

INFORMATYKA

– MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA

PROGRAM NAUCZANIA INFORMATYKI Z ELEMENTAMI
PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH

Informatyka – poziom rozszerzony

Serwis WWW w projekcie edukacyjnym
– prowadzenie obliczeń i prezentacja informacji

Piotr Kopciał

$$\sum_{i=1}^n$$

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



WARSZAWSKA
WYŻSZA SZKOŁA
INFORMATYKI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Tytuł: ***Serwis WWW w projekcie edukacyjnym
– prowadzenie obliczeń i prezentacja informacji***

Autor: ***Piotr Kopciał***

Redaktor merytoryczny: ***prof. dr hab. Maciej M. Sysło***

Materiał dydaktyczny opracowany w ramach projektu edukacyjnego
***Informatyka – mój sposób na poznanie i opisanie świata.
Program nauczania informatyki z elementami przedmiotów
matematyczno-przyrodniczych***

www.info-plus.wysi.edu.pl

infoplus@wysi.edu.pl

Wydawca: Warszawska Wyższa Szkoła Informatyki
ul. Lewartowskiego 17, 00-169 Warszawa
www.wysi.edu.pl
rektorat@wysi.edu.pl

Projekt graficzny: *Marzena Kamasa*

Warszawa 2013

Copyright © Warszawska Wyższa Szkoła Informatyki 2013
Publikacja nie jest przeznaczona do sprzedaży

Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



WARSZAWSKA
WYŻSZA SZKOŁA
INFORMATYKI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY





SCENARIUSZ TEMATYCZNY

SERWIS WWW W PROJEKCIE EDUKACYJNYM – PROWADZENIE OBLICZEŃ I PREZENTACJA INFORMACJI

→ INFORMATYKA – POZIOM ROZSZERZONY

**OPRACOWANY W RAMACH PROJEKTU:
INFORMATYKA – MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA.
PROGRAM NAUCZANIA INFORMATYKI
Z ELEMENTAMI PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH**

Streszczenie

Internet wkracza w coraz to nowe obszary naszego życia: e-nauczanie, elektroniczne biblioteki, wirtualne laboratoria, medycyna, usługi (bankowość, turystyka). Podstawowym elementem tych i podobnych serwisów są strony internetowe, które składają się na bardziej złożone witryny, portale i platformy internetowe. Niniejszy scenariusz jest poświęcony tworzeniu i funkcjonowaniu serwisów internetowych. W pierwszej części opisane są mechanizmy działania stron internetowych, w tym m.in. komunikacja w standardzie klient-serwer i strony dynamiczne. Omawiane są również zalety i wady stron statycznych i dynamicznych. Jednym z celów jest uwrażliwienie ucznia na dobre praktyki projektowania i tworzenia stron internetowych i przygotowanie do tworzenia własnych stron.

W części praktycznej uczniowie poznają język opisu struktury strony – HTML – oraz ściśle z nim związany język CSS, odpowiedzialny za prezentację treści na stronie. Uczą się, jak korzystać z edytora stron – programu do tworzenia stron internetowych. Tematyka zaproponowanych lekcji obejmuje również metody i techniki tworzenia dynamicznych stron internetowych. Uczniowie poznają podstawy języka PHP oraz SQL, a także sposób ich współdziałania w praktycznych przykładach. Zajęcia obejmują również instalowanie i konfigurowanie serwera WWW Apache oraz relacyjnej bazy danych MySQL, a także obsługę środowiska programistycznego, służącego do tworzenia dynamicznych stron internetowych.

Wiedzę i umiejętności zdobyte w trakcie realizacji ćwiczeń, uczniowie zweryfikują projektując własny dynamiczny serwis sieciowy w technologii klient-serwer. Serwis ten poza stroną wizualną, zawierać będzie funkcje takie jak: obsługa interaktywnych formularzy użytkownika, przetwarzanie danych wprowadzonych przez użytkownika i inne. Materiał służy opanowaniu umiejętności tworzenia atrakcyjnych, pełnowartościowych serwisów internetowych, a efektem końcowym ma być serwis internetowy utworzony samodzielnie przez każdego uczestnika.

Czas realizacji

4 x 45 minut

Tematy lekcji

1. Witryna w Internecie – zasady tworzenia i funkcjonowania (1 x 45 minut)
2. Tworzenie własnego serwisu WWW (3 x 45 minut)

LEKCJA NR 1

TEMAT: Witryna w Internecie – zasady tworzenia i funkcjonowania

Streszczenie

Lekcja obejmuje następujące zagadnienia:

- Strona, witryna, serwis, portal, platforma
- Język HTML i struktura dokumentu HTML
- Zasada działania strony internetowej
- Dynamiczna strona internetowa i zasady jej działania
- Zalety i wady stron statycznych i dynamicznych
- Tworzenie stron dynamicznych – język skryptowy
- Narzędzia do tworzenia stron
- Ćwiczenia z zasad tworzenia i funkcjonowania stron WWW

Podstawa programowa

Etap edukacyjny: IV, przedmiot: informatyka (poziom rozszerzony)

Cele kształcenia – wymagania ogólne

IV. Wykorzystanie komputera oraz programów i gier edukacyjnych do poszerzania wiedzy i umiejętności z różnych dziedzin i rozwijania zainteresowań.

Treści nauczania – wymagania szczegółowe

1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej. Uczeń:
 - 1) przedstawia sposoby reprezentowania różnych form informacji w komputerze: liczb, znaków, obrazów, animacji, dźwięków.
2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:
 - 4) znajduje odpowiednie informacje niezbędne do realizacji projektów z różnych dziedzin.

Etap edukacyjny: IV, przedmiot: chemia (poziom podstawowy i rozszerzony)

Cele kształcenia – wymagania ogólne

I. Wykorzystanie, przetwarzanie i tworzenie informacji.

Cel

Umiejętność rozróżniania pojęć: strona, witryna, serwis, portal, platforma internetowa. Znajomość mechanizmów działania stron statycznych i dynamicznych. Znajomość zalet i wad stron statycznych i dynamicznych.

Słowa kluczowe

adres URL, dokument HTML, dynamiczna strona internetowa, Internet, klient-serwer, PHP, platforma edukacyjna, protokół http, serwer, serwis WWW, skrypt, statyczne strony WWW, Web 2.0, Web 3.0, witryna internetowa



Co przygotować

- Prezentacja 1
- Film 1 – „Konfigurowanie przeglądarki i edytora”



MATERIAŁ TEORETYCZNY

Wprowadzenie

Pierwsze strony WWW, towarzyszące pojawieniu się Internetu na początku lat 90. XX wieku, były stosunkowo proste. Zawierały bloki tekstu uzupełnione ilustracjami. Sieć Internet służyła wtedy do przeglądania raportów i prac naukowych. Dokumenty tekstowe były oznakowane za pomocą znaczników języka HTML, dzięki czemu przeglądarka internetowa wiedziała, w jaki sposób je wyświetlać. Hiperłącza umożliwiające nawigację pomiędzy stronami, skuteczność dystrybucji danych oraz coraz częściej pojawiające się treści multimedialne przyczyniły się do szybkiego rozpowszechnienia Internetu. Podstawowy język tworzenia stron internetowych, HTML, zaczęto wspierać dodatkowymi technologiami, które umożliwiały wzbogacanie stron WWW o nowe elementy.

Technologie internetowe w ostatnich latach znacznie ewoluowały. Dziś istnieje wiele portali internetowych, ułatwiających użytkownikowi zalogowanie się do systemu i dostęp do bogatych zasobów, takich jak poczta elektroniczna, własna galeria zdjęć, wirtualny dziennik, interaktywny kalendarz, najnowsze informacje na wybrany przez użytkownika temat. Użytkownik może dostosowywać sposób prezentowania informacji według własnych potrzeb i upodobań.

Internet pełni obecnie inną rolę niż na początku jego istnienia – bardziej komercyjną. Nadzieje i oczekiwania w nim pokładane przyciągają inwestorów, angażujących spory kapitał. Globalna sieć stała się miejscem prowadzenia interesów. W początkowej fazie były to aukcje internetowe. Dziś funkcjonują wyspecjalizowane sklepy (np. e-apteki), banki internetowe. Szacuje się, że e-biznes stanie się główną siłą ekonomii XXI wieku. Handel elektroniczny (tzw. e-handel) postrzegany jest obecnie jako jeden z najważniejszych aspektów sieci WWW.

Interaktywne serwisy WWW znajdują coraz szersze zastosowanie. Przykładem może być edukacja internetowa. Studenci mogą odbywać zajęcia o dowolnej porze dnia, w dowolnym dniu tygodnia. Platforma edukacyjna to połączenie strony internetowej, poczty elektronicznej, narzędzi multimedialnych i programów do nauczania oraz narzędzi wzajemnej komunikacji pomiędzy użytkownikami. Dane użytkowników (np. oceny z przedmiotów) zapisane w bazie danych składają się na system kontroli postępów w przyswajaniu wiedzy.

Internet to skarbnica informacji, stanowiących bazę dla wiedzy jego użytkowników, której wykorzystanie nie byłoby możliwe bez nowoczesnych technik indeksowania, katalogowania, selekcjonowania i prezentowania dostępnych informacji. Technologie, takie jak PHP, umożliwiają tworzenie dynamicznych serwisów WWW, które coraz częściej są połączone z bazami danych, np. z MySQL.

Wraz z rozpowszechnieniem Internetu pojawiły się narzędzia lepiej integrujące użytkowników. Mogą oni umieszczać własne treści w Internecie, komentować swoje wypowiedzi, pisać dzienniki (tzw. blogi), wyrażać własne opinie (np. na forach dyskusyjnych). Dzisiejsze strony WWW cechuje akcja i interaktywność, co jest zasługą nowoczesnych technologii tworzenia dynamicznych serwisów WWW.

Umiejętność tworzenia serwisów internetowych, jak również ich świadomej (wiedza „jak to działa”) obsługi jest obecnie niezwykle istotna.

Strona, witryna, serwis, portal, platforma

Strona internetowa jest wynikiem interpretacji **dokumentu HTML**, czyli dokumentu napisanego w języku HTML. Taki dokument może być pobrany z lokalnego dysku komputera lub z serwera internetowego i jest interpretowany po stronie użytkownika przez **przeglądarkę**. Na stronie internetowej można umieszczać tekst, obrazy, tabele, wstawki dźwiękowe, animacje, sekwencje wideo.

Często słyszymy określenie „witryna internetowa”. **Witryna internetowa** jest określeniem rozbudowanej strony internetowej, która może składać się w wielu stron, do których dostęp uzyskujemy poprzez wybranie odpowiedniej opcji w menu witryny. W dalszej części będziemy na ogół pisać o stronie, bo witryna to zbiór stron.

Pojęcie **serwis internetowy** odnosi się do witryny współpracującej z bazą danych, dzięki czemu jest możliwe przechowywanie np. informacji o użytkownikach i obsługa ich logowania.

Natomiast **portal internetowy** to rozbudowany serwis, oferujący użytkownikom dodatkowe mechanizmy takie jak np. forum dyskusyjne, chat, konto pocztowe e-mail, blog itp.

Coraz częściej stosowany jest również termin **platforma internetowa**. Tym mianem określa się najczęściej rozbudowane portale internetowe, przeznaczone dla wydzielonej grupy (społeczności) użytkowników, np. w firmie (w korporacji) czy w szkole. W tym ostatnim przypadku są to tzw. **platformy edukacyjne**. Po zalogowaniu do platformy użytkownicy mają do dyspozycji wiele dodatkowych narzędzi takich jak np. poczta wewnętrzna, aplikacja do konsultacji on-line, współdzielone pliki, dokumenty, kalendarze, miejsce na własne zasoby (pliki).

Internet w dzisiejszej postaci określa się mianem sieci **Web 2.0**, odnoszącym się do serwisów internetowych, w których podstawową rolę odgrywa użytkownik wraz z generowanymi przez siebie treściami, zasobami, a także serwisami.

A co czeka nas w przyszłości? Zapewne **Web 3.0** – dalsza ewolucja Internetu w kierunku systemu przekazu wiedzy i modelu sieci semantycznej, czyli sieci „rozumiejącej” swoją zawartość oraz użytkowników sieci.

Język HTML i struktura dokumentu HTML

Opis stron internetowych jest tworzony w języku HTML. **HTML** (ang. *Hypertext Markup Language*) służy do tworzenia opisów stron internetowych. Język HTML to zestaw znaczników, pomiędzy którymi umieszcza się tekst lub inne elementy mające pojawić się na stronie. Każdy znacznik umieszczony jest w nawiasach ostrych < >.

Przykładowo – znacznikiem rozpoczęcia opisu strony jest <HTML>. Większość znaczników występuje jako część otwierająca i zamykająca. Część zamykająca zawiera dodatkowy znak – ukośnik /. Znacznikiem zamykającym stronę jest zatem </HTML>.

Strukturę dokumentu HTML opisującego stronę określają 3 znaczniki: <HTML>, <HEAD> i <BODY>.

<HTML> – użycie tego znacznika jest obowiązkowe, gdyż wskazuje on na początek i koniec dokumentu. Znacznik <HTML> musi znaleźć się w pierwszym wierszu kodu strony.

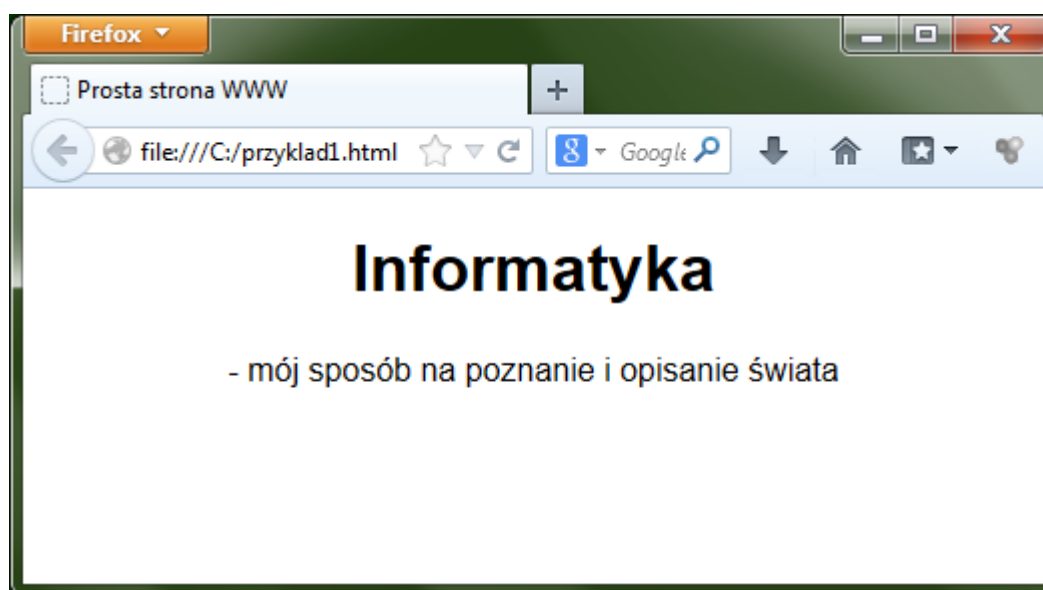
<HEAD> – znacznik definiujący nagłówek dokumentu. Można w nim określić takie elementy, jak nazwa i styl dokumentu, tytuł strony. Nagłówek umieszczamy na początku dokumentu, a kończymy go znacznikiem </HEAD>.

<BODY> – pomiędzy znacznikami <BODY> oraz </BODY> zawarta jest zasadnicza treść dokumentu. W tej części można definiować: rodzaj czcionki, kolor tekstu, tło strony itd.

Przykład prostego dokumentu HTML przedstawiono poniżej, na rysunku 1 zaprezentowano efekt jego interpretacji przez przeglądarkę.

```
<HTML>
<HEAD>
<TITLE> Prosta strona WWW </TITLE>
</HEAD>
<BODY>
<FONT FACE=„Arial”>
<CENTER>
<H1> Informatyka </H1>
– mój sposób na poznanie i opisanie świata
```

```
</CENTER>
</BODY>
</HTML>
```



Rysunek 1. Prosta strona internetowa

Serwer to komputer, na którym znajduje się plik zawierający opis strony internetowej utworzonej w języku HTML wraz z plikami zawierającymi elementy składowe strony (np. obrazy). Serwer udostępnia stronę innym komputerom za pośrednictwem sieci Internet. W sieci Internet istnieje wiele serwerów.

Przeglądarka to program służący do pobierania opisu stron internetowych z serwera i wyświetlania ich zawartości na ekranie monitora użytkownika. Przeglądarka tłumaczy kod HTML strony na postać oglądaną na ekranie.

Adres URL (ang. *Uniform Access Locator*) to adres, pod którym jest dostępna konkretna strona internetowa. Przykładowy adres URL to <http://www.google.pl/>. Adres URL jest adresem serwera, z którym przeglądarka kontaktuje się w celu pobrania opisu strony.

Znaczenie poszczególnych części adresu URL zestawiono w tabeli 1.

Tabela 1. Znaczenie poszczególnych części adresu URL

http:// (https://)	nazwa_serwera.pl/	katalog/	plik.html
nazwa protokołu sieciowego (sposobu przesyłania danych z serwera do przeglądarki)	nazwa domenowa serwera, z którego zostanie pobrany dokument HTML (wyświetlona jako strona)	nazwa folderu (katalogu) na serwerze	nazwa pobieranego pliku (dokumentu HTML) znajdującego się w tym folderze (katalogu)

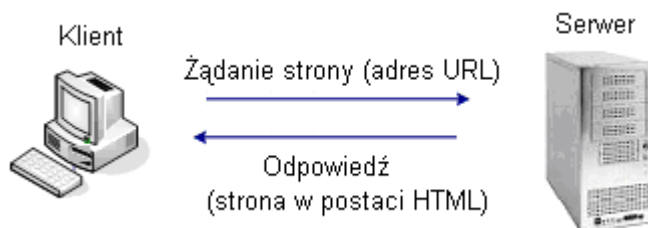
Zasada działania strony internetowej

Po utworzeniu, strona internetowa umieszczana jest na serwerze. W tym momencie staje się dostępna dla wszystkich użytkowników Internetu. Tak jak budynki na ulicy, każdy serwer ma swój adres (tzw. adres domenowy); a tak jak mieszkania w budynku – każda strona ma swój unikatowy adres.

Gdy użytkownik wpisze adres URL strony w przeglądarce, ta stara się odnaleźć w pierwszej kolejności serwer, a następnie daną stronę. Jeśli znajdzie, serwer odsyła do przeglądarki żadaną stronę w postaci pliku HTML, ewentualnie wraz z uzupełniającymi go plikami graficznymi. Przeglądarka wyświetla stronę na ekranie komputera użytkownika w postaci zdefiniowanej w pliku HTML.

Aby komputer użytkownika (a dokładniej jego przeglądarka) mógł się porozumieć z serwerem, obydwa komputery komunikują się za pomocą protokołu HTTP (ang. *Hypertext Transfer Protocol*).

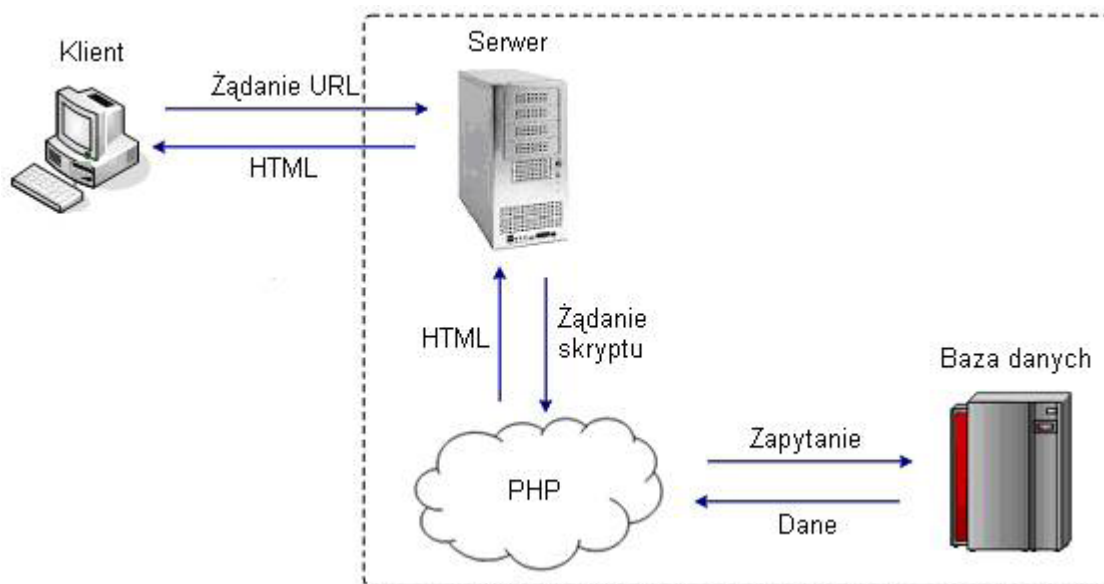
Taką komunikację nazywamy komunikacją **klient-serwer** (rys. 2). **Klientem** w tym określeniu jest komputer użytkownika, który przy użyciu przeglądarki żąda wyświetlenia wskazanej strony, której opis znajduje się na **serwerze**.



Rysunek 2. Komunikacja klient-serwer

Dynamiczna strona internetowa i zasady jej działania

Dzięki dynamicznym stronom internetowym można np. witać użytkownika odwiedzającego stronę ponownie w następujący sposób: „Witaj ponownie, Krzysiu!”. **Dynamiczna strona** internetowa jest tworzona przez serwer w momencie, kiedy użytkownik żąda jej wyświetlenia. Strony dynamiczne są generowane na bieżąco i mogą zawierać różne treści, w zależności od tego, kto je pobiera i w jakich okolicznościach. Na przykład tło strony może być jasne lub ciemne, w zależności od tego, czy akurat jest dzień czy noc.



Rysunek 3. Działanie dynamicznej strony WWW

Na rysunku 3 przedstawiono działanie dynamicznej strony WWW. Interakcja pomiędzy klientem a serwerem zaczyna się w momencie wpisania w przeglądarce adresu strony lub kliknięcia łącza do strony dynamicznej. Za pomocą protokołu HTTP przeglądarka nawiązuje połączenie z serwerem. Serwer przesyła żądanie do interpretera języka skryptowego (np. PHP), który wykonuje kod skryptu – **skrypitem** nazywamy



kod napisany w języku przeznaczonym do tworzenia stron dynamicznych. Jeśli w skrypcie PHP są zapisane zapytania do bazy danych (np. w celu pobrania informacji o użytkowniku), interpreter języka skryptowego odpowiada za komunikację serwera z bazą danych. Po pobraniu zawartości strony, przeglądarka analizuje kod HTML, po czym wyświetla gotową stronę na ekranie monitora użytkownika.

Należy zwrócić uwagę, że dynamiczne fragmenty strony internetowej nie istnieją, dopóki ktoś nie zażąda wyświetlenia strony. Dopiero wtedy serwer buduje taką stronę, według instrukcji zawartych w kodzie HTML oraz w kodzie skryptu, a gdy użytkownik zamyka stronę dynamiczną w przeglądarce, to dynamiczne fragmenty strony przestają istnieć. W przypadku kolejnego wyświetlenia takiej strony, jej dynamiczne fragmenty są tworzone na nowo. Dzięki temu na stronach mogą ulegać zmianie: godzina, data, prognoza pogody, program telewizyjny itp.

W odróżnieniu od strony dynamicznej, treść strony statycznej nie zmienia się od momentu jej utworzenia do chwili zmiany opisu strony lub usunięcia go z serwera.

Zalety i wady stron statycznych i dynamicznych

Statyczne strony WWW opisane w języku HTML, są przechowywane na serwerze i przesyłane są w takiej samej postaci do wszystkich użytkowników. Oznacza to, że każdy użytkownik widzi taką samą stronę pod względem treści i układu. Natomiast strony dynamiczne są generowane przez serwer na bieżąco, w zależności od tego kim jest użytkownik (np. użytkownik zalogowany do serwisu ma dostęp do treści niedostępnych dla użytkowników niezalogowanych). Mechanizm ten wymaga od serwera większej pracy, aniżeli w przypadku stron statycznych, kiedy to rola serwera sprowadza się do przechowywania plików, oczekiwania na żądanie i przesłania strony wskazanej przed użytkownika do jego przeglądarki.

Ponadto potrzebna jest baza danych zawierająca treści, które mają pojawić się na stronie. **Baza danych** jest elektronicznym magazynem informacji (danych) i narzędziem do zarządzania tymi informacjami.

Zarówno strony statyczne, jak i strony dynamiczne mają swoje wady i zalety, co zilustrowano w tabeli 2.

Tabela 2. Wady i zalety stron statycznych i dynamicznych

	Wady	Zalety
Strony statyczne	<ul style="list-style-type: none">– nie można szybko zmienić treści– interakcja z użytkownikiem bardzo ograniczona	<ul style="list-style-type: none">– łatwo je utworzyć (kod HTML)
Strony dynamiczne	<ul style="list-style-type: none">– trudniej je utworzyć (języki skryptowe są trudniejsze do opanowania niż HTML)– wymagają bazy danych na serwerze	<ul style="list-style-type: none">– łatwo i szybko można zmienić treść– umożliwiają interakcję z użytkownikiem

Statyczne strony WWW, nawet te najbardziej atrakcyjne pod względem treści i grafiki, mają wadę, która polega na tym, że aktualizacja ich treści zajmuje sporo czasu, ponieważ wymaga modyfikowania każdej strony. Wady tej pozbawione są witryny z elementami dynamicznymi, których treść przechowywana jest w bazie danych i pobierana przy każdym otwarciu strony przez odwiedzającego. Ponadto zmiana treści dynamicznego fragmentu strony wymaga zmiany w jednym tylko miejscu – w bazie danych.

Tworzenie stron dynamicznych – język skryptowy

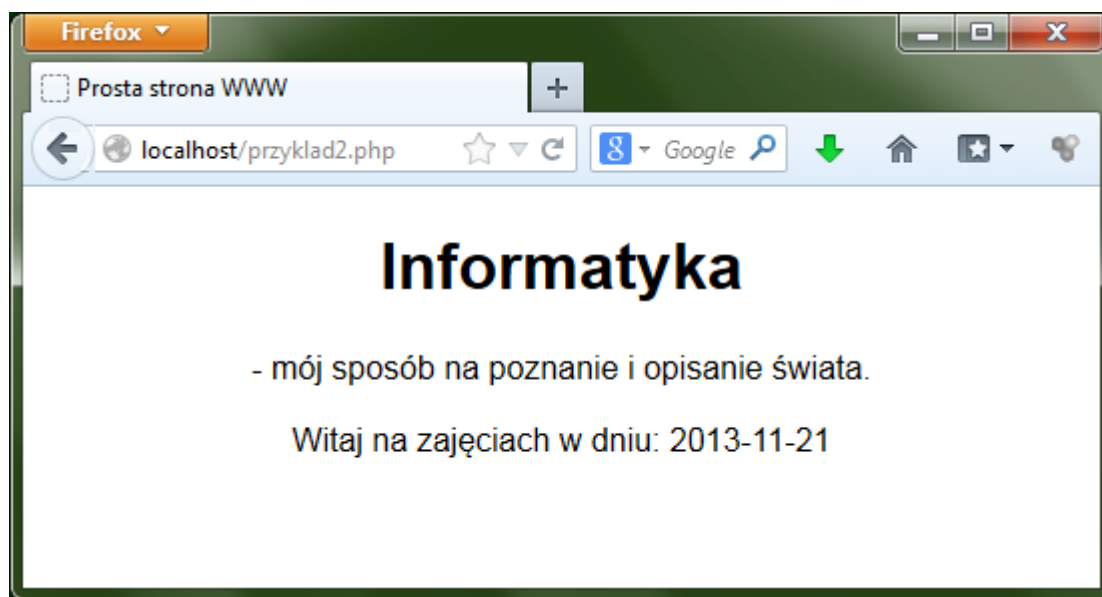
Dynamiczne strony internetowe tworzy się za pomocą tzw. **języków skryptowych**. Fragmenty kodu napisane w języku skryptowym są umieszczane pomiędzy znacznikami kodu HTML strony. W języku skryptowym definiuje się polecenia dla serwera, w jaki sposób ma budować (generować) stronę. Można np. wyświetlić aktualną datę i godzinę lub pobrać najświeższe informacje (np. na temat pogody) z bazy danych. Najczęściej stosowanym i najprostszym do nauki językiem skryptowym jest PHP. Poniżej przedstawiono kod skryptu generującego aktualną datę.

```

<HTML>
<HEAD>
<TITLE> Prosta strona WWW </TITLE>
</HEAD>
<BODY>
<FONT FACE=„Arial“>
<CENTER>
<H1> Informatyka </H1>
– mój sposób na poznanie i opisanie świata.
<p>
Witaj na zajęciach w dniu:
<?php
echo date(„Y-m-d“);
?>
</CENTER>
</BODY>
</HTML>

```

Efekt działania tego skryptu jest pokazany na rysunku 4 – za każdym razem, gdy strona jest wyświetlana, pobierana jest aktualna data.



Rysunek 4. Strona wyświetlająca aktualną datę

Narzędzia do tworzenia stron

Kod HTML można pisać w prostym edytorze tekstu, np. w Notatniku Windows. Wystarczy znać znaczniki HTML i zasady ich stosowania. Jednakże dużym ułatwieniem jest posłużenie się specjalnym programem do tworzenia stron internetowych, tzw. edytorem języka HTML. Jest to pomocne przy tworzeniu bardziej złożonych elementów stron, takich jak np. tabele. Zamiast tworzyć w Notatniku każdą komórkę tabeli osobno, wystarczy skorzystać z narzędzia tworzenia tabel w takim edytorze.

Przykładem edytora HTML jest program FrontPage firmy Microsoft. Istnieje również wiele darmowych edytorów (lub dostępnych za darmo przez określoną liczbę dni, np. 60), które można pobrać z Internetu.



Darmowe edytory HTML można pobrać z następujących stron: <http://notepad-plus-plus.org/download/v6.1.6.html> (Notepad++) <http://agerwebedytor.com/> (Ager Web Edytor), http://www.darmoweprogramy.org/programy/edytory_html.php (Alleycode HTML Editor, Web Design Toy, EasyHTML, HotHTML).

Rozwinięcie zagadnień poruszanych w niniejszym scenariuszu stanowi kurs pt. „Tworzenie serwisów internetowych”, dostępny w serwisie IT Szkoła: <http://it-szkola.edu.pl/kkurs,kurs,2>.

Literatura

1. Cohen J., *Serwisy WWW. Projektowanie, tworzenie, zarządzanie*, Helion, Gliwice 2004
2. Meloni J.C., *PHP, MySQL i Apache dla każdego*, Helion, Gliwice 2007
3. Price J., Price L., *Profesjonalny serwis WWW*, Helion, Gliwice 2002
4. Sokół R., *Internet. Ilustrowany przewodnik*, Helion, Gliwice 2007
5. Ullman L., *PHP i MySQL. Dynamiczne strony WWW*, Helion, Gliwice 2004
6. Welling L., Thomson L., *PHP i MySQL. Vademecum profesjonalisty*, Helion, Gliwice 2005

Przebieg zajęć

Wprowadzenie (15 minut)

Omówienie wprowadzenia teoretycznego do niniejszej lekcji, przy użyciu przygotowanej prezentacji.

Praca indywidualna lub w zespołach (15 minut)

Praca indywidualna lub w zespołach dwuosobowych. Uczniowie wykonują ćwiczenia, korzystając w razie potrzeby z treści wprowadzenia teoretycznego do niniejszej lekcji.

Panel ekspertów (10 minut)

Omówienie rezultatów pracy – efektów wykonania ćwiczeń.

Dyskusja podsumowująca (5 minut)

Utrwalenie:

- umiejętności rozróżniania pojęć: strona, witryna, serwis, portal, platforma internetowa,
- znajomości mechanizmów działania stron statycznych i dynamicznych,
- znajomości zalet i wad stron statycznych i dynamicznych.

Sprawdzenie wiedzy

Ćwiczenia z zasad tworzenia i funkcjonowania stron WWW

➡ Ćwiczenie 1.1.

Znajdź trzy przykłady statycznych stron/ witryn w Internecie i podaj ich adresy.

➡ Ćwiczenie 1.2.

Znajdź trzy przykłady dynamicznych stron/ witryn w Internecie i podaj ich adresy.

Dla każdej ze znalezionych stron wypisz elementy dynamiczne, które się na niej znajdują.

Ocenianie

Ćwiczenie 1.1

- ocena liczby znalezionych stron/ witryn statycznych
- ocena efektów pracy – czy znalezione strony/witryny rzeczywiście nie zawierają elementów dynamicznych

Ćwiczenie 1.2

- ocena liczby znalezionych stron/ witryn dynamicznych
- ocena liczby zidentyfikowanych elementów dynamicznych na znalezionych stronach

Zaliczenie testu wiedzy w przypadku co najmniej połowy poprawnych odpowiedzi.

Dostępne pliki



1. Prezentacja 1
2. Ćwiczenia (Zadania 1)
3. Film 1



LEKCJA NR 2, 3, 4

TEMAT: Tworzenie własnego serwisu WWW

Streszczenie

Treść lekcji zawiera następujące zagadnienia:

- Tworzenie stron statycznych do prezentowania informacji
 - treść – język HTML
 - wygląd – język CSS
- Tworzenie stron dynamicznych do prowadzenia obliczeń
 - sposób działania – język PHP
 - instalowanie i konfigurowanie serwera WWW i interpretera języka PHP
- Ćwiczenia z tworzenia własnego serwisu WWW

Podstawa programowa

Etap edukacyjny: IV, przedmiot: informatyka (poziom rozszerzony)

Cele kształcenia – wymagania ogólne

IV. Wykorzystanie komputera oraz programów i gier edukacyjnych do poszerzania wiedzy i umiejętności z różnych dziedzin i rozwijania zainteresowań.

Treści nauczania – wymagania szczegółowe

Poziom podstawowy:

1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej. Uczeń:
 - 1) przedstawia sposoby reprezentowania różnych form informacji w komputerze: liczb, znaków, obrazów, animacji, dźwięków.
2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:
 - 4) znajduje odpowiednie informacje niezbędne do realizacji projektów z różnych dziedzin.

Etap edukacyjny: IV, przedmiot: chemia (poziom podstawowy i rozszerzony)

Cele kształcenia – wymagania ogólne

I. Wykorzystanie, przetwarzanie i tworzenie informacji.

Cel

Uświadomienie uczniom możliwości nauki tworzenia stron WWW z wykorzystaniem bezpłatnych technologii PHP, MySQL i Apache oraz bezpłatnych szablonów witryn WWW. Nauka instalacji i konfiguracji serwera WWW. Nauka elementów języka PHP.

Słowa kluczowe

adres URL, baza danych, dokument HTML, dynamiczna strona internetowa, Internet, klient-serwer, MySQL, PHP, protokół http, serwer, serwis WWW, skrypt, statyczne strony WWW, witryna internetowa

Co przygotować



- Prezentacja – 2
- Pliki przygotowane do ćwiczeń do niniejszej lekcji (materiały pomocnicze)



- Film 2 – „Pobieranie i edycja szablonu witryny”
- Film 3 – „Instalowanie i konfigurowanie pakietu XAMPP”

MATERIAŁ TEORETYCZNY

Od czego zacząć?

Tworzenie skomplikowanego projektu najlepiej zacząć od rzeczy najłatwiejszych. W przypadku serwisu internetowego będzie to z pewnością jego treść, a w dalszej kolejności wygląd. Przy użyciu znaczników języka HTML, który jest stosunkowo łatwy do opanowania, możemy utworzyć poszczególne podstrony serwisu (strona główna, kontakt, pomoc, regulamin itp.). Wygląd tych stron możemy definiować przy użyciu kaskadowych arkuszy stylów – CSS. Natomiast najwięcej pracy pochłania zaprogramowanie działania poszczególnych podstron i powiązanie ich ze sobą.

Przy tworzeniu pojedynczych stron internetowych, ich struktury i treści, warto skorzystać z otwartych zasobów edukacyjnych. Wykłady i ćwiczenia udostępnione są pod adresem http://wazniak.mimuw.edu.pl/index.php?title=Aplikacje_WWW.

Treść – język HTML

Informacje na temat wykorzystania znaczników HTML są udostępnione pod adresem <http://wazniak.mimuw.edu.pl/index.php?title=AWWW-1st3.6-w01.tresc-1.0-toc>, ćwiczenia utrwalające wiedzę znajdziemy tutaj – <http://wazniak.mimuw.edu.pl/index.php?title=AWWW-1st3.6-l01.tresc-1.0>, natomiast praktyczny kurs języka HTML można przejrzeć na stronie <http://www.kurshtml.boo.pl/html/zielony.html>.

Język HTML jest wykorzystywany m.in. do tworzenia formularzy, które wypełnia użytkownik odwiedzający stronę WWW. Po wypełnieniu formularza wprowadzone dane zostają następnie przetworzone przez odpowiedni skrypt PHP. Kod strony prezentującej formularz może być następujący:

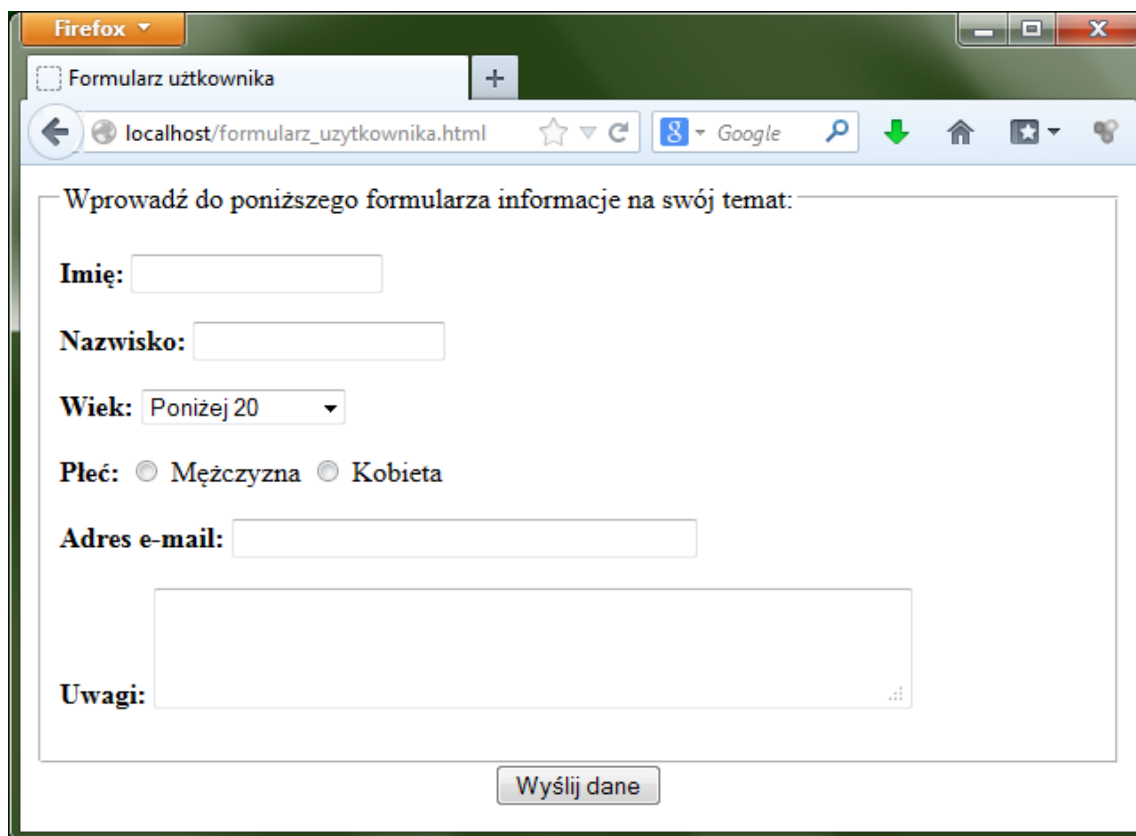
```
<head>
<title>Formularz użytkownika</title>
</head>
<body>
<form action=„obsługa_formularza.php” method=„post”>
<fieldset><legend>Wprowadź do poniższego formularza informacje na swój temat:</legend>
<p><b>Imię:</b> <input type=„text” name=„imie” size=„20” maxlength=„40” /></p>
<p><b>Nazwisko:</b> <input type=„text” name=„nazwisko” size=„20” maxlength=„40” /></p>
<p><b>Wiek:</b>
<select name=„wiek”>
<option value=„0-20”>Poniżej 20</option>
<option value=„20-40”>Między 20 a 40</option>
<option value=„40+”>Powyżej 40</option>
</select></p>
<p><b>Płeć:</b> <input type=„radio” name=„plec” value=„M” /> Mężczyzna <input type=„radio”
name=„plec” value=„K” /> Kobieta</p>
<p><b>Adres e-mail:</b> <input type=„text” name=„email” size=„40” maxlength=„60” /> </p>
<p><b>Uwagi:</b> <textarea name=„uwagi” rows=„3” cols=„50”></textarea></p>
```

```

</fieldset>
<div align=„center“><input type=„submit“ name=„submit“ value=„Wyślij dane“ /></div>
</form>
</body>

```

Efekt wyświetlenia formularza w przeglądarce powinien wyglądać jak na rysunku 5.



Rysunek 5. Formularz utworzony w języku HTML

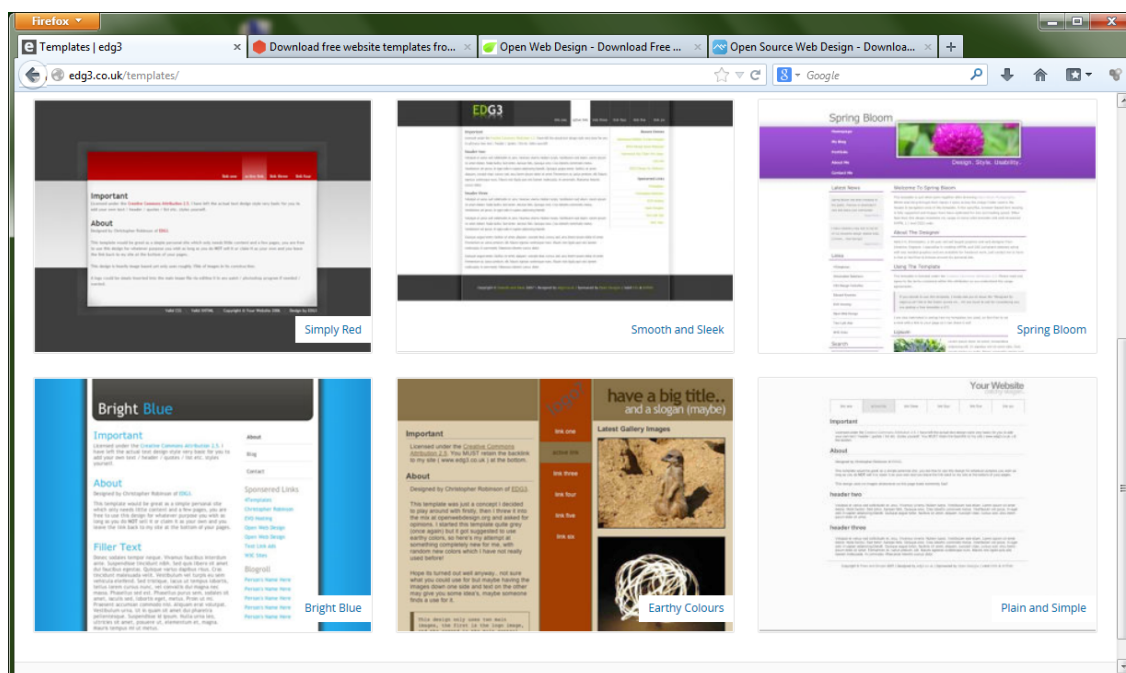
Wygląd – język CSS

Kaskadowe arkusze stylów (CSS) stanowią uzupełnienie języka HTML. Podczas gdy HTML definiuje strukturę i treść strony, CSS odpowiada za formatowanie jego wyglądu. Informacje na temat wykorzystania kaskadowych arkuszy stylów, uzupełnione o ćwiczenia można znaleźć pod adresami internetowymi przytoczonymi w poprzednim punkcie.

Aby nadać nowo tworzonej witrynie bardziej atrakcyjny wygląd, można posłużyć się gotowymi szablonami witryn, przygotowanymi w oparciu o arkusze stylów CSS.

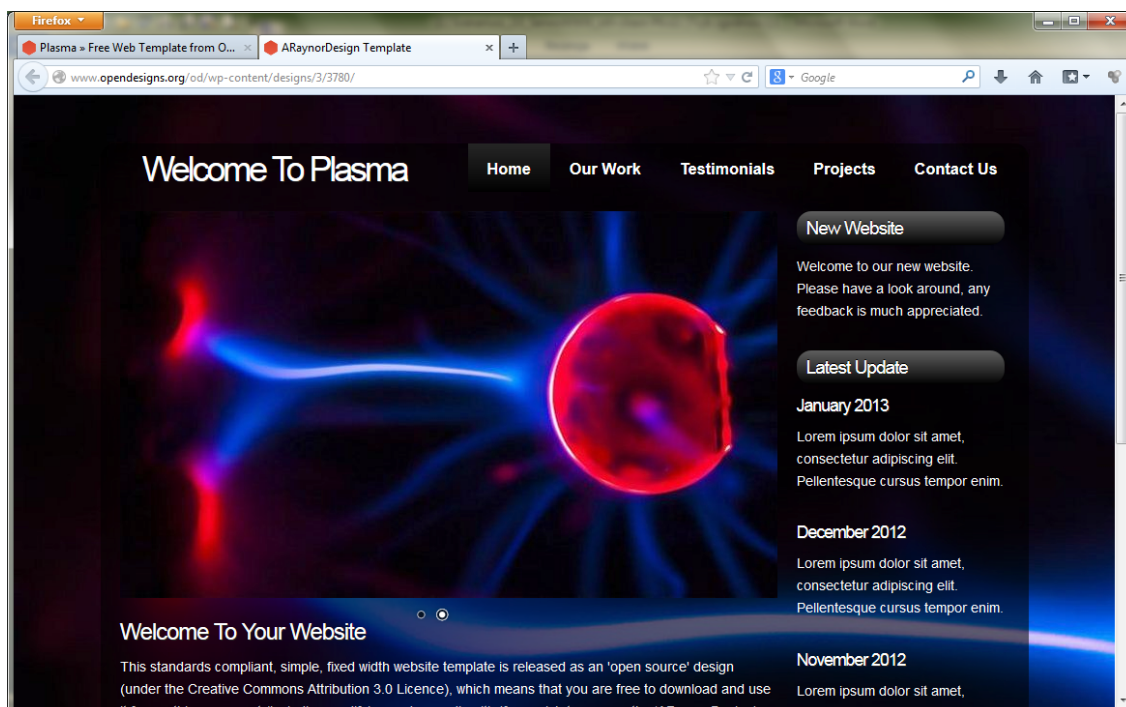
Do sformatowania wyglądu strony użyto arkuszy stylów CSS, zawartych w szablonach udostępnionych na stronie <http://www.edg3.co.uk/templates/> – rysunek 6. Więcej propozycji arkuszy stylów znajdziemy na stronach: <http://www.opendesigns.org/>, <http://www.openwebdesign.org/>, <http://www.oswd.org/>.

Warto podkreślić, że wiele szablonów dostępnych w Internecie jest bezpłatnych. Można je pobrać i zbudować na nich własną witrynę. Natomiast wszystkie szablony (również te płatne) mogą posłużyć jako doskonały materiał do nauki – możemy je pobrać, dowiedzieć się, jak zostały utworzone (analizując kod HTML i CSS) i wykorzystać tę wiedzę przy tworzeniu własnej witryny.



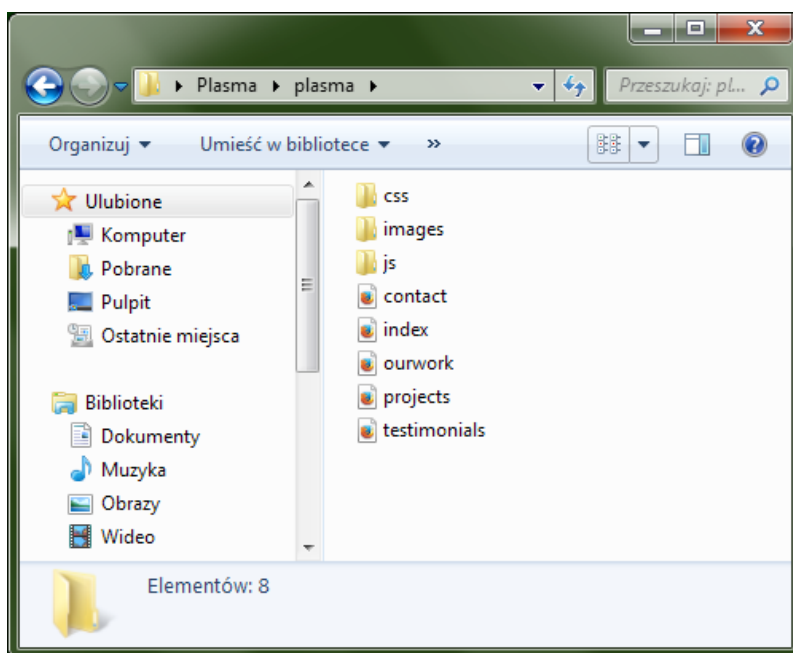
Rysunek 6. Szablony witryn WWW dostępne w Internecie

Interdyscyplinarny charakter niniejszego scenariusza polega na wyborze tematu serwisu internetowego – z dziedziny **chemii/ fizyki**. Tematem tym może być np. **rozpad promieniotwórczy**. Do wybranego tematu należy dostosować wygląd serwisu. W tym celu należy wyszukać odpowiedni szablon spośród tysięcy dostępnych wzorów – rysunek 7.



Rysunek 7. Szablon wybrany do witryny poświęconej projektowi edukacyjnemu pt. „Rozpad promieniotwórczy” [Zaprezentowany szablon jest dostępny w folderze ‚Pliki do zadań’ stanowiącym uzupełnienie niniejszego scenariusza. Szablon jest autorstwa ‚araynordesign’ i pochodzi ze strony <http://www.opendesigns.org/design/plasma/> i jest udostępniany na licencji Creative Commons. Z warunkami licencji można się zapoznać na stronie <http://www.opendesigns.org/help/> oraz na stronie <https://creativecommons.org/licenses/>]

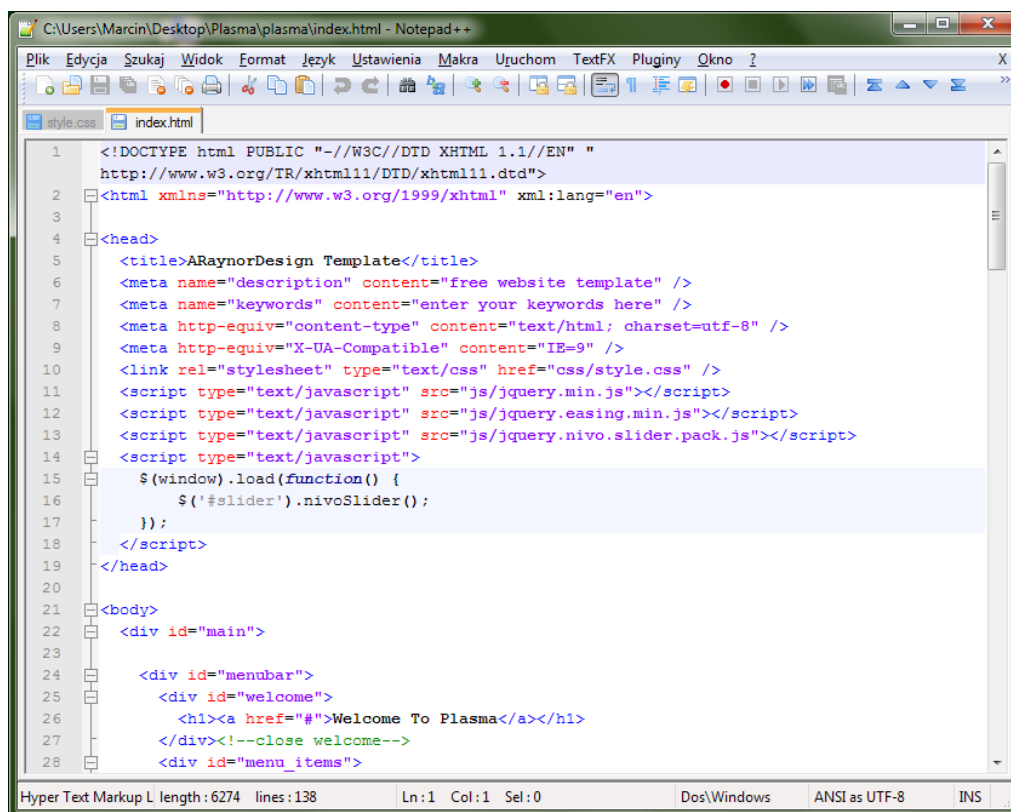
Po wybraniu szablonu należy go pobrać na dysk twardy, a następnie rozpakować – co ilustruje rysunek 8.



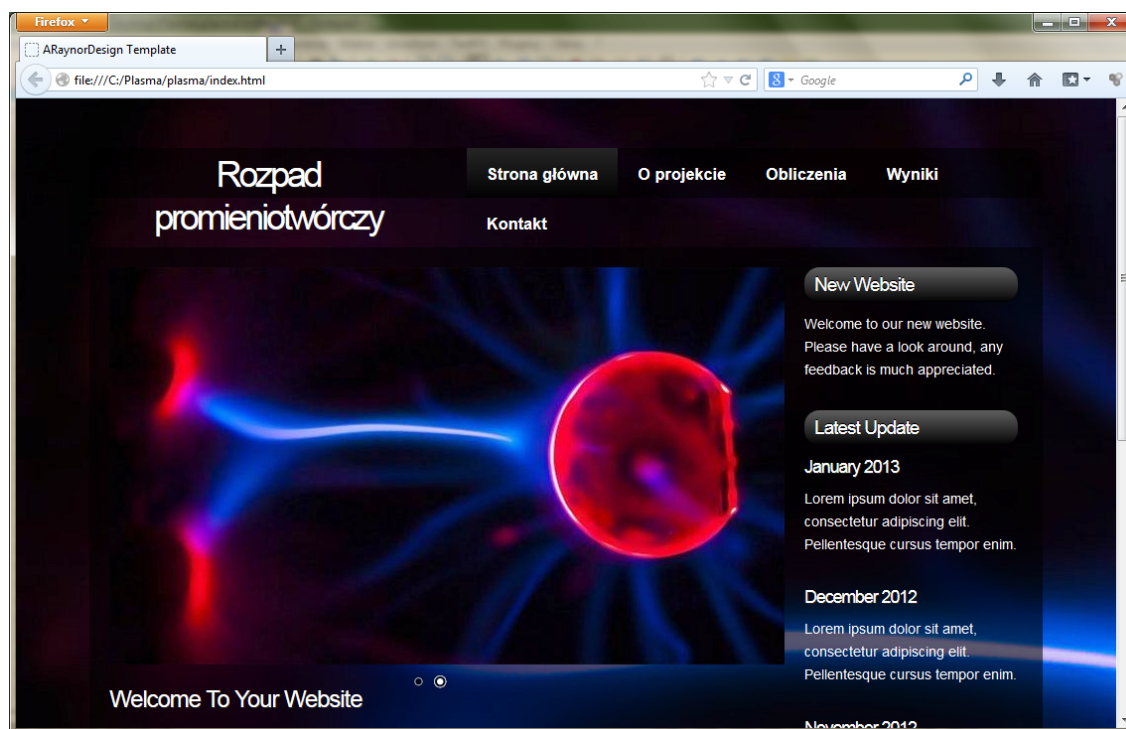
Rysunek 8. Zawartość szablonu witryny WWW

Na rysunku 8 widzimy, że w skład szablonu wchodzi pliki HTML stanowiące stronę główną (o nazwie *index*) i podstrony (*our work*, *projects*, *contact*, *testimonials*) oraz trzy foldery. Folder *css* zawiera kod arkuszy stylów CSS, folder *images* – pliki graficzne, a folder *js* zawiera kod JavaScript.

Po otwarciu w edytorze (np. w programie Notepad++ – rys. 9) plików *index.html* oraz *style.css* możemy je modyfikować, zmieniając treść witryny – w pliku HTML – oraz jej wygląd (szerokość kolumn, wielkość czcionki, kolor tekstu i tła, itp.). Na rysunku 10 przedstawiono szablon po edycji nazwy serwisu i nazw przycisków menu.



Rysunek 9. Edycja szablonu witryny w programie Notepad++



Rysunek 10. Szablon witryny WWW po zmianie nazwy i treści przycisków menu

Sposób działania – język PHP

Zaprogramowanie działania serwisu internetowego jest niewątpliwie największym wyzwaniem dla początkującego twórcy stron internetowych. Jednakże, kiedy poświęcimy czas na zrozumienie działania języka skryptowego (w naszym przypadku jest to PHP), opanujemy i tę umiejętność. Aby zapoznać się z podstawowymi pojęciami i logiką języka PHP, warto posłużyć się kursem dostępnym w Internecie: <http://phpkurs.pl/>. Ćwiczenia można znaleźć pod adresem `target=„_blank”http://wazniak.mimuw.edu.pl/index.php?title=AWWW-1st3.6-106.tresc-1.0`.

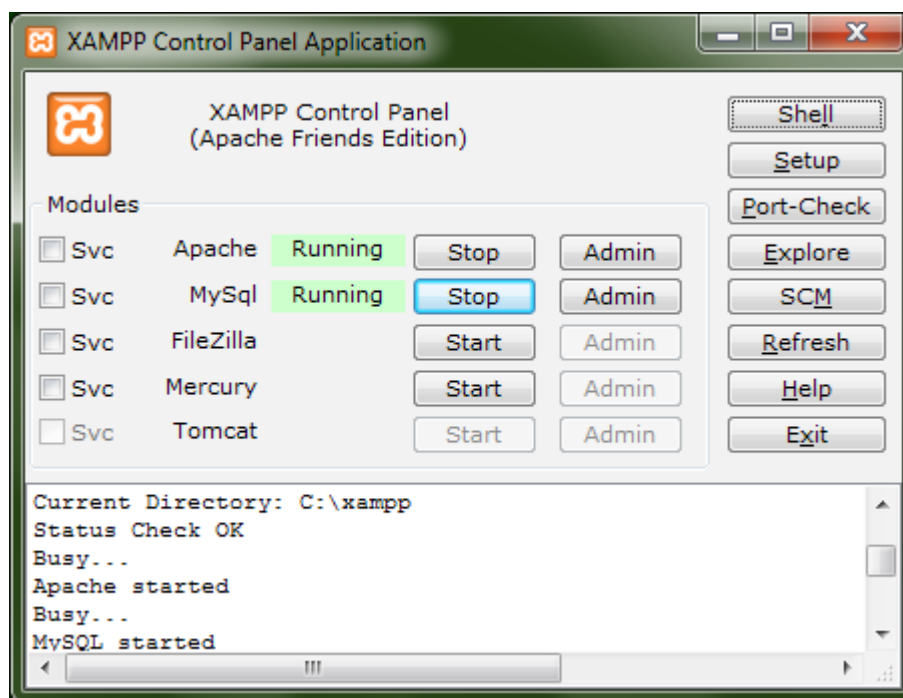
Aby wykonać ćwiczenia poświęcone działaniu serwisu, należy zainstalować serwer WWW i interpreter języka PHP, a także (do dalszej rozbudowy serwisu) bazę danych MySQL.

Instalowanie i konfigurowanie środowiska programistycznego MySQL, Apache i PHP

Do tworzenia i testowania działania dynamicznych stron internetowych niezbędne jest zainstalowanie i skonfigurowanie środowiska programistycznego. Trzy najpopularniejsze technologie tworzenia dynamicznych stron WWW to:

- ASP.NET,
- PHP & MySQL,
- JSP (Java Server Pages).

W niniejszym scenariuszu oparto się na technologii PHP & MySQL: serwerze WWW o nazwie Apache oraz środowisku obsługi języka skryptowego PHP, uzupełnionym systemem baz danych MySQL. Każdy z powyższych komponentów można instalować i konfigurować oddzielnie, jednakże wygodniejszym rozwiązaniem jest użycie aplikacji, która zainstaluje wszystkie te komponenty we właściwej kolejności. Jednym z programów, które instalują i konfiguruje serwer WWW, PHP oraz MySQL jest XAMPP. Program ten instaluje dodatkowo serwer poczty elektronicznej oraz aplikację phpMyAdmin, służącą do obsługi bazy danych. Program XAMPP odnajdziemy za pośrednictwem witryny <http://www.apachefriends.org>. Po dwukrotnym kliknięciu na pobranym pliku rozpoczyna się instalacja (Uwaga: pakiet XAMPP instalujemy w katalogu C:/xampp). Po zainstalowaniu możemy uruchomić panel administracyjny – rysunek 11.

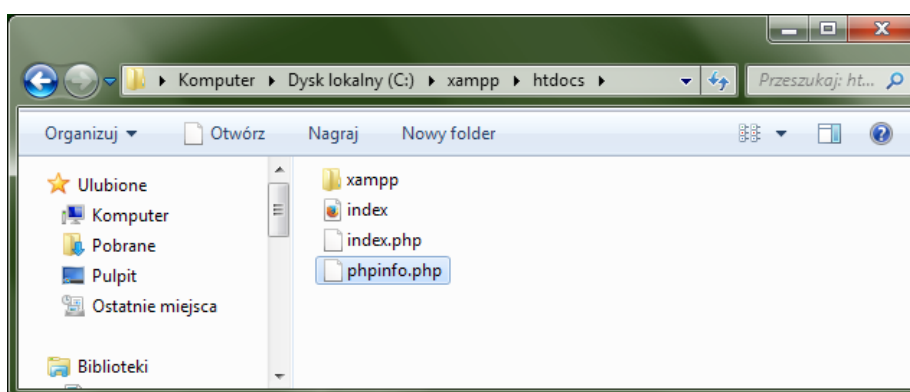


Rysunek 11. Panel administracyjny XAMPP

W celu przetestowania instalacji, należy utworzyć dwa skrypty PHP. Pierwszy z nich o nazwie `phpinfo`. Php sprawdza, czy oprogramowanie PHP jest aktywne i wyświetla dane o instalacji. Treść pliku `phpinfo.php` jest następująca:

```
<?php
phpinfo();
?>
```

Po zapisaniu treści pliku pod nazwą `phpinfo.php` umieszczamy go w podkatalogu `htdocs` katalogu `xampp` na dysku:

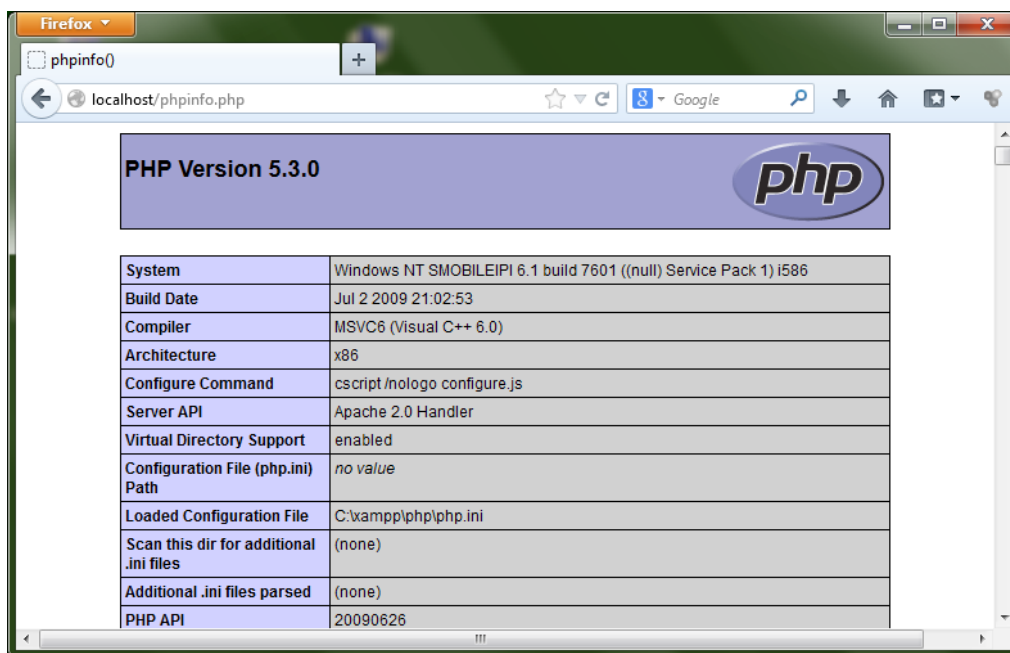


Rysunek 12. Zapisanie `phpinfo.php` w podkatalogu `htdocs` katalogu `xampp`

Uwaga 1. Aby kod języka PHP był interpretowany przez serwer WWW, czyli działał w przeglądarce, należy wszystkie pliki umieszczać w tym katalogu `htdocs`, a następnie uruchamiać poprzez wpisanie w przeglądarce `http://localhost/nazwa_skryptu.php`.

Uwaga 2. Po wyłączeniu i ponownym uruchomieniu komputera serwer WWW należy ponownie włączyć. W tym celu uruchamiamy panel administracyjny XAMPP [Start->Wszystkie programy -> XAMPP for Windows -> XAMPP Control panel] i przy nazwie serwera WWW Apache klikamy przycisk [Start]. Jeśli chcemy,

aby serwer WWW włączył się automatycznie z włączeniem komputera, należy zaznaczyć pole [Svc] znajdujące się po lewej stronie nazwy serwera WWW Apache w panelu administracyjnym XAMPP. Efekt jego uruchomienia w przeglądarce będzie taki jak na rysunku 13.



Rysunek 13. Informacja o zainstalowanej wersji oprogramowania PHP

Warto przyrzeć się witrynie XAMPP, dostępnej po wpisaniu <http://localhost/xampp/> w przeglądarce. Przy użyciu tej strony możemy np. sprawdzić stan działania środowiska programistycznego – rys. 14, jak również zapoznać się z dokumentacją, przykładowymi aplikacjami.



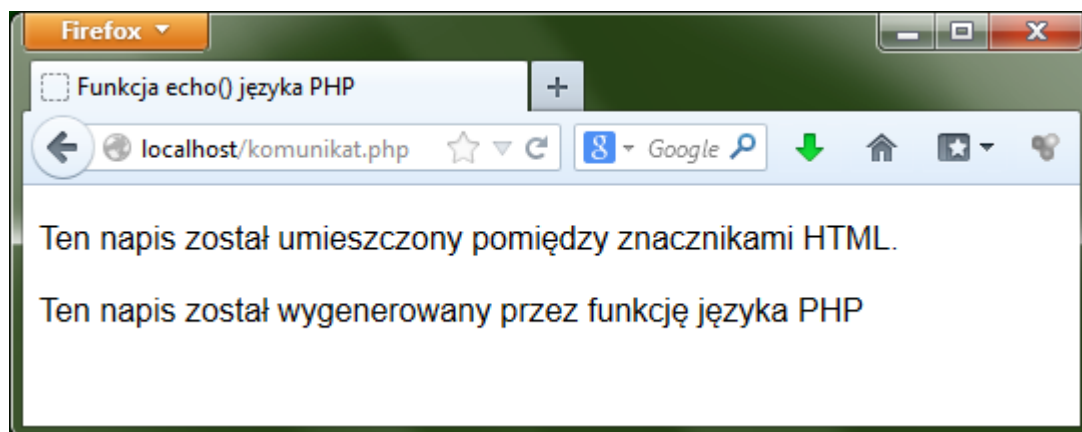
Rysunek 14. Witryna XAMPP na serwerze lokalnym (własnym komputerze)

Przed przystąpieniem do tworzenia dynamicznych stron internetowych należy poznać, w jaki sposób przysyłać dane do przeglądarki. Jedną z najpopularniejszych funkcji służących do tego celu jest funkcja echo(). Aby przesłać dane do przeglądarki, należy wykonać następujące kroki:

1. W edytorze (np. Notepad++) utworzyć plik o nazwie komunikat.php,
2. Uzupełnić treść pliku jak poniżej:

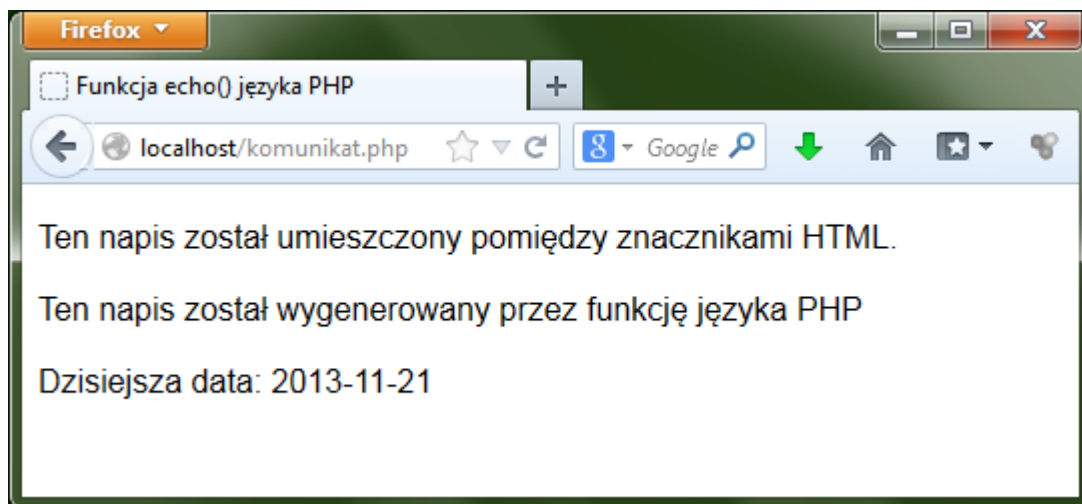
```
<html>
<head>
<title> Funkcja echo() języka PHP</title>
</head>
<body>
<font face=„Arial”>
<p> Ten napis został umieszczony pomiędzy znacznikami HTML.</p>
<?php
echo 'Ten napis został wygenerowany przez funkcję języka PHP'
;?>
</body>
</html>
```

3. Po zapisaniu zmian w pliku, należy uruchomić go w przeglądarce. Efekt powinien być podobny jak na rysunku 15.



Rysunek 15. Działanie funkcji echo języka PHP

W kolejnym kroku można utworzyć pierwszy dynamiczny element na stronie – datę jej wyświetlenia (rys. 16). Wystarczy zmodyfikować kod pliku komunikat.php, dodając w odpowiednim miejscu wiersz: `echo date(„Y-m-d”);`

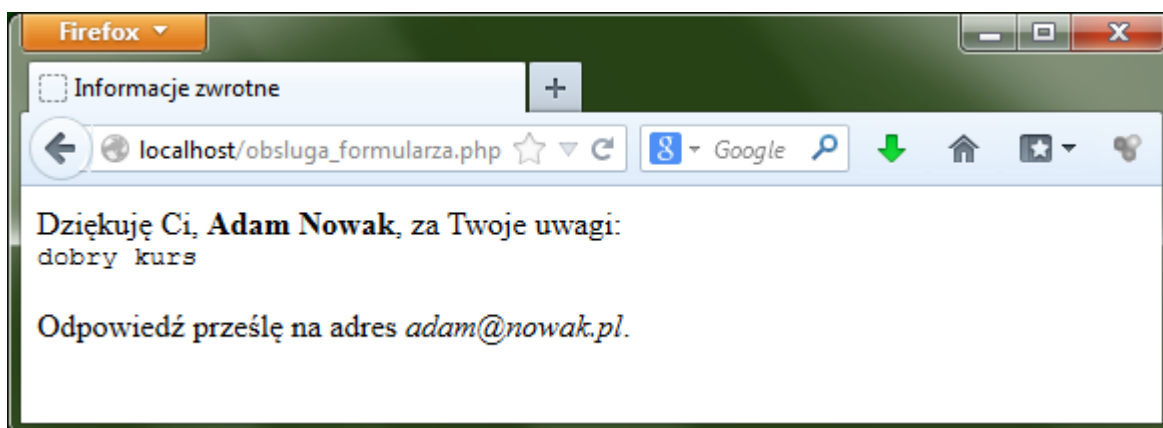


Rysunek 16. Wykorzystanie funkcji date() języka PHP – strona wyświetla aktualną datę

Ważnym krokiem w tworzeniu serwisu WWW jest tworzenie stron przetwarzających formularze wypełniane przez użytkowników. Aby formularz zaprezentowany na początku niniejszej lekcji mógł zostać przetworzony, należy utworzyć skrypt obsługa_formularza.php o następującej treści.

```
<html>
<head>
<title>Informacje zwrotne</title>
</head>
<body>
<?php
// Tworzymy skróty zmiennych formularza
$imie = $_REQUEST['imie'];
$nazwisko = $_REQUEST['nazwisko'];
$email = $_REQUEST['email'];
$uwagi = $_REQUEST['uwagi'];
// Wyświetlamy otrzymane informacje
echo „<p>Dziękuję Ci, <b>$imie $nazwisko</b>, za Twoje uwagi:<br />
<tt>$uwagi</tt></p>
<p>Odpowiedź prześlę na adres <i>$email</i>.</p>\n”;
?>
</body>
</html>
```

Po wprowadzeniu danych przez użytkownika, zostanie wyświetlona strona jak na rysunku 17.

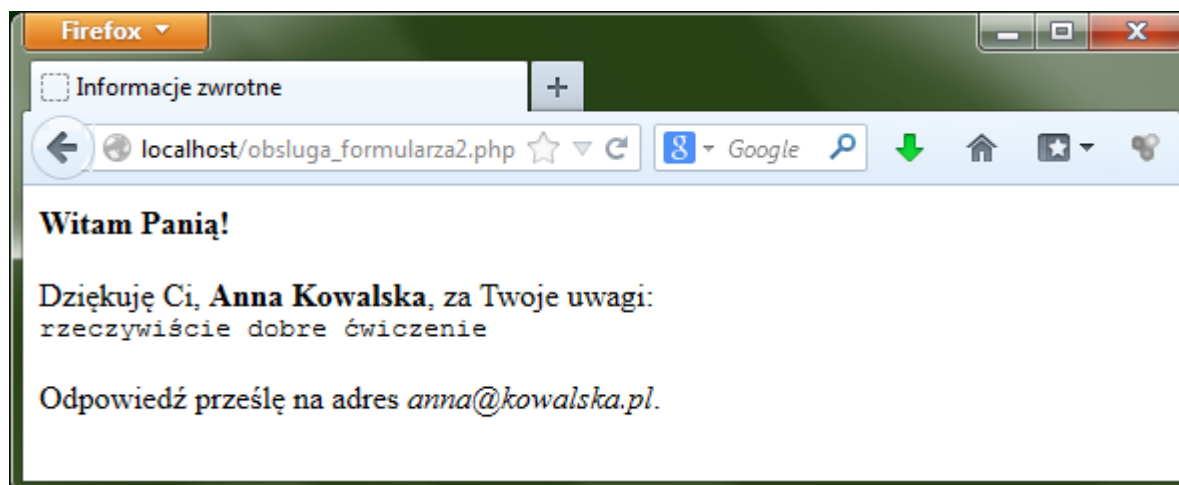


Rysunek 17. Informacja zwrotna do użytkownika

Kolejnym krokiem może być modyfikacja skryptu `obsługa_formularza.php` tak, aby kod PHP sprawdzał, jakiego wyboru dokonał użytkownik – i w zależności od tego – wyświetlał odpowiednie dane. W ćwiczeniu należy wykorzystać instrukcje warunkowe i operatory języka PHP:

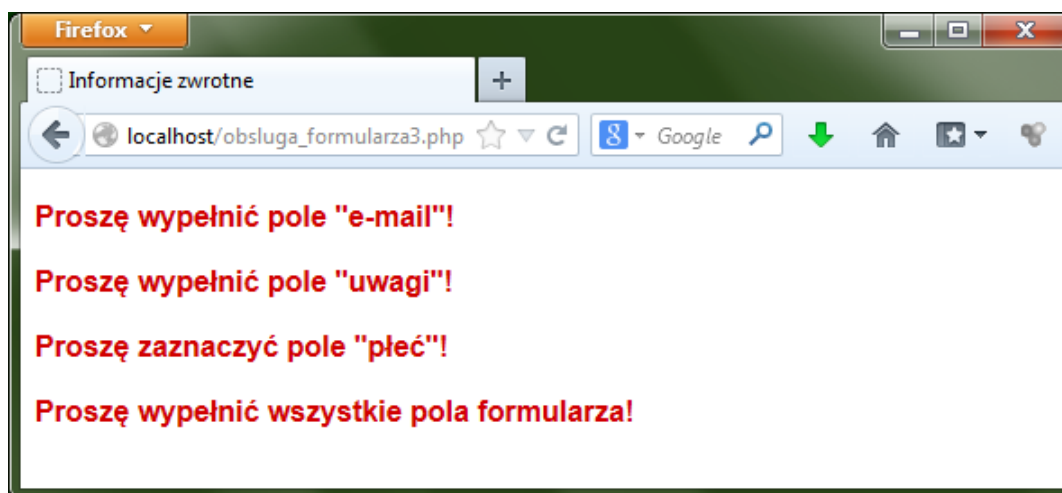
```
if ($plec == 'M') {
    echo '<p><b>Witam Pana!</b></p>';
} elseif ($plec == 'K') {
    echo '<p><b>Witam Panią!</b></p>';
}
```

Efekt działania powinien być podobny, jak na rysunku 18.



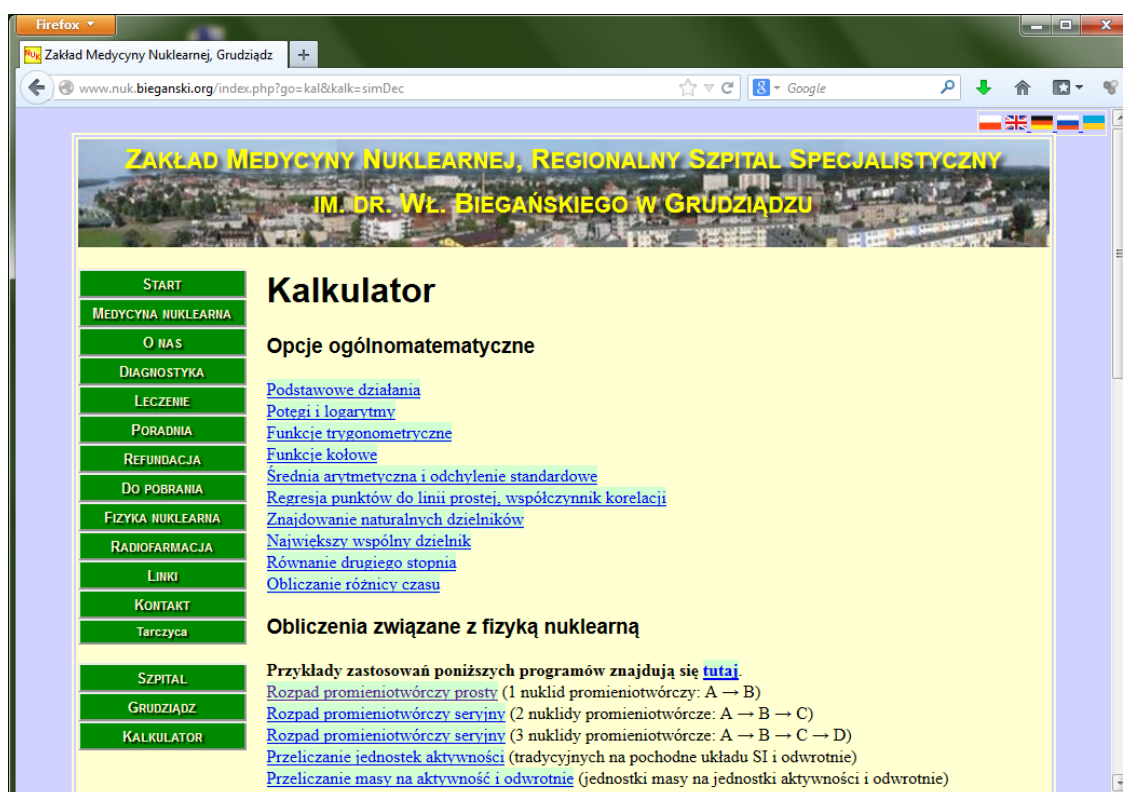
Rysunek 18. Efekt działania instrukcji warunkowych języka PHP

Niezwykle istotnym krokiem w tworzeniu serwisu WWW jest zabezpieczenie formularzy przed wprowadzaniem nieprawidłowych danych. Mechanizm weryfikacji danych wprowadzanych przez użytkowników jest nieodłącznym elementem dynamicznych serwisów internetowych. Zmodyfikowany skrypt do obsługi formularza powinien działać tak, aby w momencie przesłania błędnie wypełnionego formularza wyświetlał się odpowiedni komunikat – rysunek 19.



Rysunek 19. Komunikaty wyświetlane w przypadku błędnego wypełnienia formularza

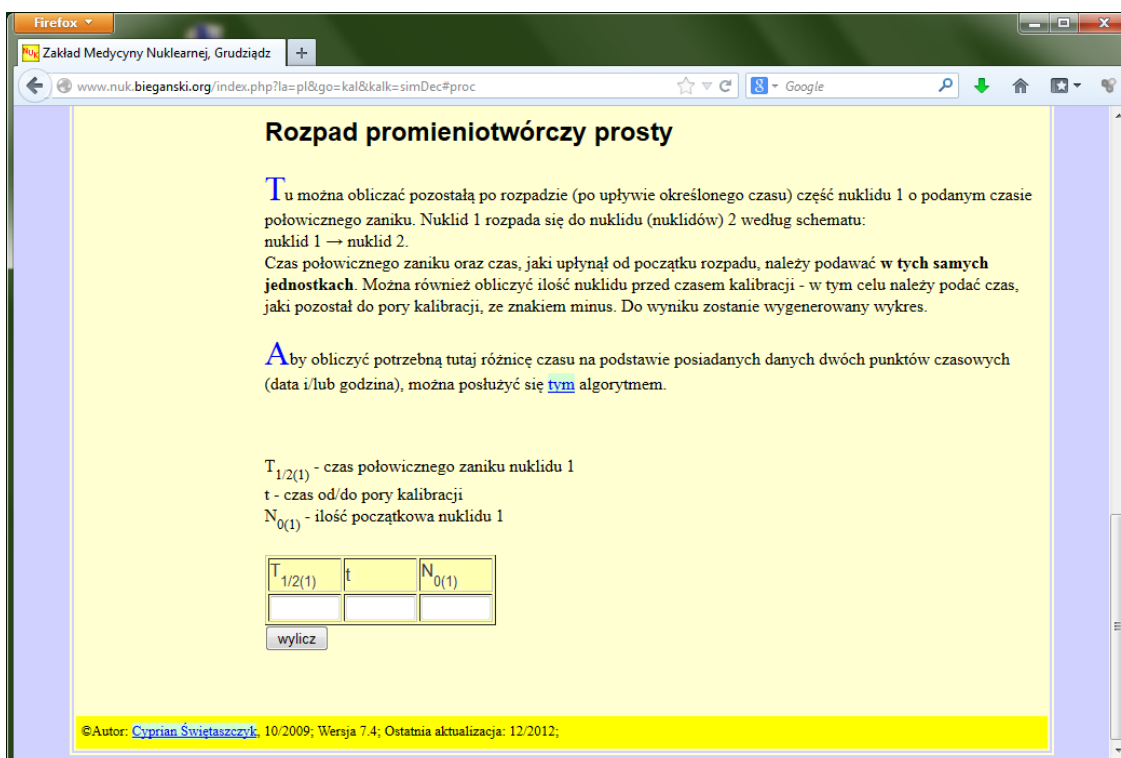
Po wykonaniu ćwiczeń poświęconych obsłudze formularza HTML przy użyciu funkcji języka PHP potrafimy utworzyć serwis WWW służący do przeliczania wyników. Przykładem takiego serwisu jest strona Zakładu Medycyny Nuklearnej RSS w Grudziądzu¹, umożliwiająca prowadzenie obliczeń związanych z fizyką nuklearną – rysunek 20:



Rysunek 20. Serwis internetowy Zakładu Medycyny Nuklearnej (ZMN) umożliwiający obliczenia na podstawie danych wprowadzanych przez użytkownika

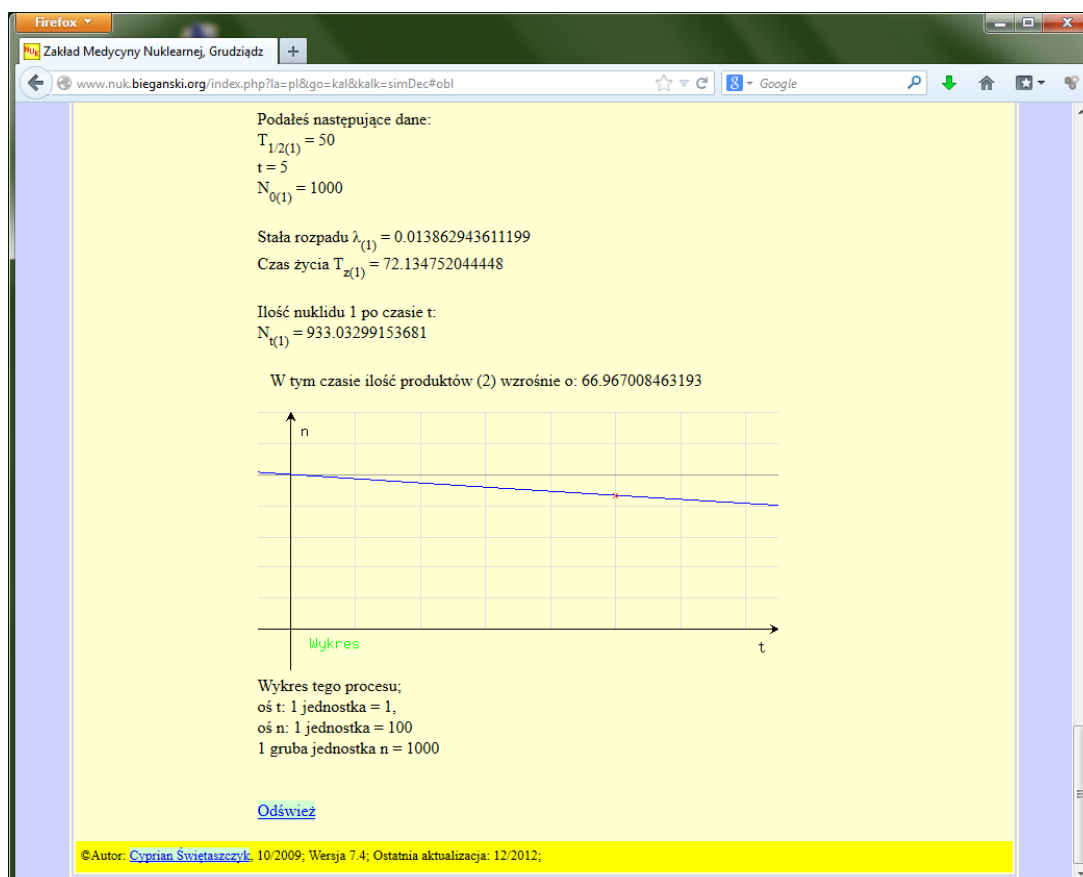
Na stronie <http://www.nuk.bieganski.org/index.php?la=pl&go=kal&kalk=simDec#proc> można obliczyć ilość pierwiastka promieniotwórczego, która pozostanie po określonym czasie (co jest związane ze stałą rozpadu – wielkością charakterystyczną dla każdego pierwiastka) – rysunek 21.

¹ <http://www.nuk.bieganski.org/index.php?go=kal&kalk=simDec>.



Rysunek 21. Serwis internetowy ZMN – formularz do obliczania rozpadu promieniotwórczego

Po wprowadzeniu do formularza przykładowych danych i naciśnięciu przycisku [Wylicz] przechodzimy do strony prezentującej wyniki obliczeń – rysunek 22.



Rysunek 22. Serwis internetowy ZMN – prezentacja wyników obliczenia rozpadu promieniotwórczego

Utworzenie algorytmu obliczającego wynik na podstawie danych z formularza na stronie WWW jest w języku PHP, podobne jak w innych językach programowania. Dysponujemy bowiem możliwością definiowania zmiennych oraz wykonywania na nich operacji matematycznych, czego przykładem może być poniższy fragment kodu:

```
<?php
$a=1;
$b=4;
$c=23;
$suma = $a+$b+$c;
echo „$suma”;
?>
```

W przedstawionym przykładzie kodu wartości zmiennych \$a, \$b, \$c mogą oczywiście pochodzić z formularza HTML wypełnianego przez użytkownika.

Nie pozostaje zatem nic innego jak do serwisu „Rozpad promieniotwórczy” utworzonego w ramach realizacji niniejszego scenariusza dodać podstronę obliczającą wynik rozpadu wybranego pierwiastka.

Rozwinięcie zagadnień poruszanych w niniejszym scenariuszu stanowi kurs pt. „Tworzenie serwisów internetowych”, dostępny w serwisie IT Szkoła: <http://it-szkola.edu.pl/kkurs,kurs,2>.

Literatura

1. Cohen J., *Serwisy WWW. Projektowanie, tworzenie, zarządzanie*, Helion, Gliwice 2004
2. Meloni J.C., *PHP, MySQL i Apache dla każdego*, Helion, Gliwice 2007
3. Price J., Price L., *Profesjonalny serwis WWW*, Helion, Gliwice 2002
4. Sokół R., *Internet. Ilustrowany przewodnik*, Helion, Gliwice 2007
5. Ullman L., *PHP i MySQL. Dynamiczne strony WWW*, Helion, Gliwice 2004
6. Welling L., Thomson L., *PHP i MySQL. Vademecum profesjonalisty*, Helion, Gliwice 2005

Przebieg zajęć

Wprowadzenie (10 minut)

Omówienie wprowadzenia teoretycznego do niniejszej lekcji, przy użyciu przygotowanej prezentacji.

Praca indywidualna lub w zespołach (30 minut)

Praca indywidualna lub w zespołach dwuosobowych (uczniowie dzielą się zadaniami w ramach przygotowywanej wspólnie witryny internetowej). Uczniowie wykonują ćwiczenia, korzystając w razie potrzeby z treści wprowadzenia teoretycznego do niniejszej lekcji.

Dyskusja podsumowująca (5 minut)

Omówienie rezultatów pracy – efektów wykonania ćwiczeń.

Uświadomienie uczniom możliwości nauki tworzenia stron WWW z wykorzystaniem bezpłatnych technologii PHP, MySQL i Apache oraz bezpłatnych szablonów witryn WWW.

Nauka instalacji i konfiguracji serwera WWW. Nauka elementów języka PHP.

Sprawdzenie wiedzy

Ćwiczenia z tworzenia własnego serwisu WWW

➡ Ćwiczenie 2.1.

Oprócz prezentowania treści, jednym z najczęstszych zastosowań serwisu WWW jest zbieranie danych od użytkowników.

Utwórz w języku HTML formularz do pobierania danych od użytkownika. Efekt wyświetlenia formularza w przeglądarce powinien być następujący:

Firefox

Formularz użytkownika

localhost/formularz_uzytkownika.html

Wprowadź do poniższego formularza informacje na swój temat:

Imię:

Nazwisko:

Wiek:

Płeć: Mężczyzna Kobieta

Adres e-mail:

Uwagi:

Wyślij dane

➡ Ćwiczenie 2.2.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, pobierz szablon witryny internetowej, a następnie dostosuj go (modyfikując kod plików HTML i CSS) do tematyki własnego projektu edukacyjnego.

➡ Ćwiczenie 2.3.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, zainstaluj pakiet XAMPP, a następnie przetestuj instalację tworząc skrypt zawierający funkcję `phpinfo()`.

➡ Ćwiczenie 2.4.

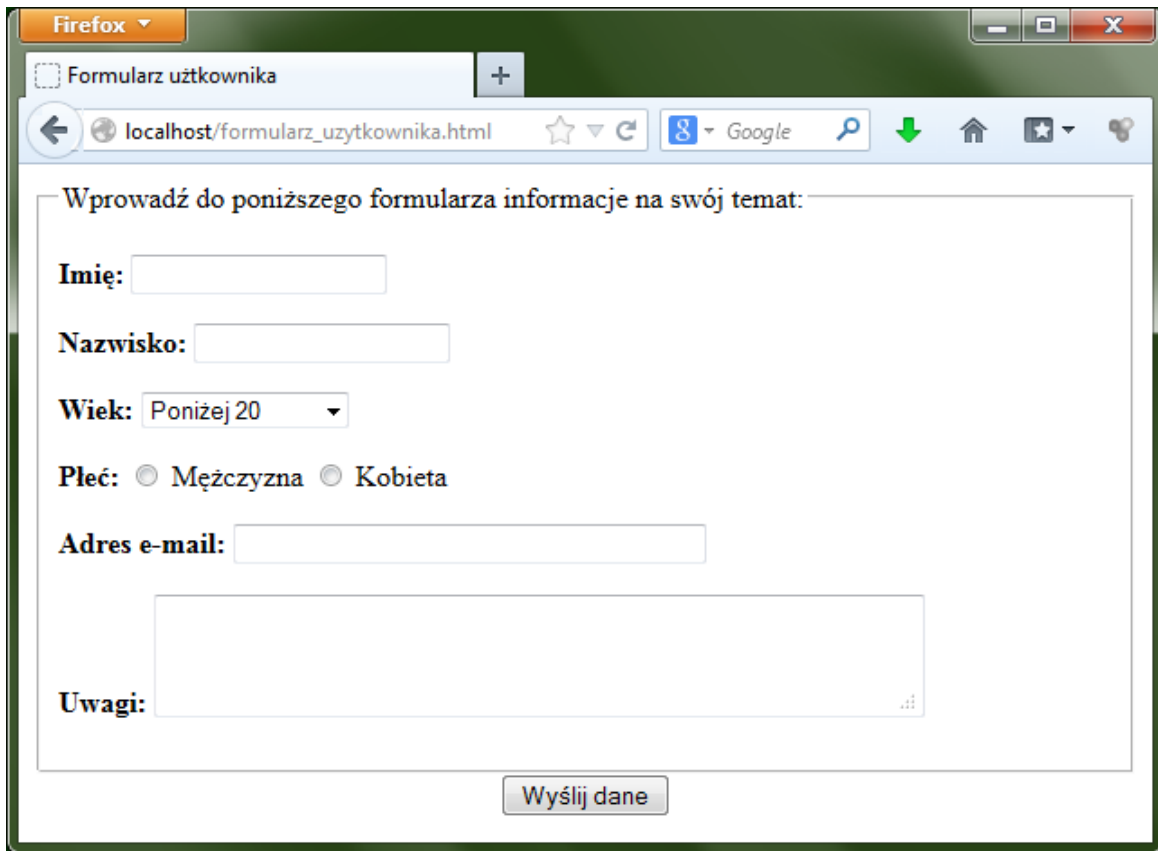
Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, utwórz skrypt zawierający funkcję `echo()` języka PHP. Przetestuj wyświetlanie tekstu przy użyciu znaczników HTML oraz przy użyciu języka PHP.

➡ Ćwiczenie 2.5.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, utwórz prostą stronę WWW wyświetlającą aktualną datę. Wykorzystaj w tym celu funkcję `date()` języka PHP.

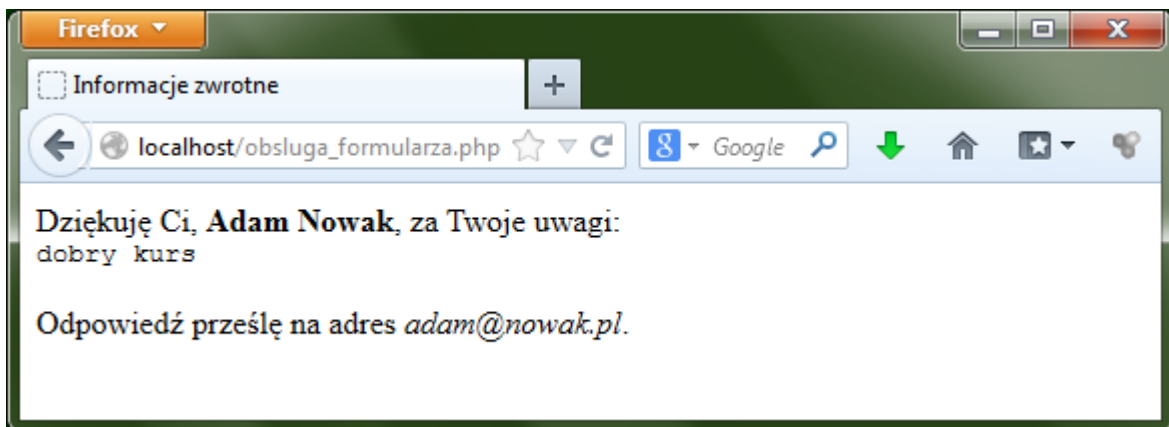
➡ Ćwiczenie 2.6.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, utwórz skrypt przetwarzający formularz, do którego użytkownik wprowadza dane:



The screenshot shows a Firefox browser window with the title "Formularz użytkownika". The address bar shows "localhost/formularz_uzytkownika.html". The page content includes a heading "Wprowadź do poniższego formularza informacje na swój temat:" followed by several input fields: "Imię:" (text), "Nazwisko:" (text), "Wiek:" (dropdown menu with "Poniżej 20" selected), "Płeć:" (radio buttons for "Mężczyzna" and "Kobieta"), "Adres e-mail:" (text), and "Uwagi:" (text area). A "Wyślij dane" button is located at the bottom center.

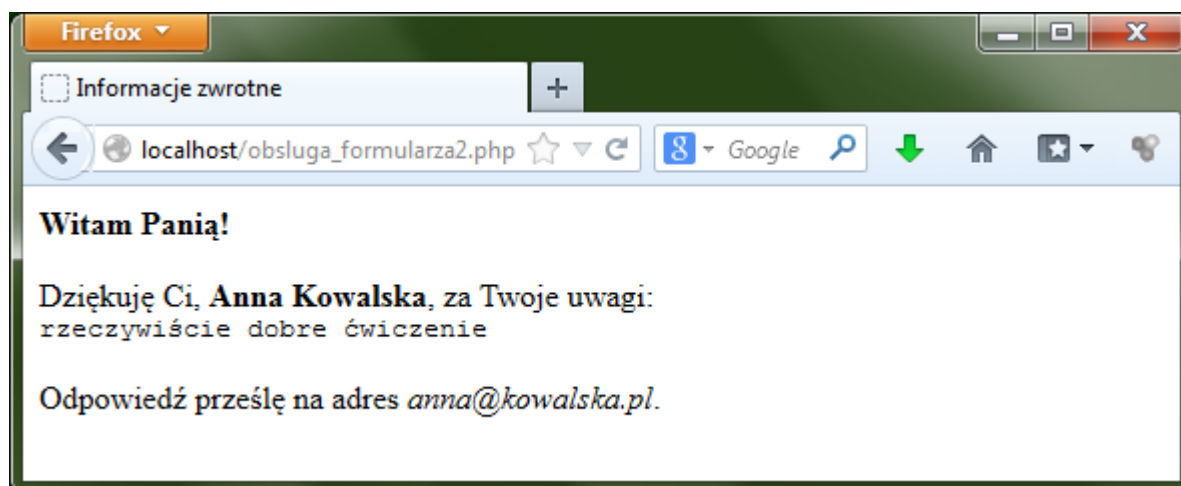
Efekt wypełnienia formularza i kliknięcia przycisku [Wyślij dane] powinien być następujący:



The screenshot shows a Firefox browser window with the title "Informacje zwrotne". The address bar shows "localhost/obsługa_formularza.php". The page content displays a confirmation message: "Dziękuję Ci, **Adam Nowak**, za Twoje uwagi: dobry kurs" and "Odpowiedź prześlę na adres *adam@nowak.pl*."

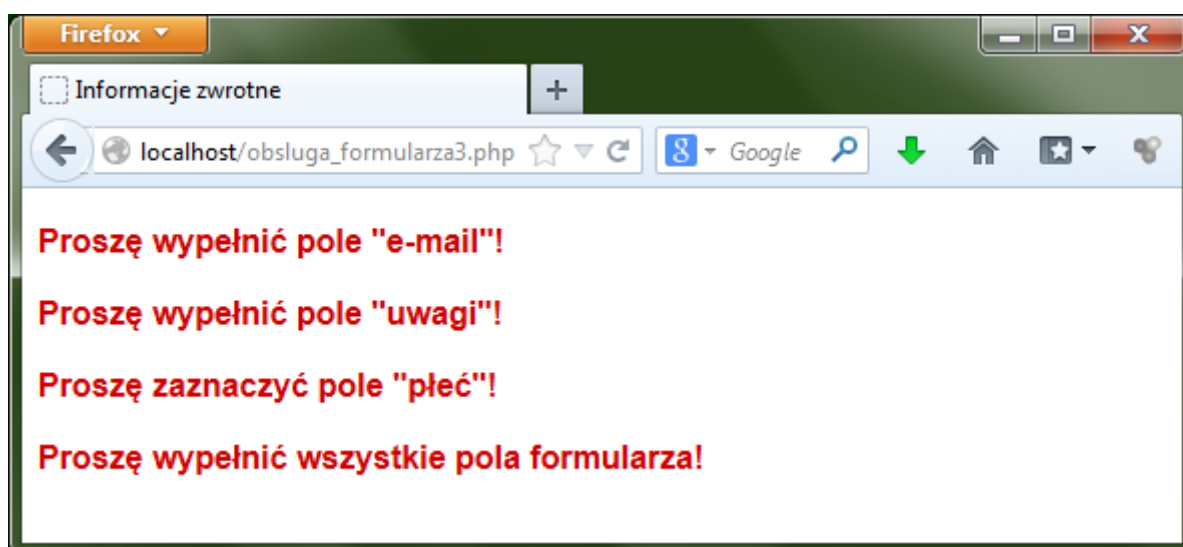
➡ Ćwiczenie 2.7.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, utwórz skrypt przetwarzający formularz w taki sposób, aby serwer wyświetlał powitanie „Witam Pana” lub „Witam Panią” w zależności od tego, którą płeć zaznaczy użytkownik w formularzu:



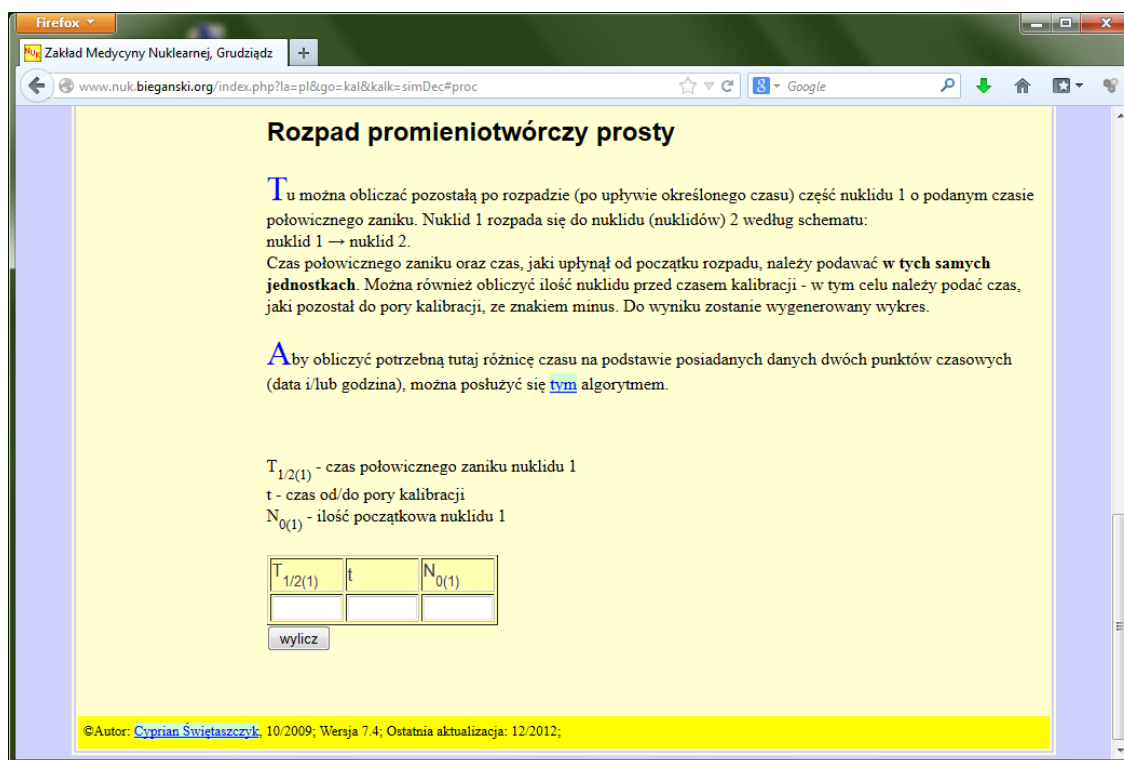
➡ Ćwiczenie 2.8.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, utwórz skrypt przetwarzający formularz w taki sposób, aby serwer wyświetlał stosowne komunikaty w przypadku próby przesłania niepełnego formularza:



➡ Ćwiczenie 2.9.

Postępując według wskazówek zawartych we wprowadzeniu teoretycznym, utwórz podstronę serwisu obliczającą wynik rozpadu promieniotwórczego, działającą podobnie jak serwis internetowy ZMN:



Wskazówki:

1. Formularz HTML może zawierać np.:
 - a) listę rozwijaną z nazwami pierwiastków promieniotwórczych,
 - b) pole do wpisania ilości pierwiastka,
 - c) pole do wpisania czasu trwania rozpadu.
2. W kodzie PHP należy każdemu pierwiastkowi przypisać odpowiednią stałą rozpadu, a następnie zapisać wzór na rozpad określonej ilości pierwiastka o określonej stałej rozpadu w określonym czasie; przy czym do tego wzoru będą podstawiane wartości zmiennych na podstawie formularza wypełnianego przez użytkownika.

Test wiedzy na zakończenie wszystkich lekcji

Ocenianie

Ćwiczenie 2.1

- porównanie efektu wyświetlenia formularza ze wzorem widocznym na rysunku do ćwiczenia

Ćwiczenie 2.2

- ocena trafności wyboru szablonu witryny (pod względem kolorystyki i rozmieszczenia elementów na stronie głównej) z tematem serwisu
- ocena liczby dokonanych zmian (edycja treści, edycja wyglądu itp.)

Ćwiczenie 2.3

- ocena prawidłowości instalacji i konfiguracji pakietu XAMPP (czy działa skrypt wyświetlający funkcję phpinfo())

Ćwiczenie 2.4

- ocena efektu działania utworzonego skryptu (czy jest zgodny z poleceniem zawartym w ćwiczeniu)



Ćwiczenie 2.5

- ocena efektu działania utworzonego skryptu (czy jest zgodny z poleceniem zawartym w ćwiczeniu)

Ćwiczenie 2.6

- ocena efektu działania utworzonego skryptu (czy jest zgodny z poleceniem zawartym w ćwiczeniu)

Ćwiczenie 2.7

- ocena efektu działania utworzonego skryptu (czy jest zgodny z poleceniem zawartym w ćwiczeniu)

Ćwiczenie 2.8

- ocena efektu działania utworzonego skryptu (czy jest zgodny z poleceniem zawartym w ćwiczeniu)

Ćwiczenie 2.9

- ocena efektów działania serwisu:
 - jaki wzór na obliczanie rozpadu promieniotwórczego uczeń zaimplementował
 - czy uczeń utworzył odpowiedni formularz do pobierania danych od użytkownika
 - czy uczeń utworzył skrypt obliczający wynik na podstawie danych z formularza
- ocena estetyki serwisu (wygląd, prezentacja formularza i wyników obliczeń)

Zaliczenie testu wiedzy w przypadku co najmniej połowy poprawnych odpowiedzi.

Dostępne pliki

1. Prezentacja 2
2. Instalator programu XAMPP v1.7.2
3. Ćwiczenia (Zadania 2)
4. Pliki do ćwiczeń (materiały pomocnicze)
5. Film 2 – „Pobieranie i edycja szablonu witryny”
6. Film 3 – „Instalowanie i konfigurowanie pakietu XAMPP”
7. Test



Człowiek - najlepsza inwestycja



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



WARSZAWSKA
WYŻSZA SZKOŁA
INFORMATYKI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego