

Spletne tehnologije, UL, FRI (VSP)

P7 PHP

doc.dr. Mira Trebar

Vsebina

▶ PHP

- ▶ Stavki: pogojni, ponavljalni, izbira
- ▶ Funkcije
- ▶ Nizi, Polja, Razredi
- ▶ Obrazec
- ▶ Datoteke in direktoriji



▶ Literatura:

- ▶ Meloni, Julie C., PHP, MySQL and Apache (All in One), USA 2012
- ▶ Nixon, Robin, Learning PHP, MySQL&JavaScript, 4th Edition, 2014
- ▶ <http://www.w3schools.com/php/default.asp>

▶ Orodje (Lokalni strežnik):

- ▶ XAMPP - Windows, Linux, OS X (<https://www.apachefriends.org/index.html>)
- ▶ WAMP - A Windows Web development environment for Apache, MySQL, PHP databases (<http://sourceforge.net/projects/wampserver/>)
- ▶ MAMP – for Windows and Mac OS X (<http://www.mamp.info/en/downloads/>)

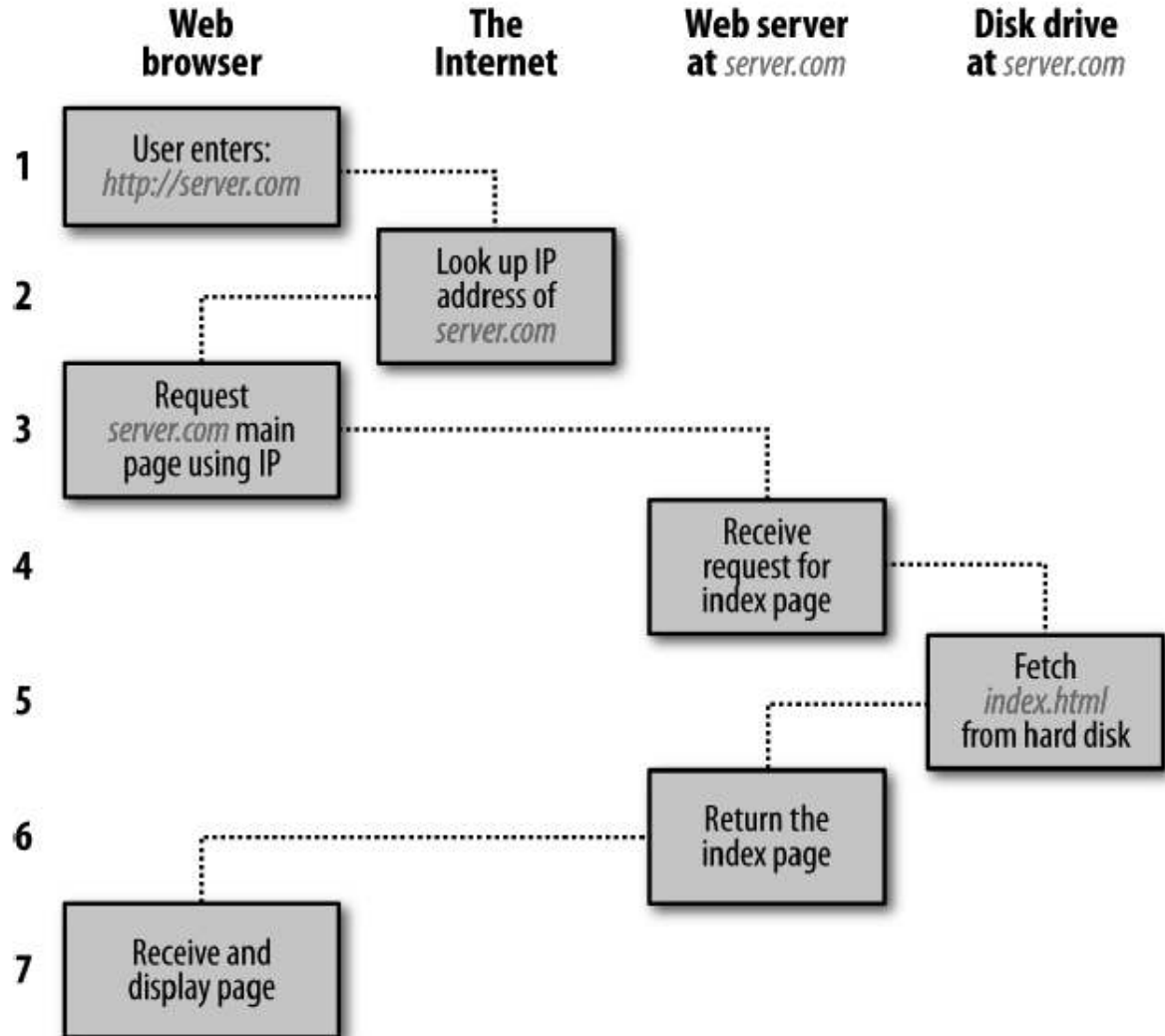
Uvod

- ▶ "PHP: Hypertext Preprocessor"
- ▶ Kreiral ga je Rasmus Lerdorf, 1994 (PHP: Personal Home Page)
- ▶ Verzija 7.x, od 2015 dalje
- ▶ Razširjen odprto-kodni skriptni jezik
- ▶ PHP skripte se izvajajo na strežniku
- ▶ Datoteke PHP:
 - ▶ imajo končnico .php
 - ▶ lahko vsebujejo: tekst, HTML, JavaScript, kodo PHP
- ▶ Rezultat izvedene kode PHP je vrnjen brskalniku kot HTML.
- ▶ Kompatibilen s strežniki: Apache, IIS, ...
- ▶ Deluje na različnih platformah: Windows, Linux, Unix, Mac OS,...
- ▶ Podpira različne podatkovne baze: MySQL, ...

-
- ▶ Generiranje dinamičnih vsebin spletnih strani
 - ▶ Delo z datotekami na strežniku:
 - ▶ create,
 - ▶ open,
 - ▶ read, write,
 - ▶ delete,
 - ▶ close
 - ▶ Pridobivanje podatkov iz obrazca
 - ▶ Delo s piškoti (angl. Cookie): send, receive
 - ▶ Podatkovna baza - PHP omogoča: add, delete, modify
 - ▶ Nadzor dostopa uporabnikov
 - ▶ Kriptiranje podatkov
 - ▶ Posredovanje HTML (slike, pdf datoteke, video, ..), tekst, XML

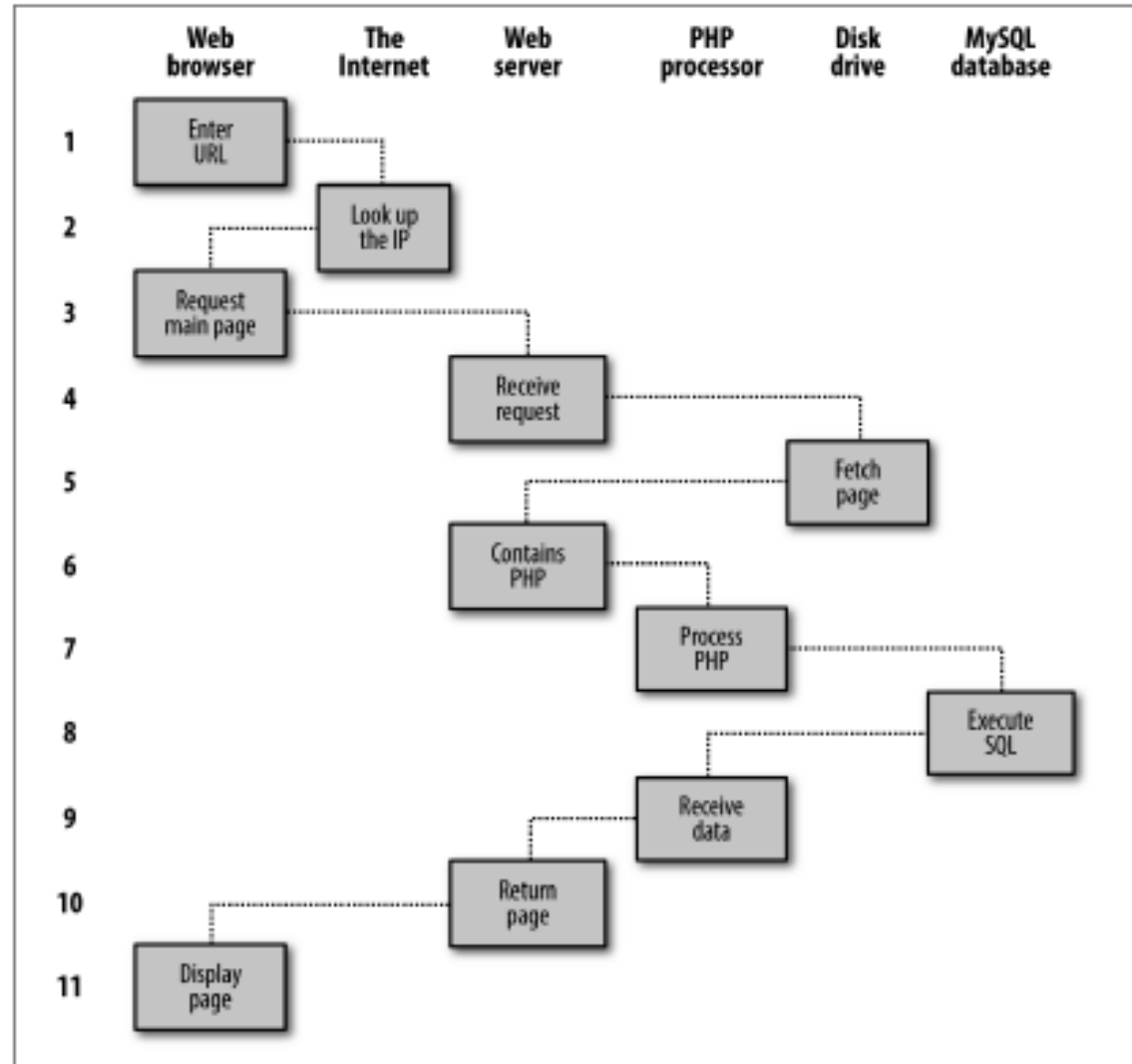
Zahteva-Odgovor (HTML)

- ▶ Brskalnik zahteva spletno stran
- ▶ Strežnik vrne spletno stran



Zahteva – Odgovor (PHP, MySQL)

- ▶ Brskalnik zahteva spletno stran
- ▶ Vključena sta
 - ▶ PHP
 - ▶ podatkovna baza
- ▶ Strežnik vrne spletno stran



PHP sintaksa

- ▶ Datoteka .php vsebuje:

- ▶ značke HTML
- ▶ skriptno kodo PHP

- ▶ Samostojna koda PHP

```
<?php
    // ... koda PHP
?>
```

- ▶ PHP stavki se končajo s podpičjem (;)

- ▶ Komentar:

- ▶ Ena vrstica: //, #,
- ▶ Več vrstic: /* ...*/

- ▶ Izpis na zaslon z vgrajenima funkcijama:

- ▶ echo
- ▶ print

Primer: Koda PHP je v HTML

```
<!DOCTYPE html>
<html>
<head> </head>
<body>
    <p> Prva PHP stran </p>
    <?php
        echo "My name is ...";
    ?>
</body>
</html>
```

PHP - konstante

- ▶ Konstante (ang. Constants) so globalne
- ▶ Ni jih mogoče brisati, ko so enkrat definirane
- ▶ Konstanta je ime za vrednost
- ▶ Ime konstante se prične s črko
- ▶ Sintaksa:

`define(name, value, case-insensitive)`

- ▶ *name*: ime konstante
 - ▶ *value*: vrednost konstante
 - ▶ *case-insensitive*: odvisnost od velikih in malih črk, default=false
- ▶ Primer:

```
<?php
    define("GREETING", "Welcome!");           // GREETING = "Welcome!"
    define("alfa", 90);                        // alfa=90
?>
```


PHP - spremenljivke

- ▶ Spremenljivke (ang. Variables) se pričnejo se z znakom \$
- ▶ Deklaracija tipa ni potrebna
- ▶ Imena spremenljivk:
 - ▶ Pričnejo se s črko (a-z, A-Z) ali cifro (0-9) ali podčrtaj (_)
 - ▶ Občutljiva na velikost črk (\$aI, \$AI – različni spremenljivki)
- ▶ Primeri:
 - \$x;
 - \$num1=5; (integer)
 - \$num2 = 23.5; (double, decimalno ločilo je pika)
 - \$userName = "MihaPetek"; (string, tekst je v navednicah)
- ▶ Funkciji za izpis spremenljivk: echo in print
 - \$txt ="spremenljivke";
 - echo "PHP " . \$txt;
 - print "PHP " . \$txt;
 - \$x = 5; \$y = 4;
 - echo (\$x + \$y);

► **Tipi spremenljivk:**

- Lokalne (angl. Local) – deklarirane v funkcijah
- Globalne (angl. Global) – deklarirane izven funkcij
- Statične (angl. Static) – deklarirane v funkciji

► **Superglobalne spremenljivke (ang Superglobal variables)**

- `$_GET` – spremenljivke zagotovljene skripti z metodo GET
- `$_POST` – spremenljivke zagotovljene skripti z metodo POST
- `$_COOKIE` – spremenljivke zagotovljene skripti skozi piškot (cookie)
- `$_FILES` – spremenljivke zagotovljene skripti skozi nalaganje datotek
- `$_SERVER` – informacija kot je glava, poti datotek, in lokacije skript
- `$_ENV` - – spremenljivke zagotovljene skripti kot del strežniškega okolja
- `$_REQUEST` – spremenljivke zagotovljene skripti z GET, POST in COOKIE vhodne mehanizme
- `$_SESSION` – spremenljivke, ki so trenutno registrirane v seji

PHP - podatkovni tipi

▶ Podatkovni tipi

- ▶ Boolean (bool): `$x = true; $y=false;`
- ▶ Integer (integer): `$x = 5985;`
- ▶ Float (double): `$x = 10.365;`
- ▶ String (string): `$x = "Hello!";`
- ▶ Object: primeri so v nadaljevanju
- ▶ Array: `$cars = array("Volvo","BMW","Toyota");`
- ▶ NULL (spremenljivka je kreirana brez vrednosti)
- ▶ Resource: klic podatkovne baze

▶ Spreminjanje podatkovnega tipa

`settype($variabletochange, 'new type');`

Primer: `$kpi = 3.14;`

`settype($kpi, 'string');` (sprememba iz double -> string);

▶ Testiranje podatkovnega tipa: `is_null()`, `is_bool()`, `is_int()`, ...

Primer: `echo "is a string? ".is_string($kpi);`

PHP - operatorji in izrazi

- ▶ Prireditveni operator in izraz: `$name = "Jaka";`
- ▶ Aritmetični operatorji: `+, -, /, *, %` in `**` (PHP 5.6)
- ▶ Sestavljeni prireditveni operatorji: `+=, -=, /=, *=, %=`
- ▶ Inkrement in dekrement: `++, --`
`++$a, $a++, -$a, $a--`
- ▶ Primerjalni operatorji: `==, ===, !=, !==, <>, >, >=, <, <=`
- ▶ Logični operatorji: `||` - or, `&&` - and, `!` – not
`and, or, xor`
- ▶ Konkatencija ali združevanje: `$x1 . $x2` (operacije nad nizi)
`$x1 .= $x2`
- ▶ Operatorji za polje (array): `+, ==, ===, !=, !==, <>`
- ▶ Primeri: https://www.w3schools.com/php/php_operators.asp

PHP - pogojni stavki (condition)

IF: if (*expression*) { // koda
 }

IF-ELSE: if (*expression*) { // koda
 }
 else { // koda
 }

IF-ELSEIF-ELSE: if (*expression*) { // koda
 }
 elseif (*another expression*) { // koda
 }
 else { // koda
 }

- ▶ Ternarni operator (podoben if) – uporabi vprašaj in dvopičje (? :)
(expression) ? returned_if_expression_is_true : returned_if_expression_is_false;

PHP- zanke (loops)

► Stavek **while**

```
while (expression) {  
    // do something  
}
```

► Stavek **do ... while**

```
do {  
    // code to be executed  
} while (expression);
```

► Prekinitev zank: **break**, **continue**

► Gnezdenje zank

```
<?php  
$x = 1;  
while($x <= 5) {  
    echo "x = " . $x . " <br>";  
    $x++;  
}
```

```
$x = 1;  
do {  
    echo "x = " . $x . " <br>";  
    $x++;  
} while ($x <= 5);  
?>
```

► **Stavek for**

```
for (init counter; test counter; increment counter) {  
    // code to be executed  
}
```

► **Stavek foreach**

```
foreach ($array as $value) {  
    // code to be executed;  
}
```

► **Prekinitivni zank:**

- break,
- continue

► **Gnezdenje zank**

```
<?php  
for ($x = 0; $x <= 10; $x++) {  
    echo "The number is: $x <br>";  
}  
?>
```

```
<?php  
$colors = array("red", "green", "blue");  
  
foreach ($colors as $value) {  
    echo "$value <br>";  
}  
?>
```

PHP- izbira (switch)

```
switch (expression) {  
    case result1:  
        // execute this if expression results in result1  
        break;  
    case result2:  
        // execute this if expression results in result2  
        break;  
    ...  
    default:  
        // execute this if no break statement  
        // has been encountered  
}
```


PHP- funkcije

- ▶ Uporabnik napiše svoje funkcije (male in velike črke- neodvisno)

- ▶ Klic funkcije:

```
some_function($a1_argument, $a2_argument);
```

- ▶ Definicija funkcije:

```
function some_function($argument1, $argument2)
{
    // koda
    return $result; // opcijsko
}
```

- ▶ Vračanje vrednosti iz funkcije uporabnika - return:

lokalna spremenljivka je \$result

-
- ▶ Spremenljivke:
 - ▶ Deklarirane v funkciji – lokalne
 - ▶ Dostop do globalnih spremenljivk v funkciji: `global $life;`
 - ▶ Program za izpis globalne spremenljivke:

```
<?php
    $life=42;                // globalna spremenljivka
    function meaningOfLife()
    {
        global $life;        // globalna spremenljivka
        echo "The meaning of life is ".$life;
    }
    meaningOfLife();
?>
```

-
- ▶ Shranjevanje stanja med klici funkcij:

```
function numberedHeading() {  
    static $num_of_calls = 0;  
    $num_of_calls++;  
    ...  
}  
  
numberedHeading();    // 1.klic ($num_of_calls = 1)  
numberedHeading();    // 1.klic ($num_of_calls = 2)
```

- ▶ Argumenti funkcij:

- ▶ Privzete vrednosti (addfive.php)
(function addFive(\$num))
- ▶ Prenos referenc spremenljivke funkciji (addfive2.php)
(function addFive(&\$num)) - znak & pred argumentom

- ▶ Testiranje obstoja funkcije, vrne true ali false:

```
function_exists() (exists.php)
```

Primer:

- Dostop do globalne spremenljivke v funkciji

```
<?php
$x = 5;
$y = 10;

function myTest() {
    global $x, $y;
    $y = $x + $y;
}

myTest();
echo $y;    // 15
?>
```

GLOBALS[index],
index je ime spremenlj,
spreminjanje \$x in \$y
poteka v funkciji

```
<?php
$x = 5;
$y = 10;

function myTest() {
    $GLOBALS['y']
    = $GLOBALS['x']
    + $GLOBALS['y'];
}

myTest();
echo $y;    // 15
?>
```

- Spremenljivka \$x se ne briše po izvedbi funkcije

```
<?php
function myTest() {
    static $x = 0;
    echo $x;
    $x++;
}

myTest();    // 0
myTest();    // 1
myTest();    // 2
?>
```

PHP - polje (Array)

- ▶ Polje sestavljata:
 - ▶ ključ (key)
 - ▶ vrednost (value)
- ▶ Kreiranje polja:
 - ▶ **Numerično indeksirano polje** (ključ je indeks pozicije, začetna vrednost je 0 , nato se povečuje za 1)

```
$color=array("red","blue" );  
$color[ ] ="red"; $color[ ] ="blue";  
$color[0] ="red"; $color[1] ="blue";  
echo $color[1];                // vrne "blue"
```

- ▶ **Asociativno polje:** (ključ je dejansko ime)

```
$person=array("name" => "Bob","age" => "43");  
echo $person["name"];          // vrne "Bob"
```

► **Večdimenzionalno polje - 2D polje:**

```
$cars = array (  
    array("Volvo",22,18),  
    array("BMW",15,13),  
    array("Saab",5,2),  
    array("Land Rover",17,15)  
);
```

```
// Izpis 2D polja v seznamu  
for ($row = 0; $row < 4; $row++) {  
    echo "<p><b>Row number $row</b></p>";  
    echo "<ul>";  
    for ($col = 0; $col < 3; $col++) {  
        echo "<li>".$cars[$row][$col]."</li>";  
    }  
    echo "</ul>";  
}
```

Row number 0

- Volvo
- 22
- 18

Row number 1

- BMW
- 15
- 13

Row number 2

- Saab
- 5
- 2

Row number 3

- Land Rover
- 17
- 15

-
- ▶ Funkcije za sortiranje polja:
 - ▶ `sort()` – naraščajoč vrstni red
 - ▶ `rsort()` – padajoč vrstni red
 - ▶ `asort()` – naraščajoč vrstni red v asociativnem polju glede na vrednost
 - ▶ `ksort()` – naraščajoč vrstni red v asociativnem polju glede na ključ
 - ▶ `arsort()` – padajoč vrstni red v asociativnem polju glede na vrednost
 - ▶ `krsort()` - padajoč vrstni red v asociativnem polju glede na ključ

- ▶ Primer:

```
$cars = array("Volvo", "BMW", "Toyota");  
sort($cars);           // ("BMW", "Toyota", "Volvo")
```

```
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");  
arsort($age);
```

-
- ▶ `count()`, `sizeof()` – število elementov v polju
 - ▶ `reset()` – notranji kazalec polja postavi na prvi element
 - ▶ `current()` – vrne trenutni element v polju
 - ▶ `array_push()` – vstavi enega ali več elementov na konec polja
 - ▶ `array_pop()` – briše zadnji element v polju
 - ▶ `array_product()` – izračuna zmnožek vrednosti v polju
 - ▶ `array_shift()` – odstrani prvi element in vrne njegovo vrednost
 - ▶ `array_merge()` – združi eno ali več polj v eno polje
 - ▶ `array_keys()` – vrne vse ključe polja
 - ▶ `array_key_exists()` – preveri obstoj določenega ključa v polju
 - ▶ `array_values()` – vrne vse vrednosti polja
 - ▶ `array_intersect()` – primerja polja in vrne ujemanje (primerja vrednosti)
 - ▶ `array_intersect_key()` – primerja polja in vrne ujemanje (primerja ključe)
 - ▶ ... https://www.w3schools.com/php/php_ref_array.asp

Primer: 1D in 2D numerično polje

- ▶ Zapis polja – 1 varianta

```
$team = array("Bill","Mary","Mike","Chris","Anne");  
echo $team[3];    // Vrne Chris
```

- ▶ Zapis polja – 2 vrianta

```
$team = array(  
    1 ⇒ "Bill",  
    2 ⇒ "Mary",  
    3 ⇒ "Mike",  
    4 ⇒ "Chris",  
    5 ⇒ "Anne"  
);
```

- ▶ 2D polje

```
$oxo = array(  
    array("x"," ","o"),  
    array("o","o","x"),  
    array("o","x"," ")  
);
```

Primer: Asociativno 2-dimenzionalno polje (2D)

► Numerični ključi

```
<?php
$a=array(0=>"red",1=>"green",2=>"blue");
echo array_shift($a)."<br>";    // odstrani prvi element
print_r ($a);
?>
```

► Ključi so imena

```
<?php
$a=array("a"=>"red","b"=>"green","c"=>"blue");
echo array_shift($a) . "<br>";
print_r ($a);
?>
```

Izpis: red

Array ([b] => green [c] => blue)

PHP- objekti

Kreiranje objekta (Creating an Object) - Deklaracija obstoja objekta:

```
class myClass {  
    //code  
}
```

Primerek objekta:

```
$object1 = new myClass();
```

(proofofclass.php)

- ▶ Lastnosti objektov (Properties of Objects), vrednosti, polja, drugi razredi
 - ▶ Prikaz lastnosti

```
<?php class myCar {  
    public$color = "silver";  
    public$make = "Mazda";  
    public$model = "Protege5";  
}
```

```
$car=new myCar();  
echo "I drive a: ".$car->color." ".$car->make." ".$car->model;  
?> (objproperties.php)
```

► Spreminjanje lastnosti objektov:

```
<?php class myCar {  
    public$color = "silver";  
    public$make = "Mazda";  
    public$model = "Protege5";  
}  
$car=new myCar();  
$car->color = "red"  
$car->make = "Renault"  
$car->model = "Clio";  
echo "I drive a: ".$car->color." ".$car->make." ".$car->model;  
?> (objproperties2.php)
```

-
- ▶ Metode objekta (funkcije definirane znotraj razreda)
 - ▶ Razred z metodo (helloclass.php)
 - ▶ Dostopanje lastnosti razreda znotraj metode (helloclass2.php)
 - ▶ Spreminjanje vrednosti lastnosti (helloclass3.php)
 - ▶ Konstruktorji - funkcije v razredu

Dedovanje objekta (Object Inheritance)

- ▶ Razred deduje funkcionalnosti iz svojega starša (inheritance.php)
- ▶ Metoda razreda otrok poveže tisto od starša (inheritance2.php)

PHP- string

- ▶ Formatiranje stringa s PHP (podobno kot v C prog. jeziku)

- ▶ Funkcija printf()

`printf("This is my number:%o", 55);` // Izpis: 67 (osmiško je $6*8+7$)

`printf("%04d", 36);` // prazna mesta zapolnijo 0 - Izpis: "0036"

- ▶ Tip izpisa:

- ▶ d- decimal (argument kot decimalno število)
- ▶ b- binary (integer kot celo število)
- ▶ c- ASCII (integer kot ASCII ekvivalent)
- ▶ f- floating-point (double), integer kot število s plavajočo vejico
- ▶ o- octal (b=8), integer kot osmiško število
- ▶ s- string (argument kot niz)
- ▶ x- hexadecimal (integer kot šestnajstiško število z malimi črkami (b=16),
- ▶ X- hexadecimal (integer kot šestnajstiško število z velikimi črkami (b=16))

```
$number = 543;
printf("Decimal: %d<br/>", $number);
printf("Binary: %b<br/>", $number);
printf("Double: %f<br/>", $number);
printf("Octal: %o<br/>", $number);
printf("String: %s<br/>", $number);
printf("Hex (lower): %x<br/>", $number);
printf("Hex (upper): %X<br/>", $number);
```

```
Decimal: 543
Binary: 1000011111
Double: 543.000000
Octal: 1037
String: 543
Hex (lower): 21f
Hex (upper): 21F
```

- ▶ Shranjevanje formatiranega niza za kasneje-funkcija sprintf()

```
$cash=sprintf("%.2f",23.33456);
echo "You have \$$cash to spend." //You have $23.33 to spend.
```

- ▶ Zapis niza

```
$test = "phpcoder";
echo $test[0]; // izpiše "p"
```

► Niz:

```
$membership = "pAB7";
```

► Dolžina stringa: strlen()

```
strlen($membership)
```

► Substring: strstr()

```
strstr($membership, "AB")
```

► Pozicija substringa: strpos()

```
strpos($membership, "mz")
```

► Sprememba stringa v velike črke: strtoupper()

```
strtoupper("Hello Web!"); // Izpiše se: HELLO WEB!
```

► Izloči del stringa: substr()

```
$test = "phpcoder";
```

```
echo substr($test,3)."<br/>"; // prints "coder"
```

► Več funkcij:

https://www.w3schools.com/php/php_ref_string.asp

PHP- datum in čas

- ▶ `time()` -

```
echo time();
```

```
// sample output: 1326853185, // this represents January 17, 2012 at 09:19PM
```

- ▶ `getdate()`

```
$date_array = getdate(); // no argument passed so today's date will be used
```

```
foreach ($date_array as $key => $val) {
```

```
    echo "$key = $val<br>";
```

```
}
```

```
echo "<p>Today's date: ".$date_array['mon']. "/" . $date_array['mday']. "/" .
```

```
    $date_array['year']. "</p>";
```

- ▶ `date()`

- ▶ `mktime()`

- ▶ `checkdate()`

PHP- obrazci

- ▶ Kreiranje obrazca – submit - izpis podatkov
 - ▶ HTML koda (simpleform.html)
 - ▶ PHP koda (send_simpleform.php)
- ▶ Obrazec – polje
 - ▶ `<input type="checkbox" id="products" name="products">`
 - ▶ HTML koda (formwithcb.html)
 - ▶ PHP koda (send_formwithcb.php)
- ▶ Združitev HTML in PHP kode
 - ▶ Primer: Ugibanje števila: (numguess.php)
- ▶ Skrita polja – shranjevanje stanja (numguess2.php)
- ▶ Preusmeritev uporabnika – ustvarjen je nov dokument .html
- ▶ Pošiljanje sporočila (Mail) ob submit:
 - ▶ HTML koda (feedback.html)
 - ▶ PHP koda (sendmail.php)

PHP- Datoteke in direktoriji

- ▶ Vključevanje datotek (običajno php skripte)

```
<?php  
    include 'myinclude.php';  
?>
```

- ▶ Vračanje vrednosti iz vključene datoteke

```
$addResult = include 'returnvalue.php';
```

- ▶ Vključitev v pogojne stavke in zanke: if, for,

```
$test = "1";  
if ($test == "2") {  
    include 'file.txt'; // won't be included  
}
```

- ▶ Uporaba: include_once, include_path

- ▶ include_once '/home/user/bob/htdocs/project4/lib/mylib.inc.php';
- ▶ include_path './home/user/bob/htdocs/project4/lib/

► Validacija datotek: `file_exists()`

```
if (file_exists('test.txt')) {  
    echo "The file exists!";  
}
```

► Datoteka:

```
if (is_file('test.txt')) {  
    echo "test.txt is a file!";  
}
```

► Direktorij:

```
if (is_dir('/tmp')) {  
    echo "/tmp is a directory";  
}
```

► Status datoteke: `is_readable()`, `is_writable()`, `is_executable()`

► Velikost datoteke: `filesize()`

► Podatki o datoteki: `filemtime()`, `filectime()`

-
- ▶ Kreiranje in brisanje datotek: `touch()`, `unlink()`
 - ▶ `touch('myfile.txt');`
 - ▶ `unlink('myfile.txt');`
 - ▶ Odpiranje datotek:
 - ▶ Branje: `$fp = fopen("test.txt", "r");`
 - ▶ Pisanje: `$fp = fopen("test.txt", "w");`
 - ▶ Dodajanje: `$fp = fopen("test.txt", "a");`
 - ▶ Datoteke ni mogoče odpreti – komentar
`($fp = fopen("test.txt", "w")) or die("Couldn't open file, sorry");`
 - ▶ Branje iz datoteke: `fgets()`, `feof()`, `fread()`, `fgetc()`, `file_get_contents()`
`$fline=fgets($file, 1024);`
 - ▶ Pisanje v datoteko: `fwrite()`, `fputs()`, `file_put_contents()`
 - ▶ Zaklepanje datoteke: `flock()`

Naloga 1

1. Šahovnica: Črno bela polja

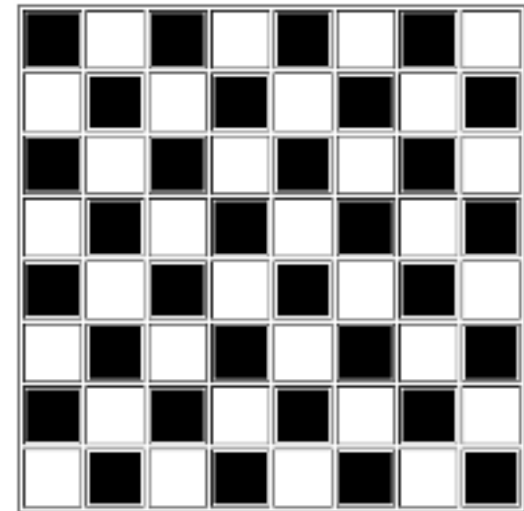
1. Slike:

1. whiteSquare.gif, blackSquare.gif
2. width='30' height='30,

2. Tabela: border = “|“

2. Šahovnica: postavitev figur

1. *P, p* - pawn - kmet ,
2. *N, n* – knight – konj,
3. *B, b* – bishop – tekač,
4. *R, r* – rook – trdnjava,
5. *Q, q* - queen - kraljica
6. *K, k* king - kralj



r	n	b	q	k	b	n	r
p	p	p	p	p	p	p	p
P	P	P	P	P	P	P	P
R	N	B	Q	K	B	N	R

Naloga 2

I. Obrazec:

1. Vnos imena in izbira proizvoda: metoda “POST”
2. PHP spremenljivka: `$_POST`
3. Izpis: seznam označenih proizvodov ali None (proizvod ni označen)

Name:

Mira

Select Some Products:

☒ Tricorder

☐ ORAC AI

☐ HAL 2000

Submit Form

Welcome, **Mira**!

Your product choices are:

- Tricorder
- HAL 2000

Welcome, **Mira**!

Your product choices are: None