platformy, wzorowanym na java-owym ANT-cie. Reguły budowania określa się w pliku xml. Po zakończonym procesie możemy zainstalować naszą aplikację, korzystając przy tym np. z rozszerzenia PEAR.</p>

Człowiek - najlepsza inwestycja



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP: Hypertext
Preprocessor

Gotowe systemy

- CMS
- Fora
- Blogi
- Sklepy
- Platformy e-learningowe



PHP: Hypertext Preprocessor - poziom średnio zaawansowany

Oprócz wcześniej wspomnianych mniejszych elementów jak biblioteki, przy tworzeniu nowej aplikacji, warto się zastanowić, czy jako podstawę do tworzenia, nie wykorzystać jakiegoś gotowego systemu, który jest udostępniany na otwartej licencji i rozwijany od wielu lat. Nasza część implementacji w tym momencie opiera się wtedy tylko na dopisaniu odpowiednich wtyczek, komponentów, bloków lub modułów, w zależności od wybranego systemu. Wśród systemów zarządzania treścią mamy ogromny wybór, polecić tu jednak można system Joomla, czy Drupal, które należą do najbardziej rozbudowanych. Jeśli chodzi o fora internetowe napisane w języku PHP, szczególną popularnością cieszy się phpBB, z dużym wyborem darmowych szat graficznych i dodatków. Wordpress jest bezkonkurencyjnym systemem, jeśli chodzi o blogi internetowe, choć swój rozwój kieruje w stronę systemu CMS, pewnie z uwagi na prostotę i wygodę zarządzania tym systemem. Jeśli chodzi o darmowe sklepy wykonane w technologii PHP, z racji ogromnej ilości pluginów, warto polecić prestashop. Jeśli chodzi o platformy e-learningowe nie ma za dużego wyboru, zgodność ze standardami i wysoki poziom funkcjonalności, zapewnia tylko system Moodle.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



<p>Slajd 16</p>	  <div>PHP: Hypertext Preprocessor</div> <h3>Instalacja CMS – Joomla</h3> <ul style="list-style-type: none"> • Zainstaluj system zarządzania treścią Joomla. • Dokonaj zmian poprzez panel administracyjny. • Zainstaluj dodatek. • Znajdź, pobierz i zainstaluj darmowy szablon zmieniający wygląd strony.  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>Dla uczniów zaawansowanych można zaproponować napisanie własnego modułu/komponentu instalowanego w Joomla.</p>
<p>Slajd 17</p>	  <div>PHP: Hypertext Preprocessor</div> <h3>Ćwiczenie praktyczne</h3> <ul style="list-style-type: none"> • Napisz własny moduł dla CMSa Joomla. Moduł powinien być instalowalny poprzez panel administracyjny Joomla. Moduł może wyświetlać dowolny komunikat, celem zadania jest poznanie struktury modułów.  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>mod_firstmodule/*.*</p>
<p>Slajd 18</p>	  <div>PHP: Hypertext Preprocessor</div> <h3>Ćwiczenie praktyczne</h3> <ul style="list-style-type: none"> • Uruchom i przeanalizuj przykład zawarty w pliku singleton2.php. Przykład ten zawiera implementację książki, która jest singletonem, oraz 2 użytkowników, którzy mogą wykonać na książce pewne operacje. Co stanie się gdy użytkownik 2 spróbuje wypożyczyć książkę, która była wypożyczona wcześniej przez użytkownika 2?  <div>PHP: Hypertext Preprocessor - poziom średnio zaawansowany</div>	<p>singleton2.php</p>

Człowiek - najlepsza inwestycja

Slajd
19



Slide 19 content includes logos for B2E (Business to Education), Szczeciński Park Naukowo-Technologiczny, and PHP: Hypertext Preprocessor. The main text reads 'Dziękujemy za uwagę' (Thank you for your attention). At the bottom, it mentions 'DAILY GROUP' and 'PHP: Hypertext Preprocessor - poziom średnio zaawansowany' (PHP: Hypertext Preprocessor - intermediate level). There are also logos for 'KAPITAŁ LUDZKI' (Human Capital) and the European Union flag.

10.10.3 Ćwiczenia

Ćwiczenie 1 Zapoznać się z dokumentacją wybranego przez siebie, nowego (nieznanego) narzędzia, frameworku lub biblioteki napisanej w języku php.

Ćwiczenie 2 Wykonać jeden lub więcej podstawowych przykładów (tutoriali) dostępnych na stronie wybranej technologii lub w jej dokumentacji.

10.10.4 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie będą mieli szerszy pogląd na temat praktycznego wykorzystania języka PHP w przyszłych projektach. Będą wiedzieli, czym są wzorce projektowe, jakie narzędzia i biblioteki warto poznać oraz jakie gotowe systemy proponuje im społeczność programistów języka PHP. Lekcja ma też na celu zachęcić ich do obrania własnej ścieżki rozwoju i dalszej pracy z językiem PHP.

10.11 Lekcja 11 - Interdyscyplinarny projekt książki teleadresowej


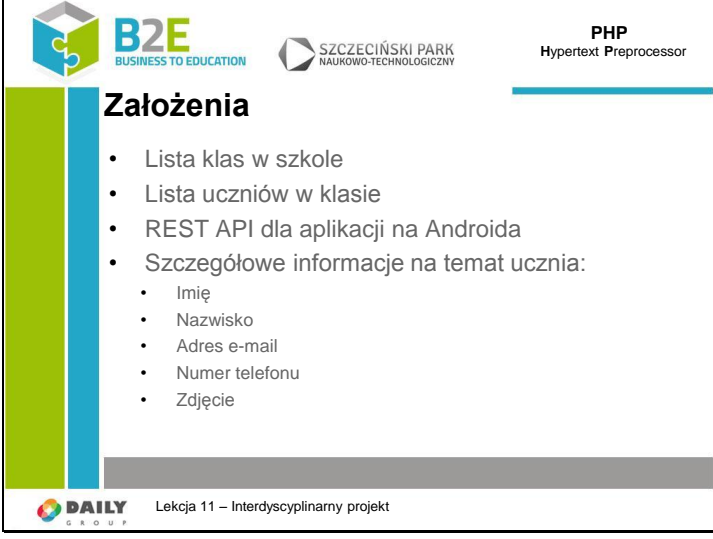
10.11.1 Cel lekcji

Celem lekcji jest zaprojektowanie strony WWW oraz w aplikacji mobilnej z funkcjonalnością książki teleadresowej kolegów i koleżanek ze szkoły lub klasy. W tej lekcji uczniowie dowiedzą się jak praktycznie wykorzystać wiedzę zdobytą w tym module na potrzeby pierwszego projektu informatycznego składającego się z elementów wykorzystujących cztery języki programowania: SQL, PHP, JavaScript i JAVA.







Człowiek - najlepsza inwestycja



10.11.2 Treść - slajdy z opisem



<p>Slajd 1</p>		<p>Proces powstawania książki teleadresowej kolegów/koleżanek ze szkoły. W module SQL pokażemy jak zaprojektować bazę danych, na PHP i JavaScript, jak stworzyć stronę WWW do dodawania kolegów i koleżanek z klasy do takiej książki, a w module JAVA pokażemy, jak zaprogramować prostą aplikację na telefon z systemem operacyjnym Android do przeszukiwania danych o uczniach z różnych klas.</p>
<p>Slajd 2</p>		<p>Założenia projektu.</p>

Człowiek - najlepsza inwestycja

<p>Slajd 3</p>	  <p>PHP Hypertext Preprocessor</p> <h3>Struktura projektu</h3> <ul style="list-style-type: none"> • Template (index.php) • Content – odpowiednie podstrony projektu (home.php, klasy.php, uczniowie.php) • Obsługa połączenia do bazy danych (db.class.php) • Klasa do wykonywania operacji na bazie danych - odczyt, zapis, usuwanie (data.class.php)  <p>Lekcja 11 – Interdyscyplinarny projekt</p>	<p>Struktura projektu.</p>
<p>Slajd 4</p>	  <p>PHP Hypertext Preprocessor</p> <h3>Alfabetyczna struktura katalogów projektu</h3> <ul style="list-style-type: none"> • API – REST API dla aplikacji na Androida • Content – kolejne podstrony aplikacji • Css, images – style oraz grafika • Js – skrypty Java Script • Lang – pliki językowe • Libs – zewnętrzne biblioteki • Upload – folder ze zdjęciami wysłanymi z aplikacji • Utils – klasy, funkcje oraz konfiguracja  <p>Lekcja 11 – Interdyscyplinarny projekt</p>	<p>Struktura projektu. API – implementacja REST API dla androida, wykorzystuje SLIM framework</p> <p>Wielojęzykowość aktualnie nie jest uruchomiona, ale stanowić może załączek dla dalszego rozwoju aplikacji i dodatkowych zadań dla ucznia. Plik index.php zawiera zakomentowane fragmenty kodu odpowiedzialne za sterowanie wielojęzykowością – wyświetlanie odpowiednich flag, załączanie odpowiednich plików językowych w zależności do wcześniejszego wyboru, przykładowe wyświetlenie zawartości pliku językowego (np menu) w zależności od wybranego języka. Aktualnie domyślnie dołączony jest polski plik językowy.</p>

Człowiek - najlepsza inwestycja


Slajd
5

PHP
Hypertext Preprocessor



Cechy projektu

- Napisany jest w PHP z wykorzystaniem klas oraz zewnętrznej biblioteki Slim dla REST API
- Jest idealną bazą do testowania, rozwijania i nauki dla poszczególnych lekcji kursu
- Posiada przykłady walidacji, obsługi błędów poprzez wyjątki
- Operacje CRUD dla klas oraz uczniów


 Lekcja 11 – Interdyscyplinarny projekt

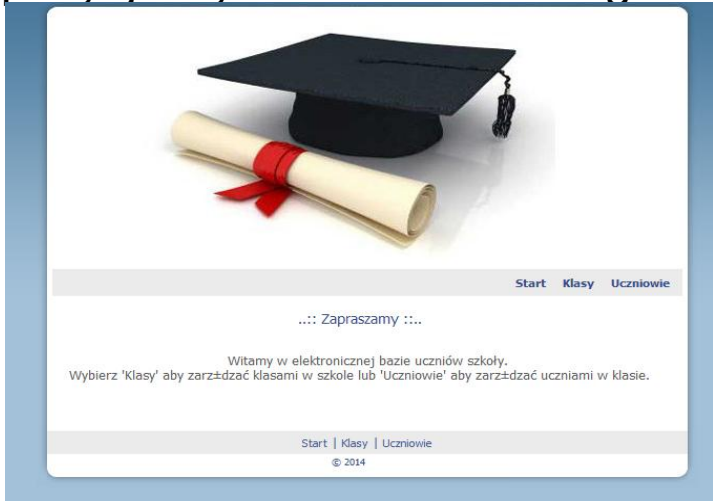
CRUD - Create, read, update and delete
Slim -
(<http://www.slimframework.com/>) jest przykładem frameworka, przy pomocy którego w szybki i prosty sposób będziemy mogli tworzyć własne API (np REST API dla aplikacji na Androidzie – Lekcja 11 Java)


Slajd
6

PHP
Hypertext Preprocessor

Propozycja wykonania – strona główna




 Lekcja 11 – Interdyscyplinarny projekt

Podstawowe komendy związane z transakcjami.

Człowiek - najlepsza inwestycja



Propozycja wykonania – widok klas

Klasy

ID	Nazwa klasy	Wychowawca	Edycja danych	Dodaj zdjęcie	Usuń
1	1A	Michał Gałas		(1)	
2	1B	Anna Komin		(1)	
3	2C	Marcelina Jeż		(1)	
4	3B	Jan Nowak		(0)	
5	4A	Ewa Komin		(0)	

Nazwa klasy:

Wychowawca:



Lekcja 11 – Interdyscyplinarny projekt

W podstawowym widoku klas wylistowana jest lista wszystkich klas w szkole. Wyświetlane jest ich:

- Id w bazie
- Nazwa klasy
- Wychowawca
- Przycisk do bezpośredniej edycji danej klasy
- Przycisk umożliwiający dodanie zdjęcia nauczyciela, oraz w nawiasie informacja czy zdjęcie już jest dodane (1) czy nie (0)
- Przycisk umożliwiający skasowanie klasy – jest to możliwe gdy w klasie nie ma żadnych uczniów

Poniżej znajduje się formularz umożliwiający dodanie nowej klasy.

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
8



B2E
BUSINESS TO EDUCATION



SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Propozycja wykonania – edycja klasy

Edycja klasy:

Nazwa klasy:

Wychowawca:

Klasy

ID	Nazwa klasy	Wychowawca	Edycja danych	Dodaj zdjęcie	Usuń
1	1A	Michał Galas		 (1)	
2	1B	Anna Komin		 (1)	
3	2C	Marcelina Jeż		 (1)	
4	3B	Jan Nowak		 (0)	
5	4A	Ewa Komin		 (0)	



Lekcja 11 – Interdyscyplinarny projekt

Formularz do edycji klasy, widoczny po wciśnięciu przycisku edycji klasy (symbol ołówka na kartce)

Człowiek - najlepsza inwestycja






Propozycja wykonania – edycja klasy, obsługa błędów





BŁĄD!
Nie można zmienić nazwy tej klasy ponieważ inna klasa o podanej nazwie już istnieje

Klasy

ID	Nazwa klasy	Wychowawca	Edycja danych	Dodaj zdjęcie	Usuń
1	1A	Michał Galas		(1)	
2	1B	Anna Komin		(1)	
3	2C	Marcelina Jeż		(1)	
4	3B	Jan Nowak		(0)	
5	4A	Ewa Komin		(0)	


Lekcja 11 – Interdyscyplinarny projekt

Obsługa błędów – zabezpieczenie uniemożliwiające zmianę nazwy klasy, na klasę która już istnieje.

Propozycja wykonania – widok klasy z uczniami

[Start](#)
[Klasy](#)
[Uczniowie](#)


Proszę wybrać klasę z listy:

1A - Michał Galas
1A - Michał Galas
1B - Anna Komin
2C - Marcelina Jeż
3B - Jan Nowak
4A - Ewa Komin

Wyswietl

[Start](#)
[Klasy](#)
[Uczniowie](#)

© 2014


Lekcja 11 – Interdyscyplinarny projekt

Widok klasy z uczniami – formularz do wyboru klasy, którą chcemy wyświetlić.

Slajd
11


B2E
BUSINESS TO EDUCATION


SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Propozycja wykonania – widok klasy z uczniami

Klasa 1A, wychowawca: Michał Galas



ID	Imię	Nazwisko	Tel	Email	Edycja danych	Dodaj zdjęcie	Usuń	Szczegóły
1	Alojzy	Bąbek	534534543	alozjy@o2.pl		(1)		
2	Michał	Nowak	6543456	nowak@gmail.com		(1)		
3	Ewa	Nowak	23452345	ewa@gmail.com		(0)		


DAILY
GROUP


Lekcja 11 – Interdyscyplinarny projekt


Widok klasy z uczniami – po wybraniu klasy, widać informację o klasie, zdjęcie wychowawcy oraz listę uczniów w klasie.

Podobnie jak dla nauczyciela, dla konkretnych uczniów mamy dostępne opcje – edycję, dodanie zdjęcia, usunięcie oraz podgląd konkretnego ucznia.

Człowiek - najlepsza inwestycja

Slajd
12

**B2E**
BUSINESS TO EDUCATION

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY


PHP
Hypertext Preprocessor


DAILY GROUP

Lekcja 11 – Interdyscyplinarny projekt

Widok klasy z uczniami – po kliknięciu na „Szczegóły” dla konkretnego ucznia uruchamiamy podgląd pojedynczego ucznia.

Slajd
13

**B2E**
BUSINESS TO EDUCATION

SZCZECIŃSKI PARK
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Imię:

Nazwisko:

Telefon:

Email:

Dodaj ucznia

Edycja ucznia:

Imię: Alojzy

Nazwisko: Bąbek

Telefon: 534534543

Email: alojzy@o2.pl

Edytuj ucznia

DAILY GROUP

Lekcja 11 – Interdyscyplinarny projekt

Widok klasy z uczniami – edycja i dodawanie.

Człowiek - najlepsza inwestycja





KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Slajd
14


B2E
 BUSINESS TO EDUCATION




PHP
 Hypertext Preprocessor

Propozycja wykonania – REST API

Do komunikacji serwera z aplikacją mobilną napisaną w Javie na telefony z Androidem wymagane jest API. Jest ona warstwą pośrednią pomiędzy klientem (smartphone) a bazą danych.


Stworzona propozycja umożliwia następujące operacje:

- /shool/class – wylistowanie klas w szkole
- /shool/class/:id – wylistowanie informacji o klasie oraz wszystkich jej uczniów, gdzie :id to identyfikator klasy w bazie
- /person/:id – wylistowanie informacji o uczniu, gdzie :id to identyfikator ucznia w bazie


 Lekcja 11 – Interdyscyplinarny projekt

Slajd
15


B2E
 BUSINESS TO EDUCATION




PHP
 Hypertext Preprocessor

REST API – przykładowa odpowiedź

Odpowiedź API po wywołaniu <http://localhost/rejestr/api/person/1>

```

{
  error: false
  _id: 1
  imie: "Alojzy"
  nazwisko: "Bąbek"
  tel: "534534543 "
  email: "alozjy@o2.pl"
  zdjecie: "http://localhost/rejestr/upload/uczen3.jpg"
}
    
```


 Lekcja 11 – Interdyscyplinarny projekt

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
 NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
 EUROPEJSKI
 FUNDUSZ SPOŁECZNY





Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

<p>Slajd 16</p>	  <p>PHP Hypertext Preprocessor</p> <h3>Propozycje rozbudowy projektu w nawiązaniu do tematyki poszczególnych lekcji</h3> <p>Lekcja 1 - Bezpieczeństwo aplikacji – walidacja formularzy po stronie serwera</p> <p>Lekcja 2 - Architektura usług sieciowych - rozbudowa REST API</p> <p>Lekcja 3 - Protokół HTTP - implementacja dodatkowych metod REST API jak PUT, DELETE - wykorzystanie w praktyce</p>  <p>Lekcja 11 – Interdyscyplinarny projekt</p>	
<p>Slajd 17</p>	  <p>PHP Hypertext Preprocessor</p> <h3>Propozycje rozbudowy projektu w nawiązaniu do tematyki poszczególnych lekcji</h3> <p>Lekcja 4 - Modyfikacja nagłówków HTTP - redirect przy logowaniu/wylogowaniu do aplikacji, modyfikacja strony kodowej wywołań do REST API</p> <p>Lekcja 5 - Cookie - informacje o użytkowniku - śledzenie aktywności użytkownika rejestru uczniów</p> <p>Lekcja 6 - Sesje - współdzielenie informacji - logowanie do aplikacji</p>  <p>Lekcja 11 – Interdyscyplinarny projekt</p>	
<p>Slajd 18</p>	  <p>PHP Hypertext Preprocessor</p> <h3>Propozycje rozbudowy projektu w nawiązaniu do tematyki poszczególnych lekcji</h3> <p>Lekcja 7 - Poczta - systemy SMTP - dodanie możliwości wysyłania wiadomości e-mail z aplikacji (znamy adres email ucznia, mail wysyłany przy pomocy kodu z aplikacji, a nie otwarcie okienka aplikacji skojarzonej z pocztą)</p> <p>Lekcja 8 - MIME - rozszerzenia multimedialne - wysyłanie zdjęcia</p> <p>Lekcja 9 - UML - wyrysowanie projektu w UMLu</p> <p>Lekcja 10 - Dobre praktyki programowania - biblioteki (mail, simple REST API), formatowanie kodu, komentarze</p>  <p>Lekcja 11 – Interdyscyplinarny projekt</p>	

Człowiek - najlepsza inwestycja



Slajd
19

 **B2E**
BUSINESS TO EDUCATION
  **SZCZECIŃSKI PARK**
NAUKOWO-TECHNOLOGICZNY

PHP
Hypertext Preprocessor

Podsumowanie

- Gotowa prosta aplikacja do gromadzenia informacji na temat uczniów i klas
- Idealna „baza” do testowania, wprowadzania poprawek i nauki na działającym przykładzie

 **KAPITAŁ LUDZKI**
NARODOWA STRATEGIA SPÓJNOŚCI
  **UNIA EUROPEJSKA**
EUROPEJSKI FUNDUSZ SPOŁECZNY

Człowiek - najlepsza inwestycja

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego


 **DAILY**
G R O U P
 Lekcja 11 – Interdyscyplinarny projekt

Tabela podsumowująca

10.11.3 Opis założonych osiągnięć ucznia

Po tej lekcji uczniowie przygotowują dla siebie ich pierwszy projekt informatyczny – książkę teleadresowej kolegów i koleżanek z klasy i ze szkoły dostępną poprzez WWW oraz aplikację mobilną.

Człowiek - najlepsza inwestycja