

# Young Programmer

[www.inceptio.org.pl](http://www.inceptio.org.pl)



**Dr inż. Małgorzata Janik**

Organizator:



Partnerzy:



Dofinansowanie:



MIASTO  
STOŁĘCZNE  
WARSZAWA



# O mnie...



**dr inż. Małgorzata Janik**

Zakład Fizyki Jądrowej

Wydział Fizyki Politechniki Warszawskiej

[\*majanik@if.pw.edu.pl\*](mailto:majanik@if.pw.edu.pl)

## **Prowadzone zajęcia na Politechnice Warszawskiej**

- Podstawy programowania (język C) - laboratoria
- Języki programowania (język C++) - laboratoria
- Komputerowa Analiza Danych Doświadczalnych (ROOT) - laboratoria
- Podstawy Technologii Informacyjnej - laboratoria
- Laboratorium Fizyki II - laboratoria
- Sieci Komputerowe - laboratoria
- Elektrodynamika - ćwiczenia

# Ramowy program warsztatów

Zajęcia 1: Zajęcia wprowadzające, HTML

(podstawy pracy w środowisku WWW, podstawy języka HTML)

Zajęcia 2: Style CSS

Zajęcia 3: Formularze HTML

Zajęcia 4: Podstawy języka PHP

Zajęcia 5: Język PHP cd.

Zajęcia 6: Synteza kursu

<http://www.if.pw.edu.pl/~majanik/wiki/index.php/HTML%2BPHP>

# Regulamin zajęć I

## I Ogólne wytyczne

- 1. Zajęcia w laboratoriach i pracowni przeznaczone są dla osób powyżej 13. roku życia.**
- 2. Na zajęcia należy przybyć punktualnie, na wyznaczoną godzinę.**
- 3. Osoby uczestniczące w zajęciach zobowiązane są do zgłaszania prowadzącym wszelkich zaistniałych problemów.**
- 4. Uczestnicy zobowiązani są do przestrzegania porządku i wykonywania poleceń prowadzących.**

# Regulamin zajęć II

## II Przebieg zajęć

- 1. Do laboratorium/pracowni uczestnicy zajęć wchodzą w obecności osoby prowadzącej zajęcia.**
- 2. Obowiązkiem każdego uczestnika jest utrzymanie porządku na stanowisku pracy oraz w laboratorium/pracowni.**
- 3. Zabrania się wnoszenia do pomieszczenia napoi oraz innych produktów spożywczych.**
- 4. Okrycia wierzchnie, parasole itp. należy pozostawić poza laboratorium lub złożyć w miejscu wskazanym przez prowadzącego, z dala od sprzętu elektronicznego.**
- 5. Wszelkie czynności w trakcie zajęć należy przeprowadzać z zachowaniem należytych środków ostrożności, aby nie uszkodzić sprzętu oraz nie narażać na niebezpieczeństwo siebie i innych.**
- 6. Każde uszkodzenie lub nienaturalne zachowanie sprzętu i każdy błąd w działaniu oprogramowania musi być natychmiast zgłoszone prowadzącemu.**

# Regulamin zajęć

## II Przebieg zajęć c.d.

**7. Na komputerach laboratoryjnych można wykonywać tylko czynności związane z prowadzonymi warsztatami. W szczególności zabrania się przeglądania innych stron WWW, portali społecznościowych, poczty itd.**

**8. Zabrania się podejmowania czynności zmierzających to umyślnego zakłócenia pracy własnego lub innych komputerów.**

**9. W razie nieszczęśliwego wypadku należy natychmiast zgłosić się do prowadzącego i podać okoliczności zdarzenia.**

**10. Zobowiązuje się wszystkich uczestników do ścisłego przestrzegania przepisów BHP dotyczących wykonywania ćwiczeń.**

**11. We wszystkich sprawach nieobjętych regulaminem należy zgłaszać się do osoby prowadzącej zajęcia.**

# **HTML+PHP #1**

## **Wprowadzenie**



**Dr inż. Małgorzata Janik**

# ***HTML+PHP #1***

## ***Wprowadzenie***

# **Wprowadzenie do systemu Linux**

*Ścieżki dostępu*  
*Prawa dostępu*

**Dr inż. Małgorzata Janik**



# Komendy systemu linux

**ls** (list) - wyświetla zawartość bieżącego katalogu lub katalogu podanego jako parametr.

**cd** (change directory) - wchodzi do katalogu, np. `cd katalog1`

**mkdir** (make directory) - do tworzenia katalogów. Przykład: `mkdir nazwa_katalogu`

**cp** (copy) - do kopiowania plików i katalogów. Przykłady: `cp plik1 plik2`

**cp -r** - kopiuje katalog wraz z zawartością np. `cp -r katalog1 katalog2`

**\*** - gwiazdka zastępuje dowolny ciąg znaków np.:

**cp \* alfa/** - kopiuje wszystkie pliki z bieżącego katalogu do katalogu alfa

**mv** (move) - przenosi plik/pliki, służy też do zmiany nazwy pliku lub katalogu.

**mv plik1 plik2** - zmienia nazwę plik1 na plik2

**rm** (remove) - usuwa pliki. Przykład:

**rm plik1** - usuwa plik1

**rm \*** - usuwa wszystkie pliki z bieżącego katalogu (należy używać bardzo ostrożnie - sprawdzić, czy rzeczywiście chcemy wszystko skasować).

**rm -r** - usuwa cały katalog razem z zawartością

**more** - pozwala na przeglądanie danych (plików, komunikatów poleceń) ekran po ekranie.

**kate plik.txt** – uruchamia edytor kate tworząc plik plik.txt

**cat** - podobnie do polecenia 'more' pokazuje zawartość pliku ale nie zatrzymuje się ekran po ekranie tylko wyświetla od razu całość.

# Ścieżki dostępu

## **Bezwzględna (przeglądanie lokalne):**

/var/www/tmp/plik.txt

/home/tycjan/public\_html/img/maly\_kaczor.png

/home/tycjan/public\_html/kurs/program.html

## **Bezwzględna (przeglądanie z serwera):**

/img/maly\_kaczor.png

/kurs/program.html

## **Względna**

img/maly\_kaczor.png

program.html

## **Ważne!**

W ścieżkach linuksowych używamy znaku / (slash)

Stronę internetową z reguły należy umieścić w katalogu public\_html

# Prawa dostępu

**r**

**w**

właściciel (user)

**x**

---

**r**

**w**

grupa (group)

**x**

---

**r**

**w**

pozostali (others)

**x**

# Prawa dostępu

	<b>Plik</b>	<b>Katalog</b>
r	odczyt	odczyt zawartości katalogu (listing plików)
w	zapis	zmiana zawartości (np. tworzenie nowych lub usuwanie starych plików)
x	uruchamianie (dotyczy skryptów i programów binarnych)	dostęp do katalogu i podkatalogów

# ***HTML+PHP #1***

## ***Wprowadzenie***

## **Podstawy HTML**

**Dr inż. Małgorzata Janik**

# Znaczniki

## Znaczniki otwierające i zamykające:

`<p>Akapit tekstu.</p>`

`<strong>Tekst pogrubiony</strong>`

## Znaczniki otwierające (bez zamykających)

`<br />`

`<hr />`

`<meta charset="UTF-8" />`

Znaczniki można w sobie zagnieżdżać!

Pamiętajmy o slashu przy znacznikach niezamykanych.

# Znaczniki

## Znaczniki otwierające i zamykające:

`<p>Akapit tekstu.</p>`

`<strong>Tekst pogrubiony</strong>`

## Znaczniki otwierające (bez zamykających)

`<br />`

`<hr />`

`<meta charset="UTF-8" />`

*atrybut*

Znaczniki można w sobie zagnieżdżać!

Pamiętajmy o slashu przy znacznikach niezamykanych.

# Ogólna struktura dokumentu

```
<!DOCTYPE html>
<html>
  <head>
    <!-- sekcja nagłówkowa, głównie znaczniki meta --> (komentarze)
    <title>Tytuł strony</title>
  </head>
  <body>
    <!-- właściwa treść strony -->
  </body>
</html>
```

**<!DOCTYPE .....** - określa typ dokumentu, informacja dla przeglądarki, jak wyświetlać dokument

**<html> </html>** - kod html musi się znaleźć między tymi znacznikami

**<!-- komentarze -->** Komentarze - tekst wewnątrz takich znaczników nie pojawi się na stronie, to komentarz dla osób projektujących stronę.  
Bardzo przydatne!



# meta – kodowanie

```
<meta charset="rodzaj_kodowania" />
```

Przykładowe opcje:

- iso-8859-2 (**windows**)
- UTF-8 (**linux**)
- windows-1250 (niezalecane)

# meta – informacje o stronie

```
<meta name="keywords" content="słowa, kluczowe" />  
<meta name="description" content="opis strony" />  
<meta name="author" content="imię i nazwisko autora strony" />  
<meta name="copyright" content="właściciel praw autorskich" />  
<meta name="generator" content="generator strony (np. notatnik)" />  
<meta name="robots" content="index, follow" />
```

Meta-informacje umieszczamy wewnątrz sekcji „head”.

# Przykładowa sekcja <head>

```
<head>
  <meta charset="UTF-8" />
  <meta name="keywords" content="warsztat,php,inceptio" />
  <meta name="description" content="Strona warsztatu poświęconego
    HTML i PHP" />
  <meta name="author" content="Tomasz Pietrzak" />
  <meta name="copyright" content="Tomasz Pietrzak" />
  <meta name="generator" content="Notepad++" />
  <meta name="robots" content="index, follow" />
  <title>Kurs HTML+PHP</title>
</head>
```

# Nagłówki i akapity

```
<p>Zwykły akapit</p>
```

```
<h1>Nagłówek pierwszego stopnia (największy)</h1>
```

```
<h2>Nagłówek drugiego stopnia</h2>
```

```
<h3>Nagłówek trzeciego stopnia</h3>
```

Wyrównanie w poziomie akapitu można określić za pomocą opcji align, która przyjmuje 4 wartości: left, right, center, justify

```
<p align="center">Zwykły akapit</p>
```

# Formatowanie tekstu

**<b>Tekst pogrubiony</b>**

<u>Tekst podkreślony</u>

*<i>Tekst pochylony</i>*

~~<del>Tekst przekreślony</del>~~

<sup><sup>Indeks górny</sup></sup>

<sub><sub>Indeks dolny</sub></sub>

# Listy

Lista numerowana:

```
<ol>  
  <li>Pierwszy element</li>  
  <li>Drugi element</li>  
  <li>Trzeci element</li>  
</ol>
```

Lista nienumerowana:

```
<ul>  
  <li>Pierwszy element</li>  
  <li>Drugi element</li>  
  <li>Trzeci element</li>  
</ul>
```

# Listy – przykład

```
<ul>
  <li>Nabiał
    <ol>
      <li>10 jaj</li>
      <li>jogurt</li>
      <li>mleko</li>
    </ol>
  </li>
  <li>Słodycze
    <ol>
      <li>cukierki</li>
      <li>czipsy</li>
      <li>batony</li>
    </ol>
  </li>
</ul>
```

- Nabiał
  1. 10 jaj
  2. jogurt
  3. mleko
- Słodycze
  1. cukierki
  2. czipsy
  3. batony

# Linki i obrazy

## linki

```
<a href="http://www.inceptio.org.pl">inceptio.org.pl</a>  
<a href="img/kaczor.png">Pokaż zdjęcie kaczora</a>  
<a href="mailto:kaczor@wp.pl">Napisz do mnie</a>
```

## obrazy

```

```

Pola *alt*, *width* i *height* są opcjonalne.



# Formatowanie tekstu: style

Znacznik typu <font> zadziała, ale został wycofany ze standardu języka. Należy używać arkuszy stylów:

## (1) Doraźnie, dla pojedynczych elementów:

```
<body style="background-color: green;">
<p>Witaj Czesławie!</p>
<p style="color: red; font-size: x-large;">Jak się masz?</p>
</body>
```



Lista nazw kolorów obsługiwana przez CSS jest tutaj:  
[http://www.w3schools.com/cssref/css\\_colornames.asp](http://www.w3schools.com/cssref/css_colornames.asp)

UWAGA: możemy zawsze zastosować specjalny znacznik <span>, który sam nic nie robi, ale możliwe jest przyporządkowanie mu konkretnego stylu.

# Formatowanie tekstu: style II

(2) Definiowane dla wszystkich znaczników w dokumencie.

```
<head>
  <!-- inne nagłówki sekcji head -->
  <style type="text/css">
    body {
      background-color: green;
    }
    p {
      color: red;
      font-size: x-large;
    }
  </style>
</head>

<body>
<p>Witaj Czesławie!</p>
<p>Jak się masz?<p>
</body>
```

Witaj Czesławie!

Jak się masz?

# Znaki specjalne

**&nbsp;** – znak odstępu  
**&lt;** – znak mniejszości „<”  
**&gt;** – znak większości „>”  
**&copy;** – znak praw autorskich ©  
**&amp;** – znak ampersandu (&)

# ***HTML+PHP #1***

## ***Wprowadzenie***

### **Zadanie**

Treść zadania do zrealizowania na pierwszych zajęciach znajduje się na stronie:

<http://www.if.pw.edu.pl/~majanik/wiki/index.php/HTML%2BPHP/Zadanie1>

# Tabele – podstawy

```
<table>
  <tr>
    <td>Pierwsza komórka</td>
    <td>Druga komórka</td>
  </tr>
  <tr>
    <td>Trzecia komórka</td>
    <td>Czwarta komórka</td>
  </tr>
</table>
```

<tr> - wiersze tabeli

<td> - komórki w wierszu „kolumny”

Pierwsza komórka Druga komórka

Trzecia komórka Czwarta komórka

# Tabele – nagłówek

```
<table>
  <tr>
    <th>Kolumna #1</th>
    <th>Kolumna #2</th>
  </tr>

  <tr>
    <td>Pierwsza komórka</td>
    <td>Druga komórka</td>
  </tr>
  <tr>
    <td>Trzecia komórka</td>
    <td>Czwarta komórka</td>
  </tr>
</table>
```

Kolumna #1	Kolumna #2
Pierwsza komórka	Druga komórka
Trzecia komórka	Czwarta komórka

**<th>** - komórka nagłówka kolumny (pogrubiona i wyśrodkowana)

# Tabele niesymetryczne

```
<table>
  <tr>
    <th>Kolumna #1</th>
    <th>Kolumna #2</th>
  </tr>

  <tr>
    <td colspan="2">Pierwsza komórka połączona</td>
    <!-- nie ma drugiej komórki -->
  </tr>
  <tr>
    <td>Trzecia komórka</td>
    <td>Czwarta komórka</td>
  </tr>
</table>
```

Kolumna #1	Kolumna #2
Pierwsza komórka połączona	
Trzecia komórka	Czwarta komórka

Analogicznie można łączyć komórki w kolumnach używając atrybutu *rowspan*.

# Young Programmer: HTML+PHP



***Powodzenia!***



**Dr inż. Małgorzata Janik**



# ***HTML+PHP #1***

## ***Wprowadzenie - Informacje dodatkowe***

Różnice HTML i XHTML: [http://wnet.strefa.pl/html/001\\_differences.html](http://wnet.strefa.pl/html/001_differences.html)

Listy z obrazkami w wypunktowaniu:

```
<ul style="list-style-image: url(nazwa.jpg);">
```