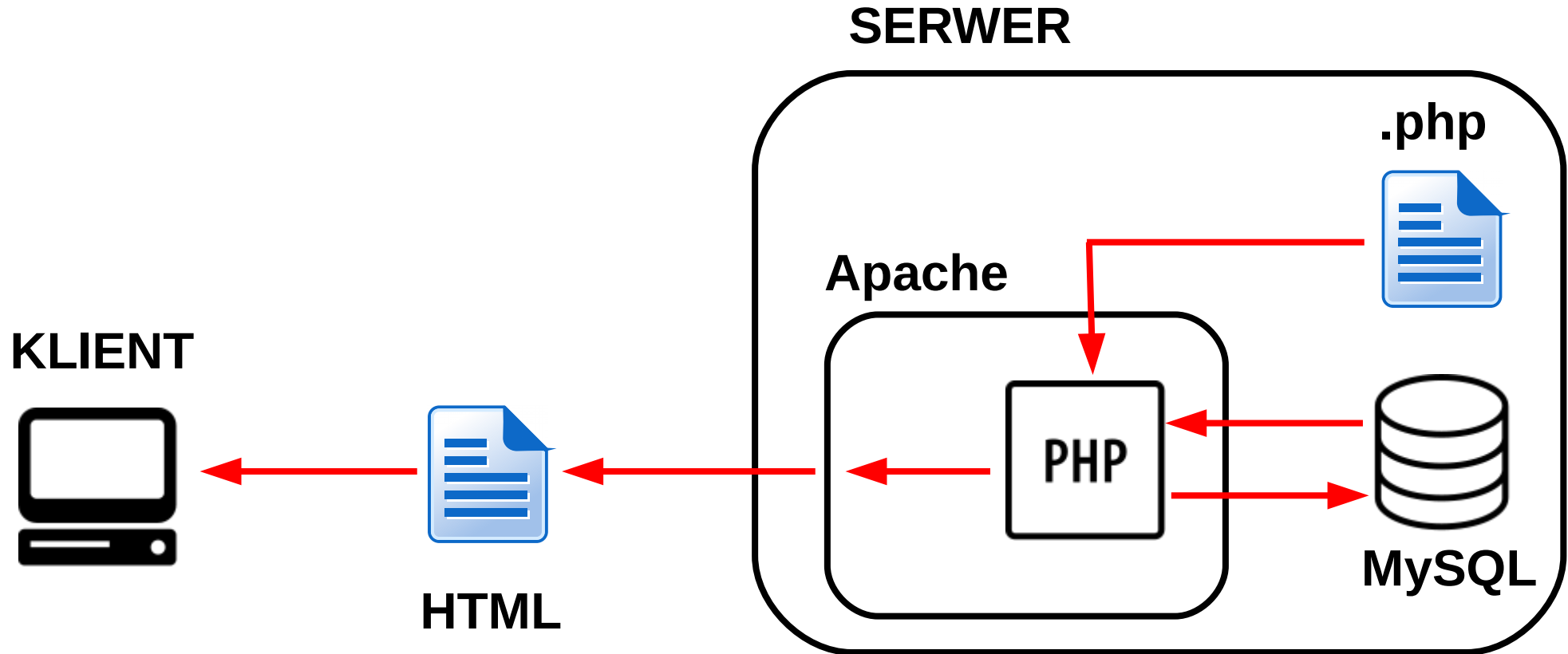


Podstawy języka PHP

Mariusz Nowak 2018

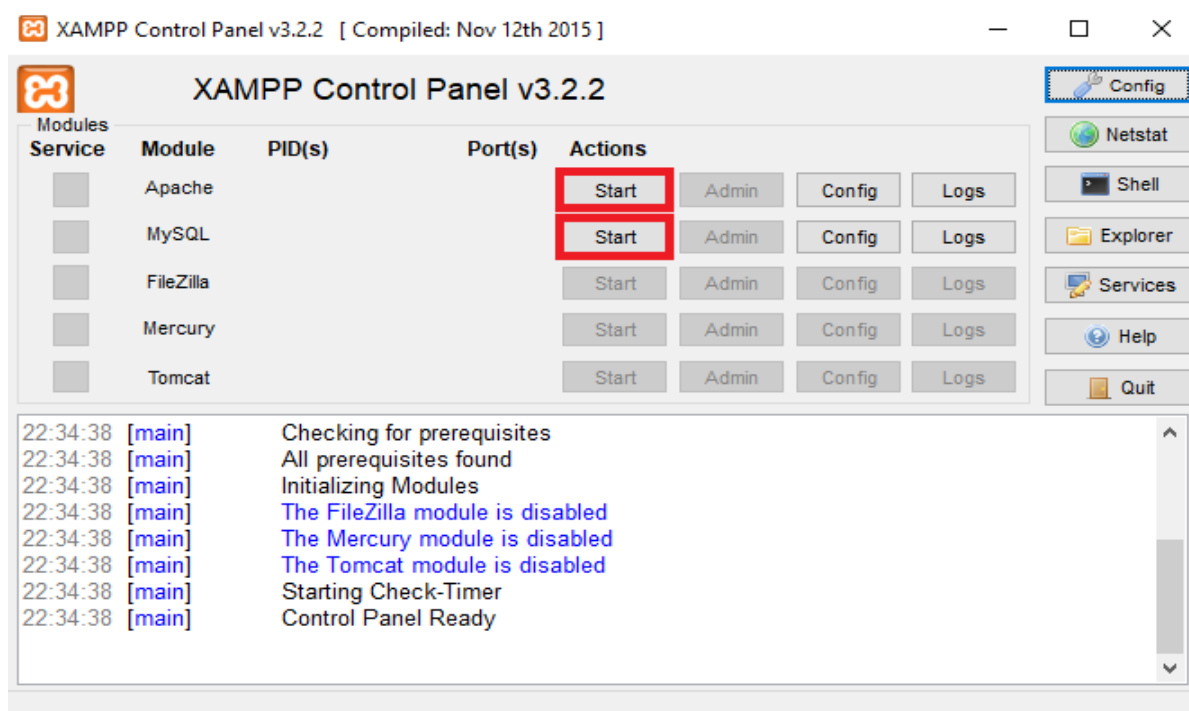


PHP: Skrypty po stronie serwera



Uruchamianie skryptów PHP (1)

1. Włączamy serwer **Apache** i **MySQL** z poziomu panelu kontrolnego **XAMPP**



XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache			Start Admin Config Logs
<input type="checkbox"/>	MySQL			Start Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

22:34:38 [main] Checking for prerequisites
22:34:38 [main] All prerequisites found
22:34:38 [main] Initializing Modules
22:34:38 [main] The FileZilla module is disabled
22:34:38 [main] The Mercury module is disabled
22:34:38 [main] The Tomcat module is disabled
22:34:38 [main] Starting Check-Timer
22:34:38 [main] Control Panel Ready

Uruchamianie skryptów PHP (2)

2. Umieszczamy stronę ze skryptem PHP w katalogu:

C:\xampp\htdocs

3. Wpisujemy w przeglądarce adres:

http://localhost/nazwa_strony.php

Pierwszy skrypt:

index.php

```
<!DOCTYPE html>
<html>
  <head> </head>
  <body>
    <?php
      echo "Witamy w języku PHP";
    ?>
  </body>
</html>
```

Początek kodu PHP

Koniec kodu PHP

Zmienne w PHP

- Nazwy zmiennych poprzedzamy znakiem dolara \$

nazwa zmiennej **wartość zmiennej**

```
<?php
$imie = "Mariusz";
echo "Twoje imię to: $imie";
?>
```

Zmienne w PHP (napisy)

- **Nazwy zmiennych** mogą zawierać litery [a-zA-Z], cyfry [0-9] i znak podkreślenia [_]

<?php

```
$Tekst_1 = "Mariusz";  
$Tekst_2 = '$abc$';
```

?>

Zmienne w PHP (liczby)

- Wielkość liter w nazwach zmiennych **ma znaczenie!**

<?php

```
$liczba = 15;  
$LICZBA = -4;
```

← liczby całkowite

```
$liczba_pi = 3.14;  
$Liczba_pi = 3.1415;
```

← liczby
zmiennoprzecinkowe

?>

Wartość logiczna, wartość null

<?php

`$prawda = true;` ← wartość: prawda (1)

`echo $prawda;` ← wyświetli: 1

`$falsz = false;` ← wartość: fałsz (0)

`echo $falsz;` ← nic nie wyświetli!!!

`$zmienna = null;` ← wartość „pusta”

`echo $zmienna;` ← nic nie wyświetli!!!

?>

Niedozwolone nazwy zmiennych

<?php

```
$9liczba = -4;  
$liczba = 15;
```

błąd: nazwa zaczyna się od cyfry

OK

```
$fałsz = false;  
$falsz = false;
```

błąd: nazwa zawiera polski znak

OK

```
$liczba 1 = 3.1415;
```

błąd: spacja w nazwie

?>

Komentarze w PHP

```
<?php
    // $tekst_1 = "Mariusz";

    /*
    $LICZBA = -4;
    $liczba_pi = 3.14;
    */

    # $nie_prawda = false;
?>
```

Jak jest generowana strona?

PHP

HTML

Widok końcowy

```
<?php  
echo "aa";  
echo "bb";  
?>
```

aabb

aabb

```
<?php  
echo "aa<br>";  
echo "bb";  
?>
```

aa
bb

aa
bb

Operatory arytmetyczne

```
<?php
```

```
$x = 10;
```

```
$y = 4;
```

```
$wynik1 = $x + $y; //dodawanie, wynik 14
```

```
$wynik2 = $x - $y; //odejmowanie, wynik 6
```

```
$wynik3 = $x * $y; //mnożenie, wynik 40
```

```
$wynik4 = $x / $y; //dzielenie, wynik, 2.5
```

```
?>
```

Operatory arytmetyczne (cd)

```
<?php
```

```
$x = 10;
```

```
$y = 4;
```

```
//reszta z dzielenia, wynik = 2
```

```
$wynik5 = $x % $y;
```

```
//potęgowanie, wynik = 10000
```

```
$wynik6 = $x ** $y;
```

```
?>
```

Inkrementacja (++)

```
<?php
    $x = 10;
    $x++;           //$x = 11;
    ++$x;          //$x = 12;

    $y = 10;
    echo $y++;     //wyświetli: 10
    echo $y;      //wyświetli: 11

    echo ++$y;    //wyświetli: 12
?>
```

Dekrementacja (--)

```
<?php
    $x = 10;
    $x--;           // $x = 11;
    --$x;          // $x = 12;

    $y = 10;
    echo $y--;     // wyświetli: 10
    echo $y;       // wyświetli: 9

    echo --$y;     // wyświetli: 8
?>
```


Operator konkatenacji (kropka)

- Kropka (.) służy do **łączenia** napisów

```
<?php
```

```
$a = "Ala ma ";
```

```
$b = "kota";
```

```
$c = $a . $b; // $c = Ala ma kota
```

```
// $d = Kot ma Alę
```

```
$d = "Kot ma " . "Alę";
```

```
?>
```

Operator kontroli błędów (@)

- Operator @ chroni przed wyświetleniem błędu dla danego wyrażenia

```
<?php
```

```
$x = (5 / 0); //wyświetli BŁĄD!!!  
//dzielenie przez 0
```

```
$x = @(5 / 0); //nadal błąd, ale  
//bez komunikatu
```

```
?>
```

Tablice

Element 0

Element 1

Element 2

```
<?php
```

```
$auta = ["Audi", "Ope1", "Kia"];
```

```
echo $auta[0]; //wyświetli: Audi  
$auta[2] = "VW"; //zmiana z Kia na VW
```

```
$auta[] = "Ford"; //dodanie elementu na  
//koniec tablicy
```

```
$auta[10] = "Fiat"; //dodanie elementu  
//nr 10
```

```
?>
```

Tablice (inny sposób definiowania)

```
<?php
```

```
    $auta = array("Audi", "Opel", "VW");
```

```
?>
```

```
<?php
```

```
    $auta[] = "Audi";           //dodanie elementu 0
```

```
    $auta[] = "Opel";          //dodanie elementu 1
```

```
    $auta[] = "VW";            //dodanie elementu 2
```

```
?>
```

Tablice – funkcja count()

```
<?php
```

```
$auta[] = "Audi";  
$auta[] = "BMW";  
$auta[] = "VW";
```

```
//wyświetli ilość elementów w tablicy: 3  
echo count($auta);
```

```
?>
```

Tablice asocjacyjne

Klucz



Wartość



```
<?php
```

```
$osoba = [ 'imie'      => 'Marcin',  
           'nazwisko' => 'Kowalski',  
           'email'    => 'poczta@xyz.pl' ];
```

```
echo $osoba[ 'imie' ]; //wyświetli: Marcin
```

```
//zmiana elementu spod klucza email:  
$osoba[ 'email' ] = 'adres@poczta.pl';
```

```
?>
```

Tablice asocjacyjne (cd)

<?php

```
$osoba['imie'] = 'Marcin';  
$osoba['nazwisko'] = 'Kowalski';  
$osoba['email'] = 'poczta@xyz.pl';
```

```
$osoba[1] = 'dowolna_wartość';  
$osoba[-10] = true;  
$osoba[1.5] = 10;
```

?>

Funkcje: var_dump(), isset()

```
<?php
```

```
$liczba = 10.5;
```

```
//wyświetli informacje o zmiennej $liczba:  
var_dump($liczba);
```

```
//zwróci true, jeżeli dana zmienna  
//instnieje lub false jeżeli nie:  
echo isset($liczba);  
echo isset($zmienna);
```

```
?>
```


Obsługa formularzy – get

```
<form method="get">  
  Podaj imię:<input type="text" name="imie">  
  <input type="submit" value="Wyślij"> <br>  
</form>
```

Podaj imię:

Obsługa formularzy – get (cd)

```
<form method="get">  
  Podaj imię:<input type="text" name="imie">  
  <input type="submit" value="Wyślij"> <br>  
</form>
```

```
<?php  
  @$imie = $_GET['imie'];  
  echo "Twoje imię to: $imie";  
?>
```

Obsługa formularzy – get (cd)

- Zmienna **\$_GET** jest tablicą zawierającą wszystkie dane z formularza wypełnionego przez użytkownika
- Parametry i wartości przekazywane są **jawnie** za pomocą adresu URL:

`http://localhost/strona.php?imie=Mariusz`

Obsługa formularzy – post

```
<form method="post">  
  Podaj imię:<input type="text" name="imie">  
  <input type="submit" value="Wyślij"> <br>  
</form>
```

```
<?php  
  @$imie = $_POST['imie'];  
  echo "Twoje imię to: $imie";  
?>
```

Obsługa formularzy – post (cd)

- Zmienna **\$_POST** jest tablicą zawierającą wszystkie dane z formularza wypełnionego przez użytkownika
- Parametry i wartości przekazywane są **niejawnie**:

`http://localhost/strona.php`

Przekazywanie danych do innej strony

```
<form method="get" action="strona2.php">  
  Podaj imię:<input type="text" name="imie">  
  <input type="submit" value="Wyślij"> <br>  
</form>
```

- Atrybut **action** znacznika **<form>** określa nazwę strony ze skryptem do którego zostaną przesłane dane z formularza:

<http://localhost/strona2.php?imie=Mariusz>

Instrukcja warunkowa if

```
if (wyrażenie) {  
    // instrukcje;  
}
```

```
if (wyrażenie) {  
    // instrukcje;  
}  
else {  
    // instrukcje;  
}
```

```
if (wyrażenie) {  
    // instrukcje;  
}  
elseif {  
    // instrukcje;  
}  
else {  
    // instrukcje;  
}
```

Operatory porównania

== równe

!= różne

> większe

>= większe lub równe

< mniejsze

<= mniejsze lub równe

=== identyczne (wartość i typ)

!== nieidentyczne (wartość i typ)

Instrukcja wyboru switch

```
switch ($zmienna) {  
    case ? :  
        //wykonaj jeżeli $zmienna = ?  
        break;  
    case ? :  
        //wykonaj jeżeli $zmienna = ?  
        break;  
    default :  
        //wykonaj domyślnie  
}
```