

# Osnove izrade PHP aplikacija

## Uvod u PHP

Izradio: Davor Cihlar

# Plan

---

- ▶ Što ćemo naučiti?
  - ▶ Općenito o PHP
  - ▶ Naredbe za ispis
  - ▶ Komentare
  - ▶ Varijable i tipove
  - ▶ Reference
  - ▶ Stringove
  - ▶ Petlje
  - ▶ Uvjete

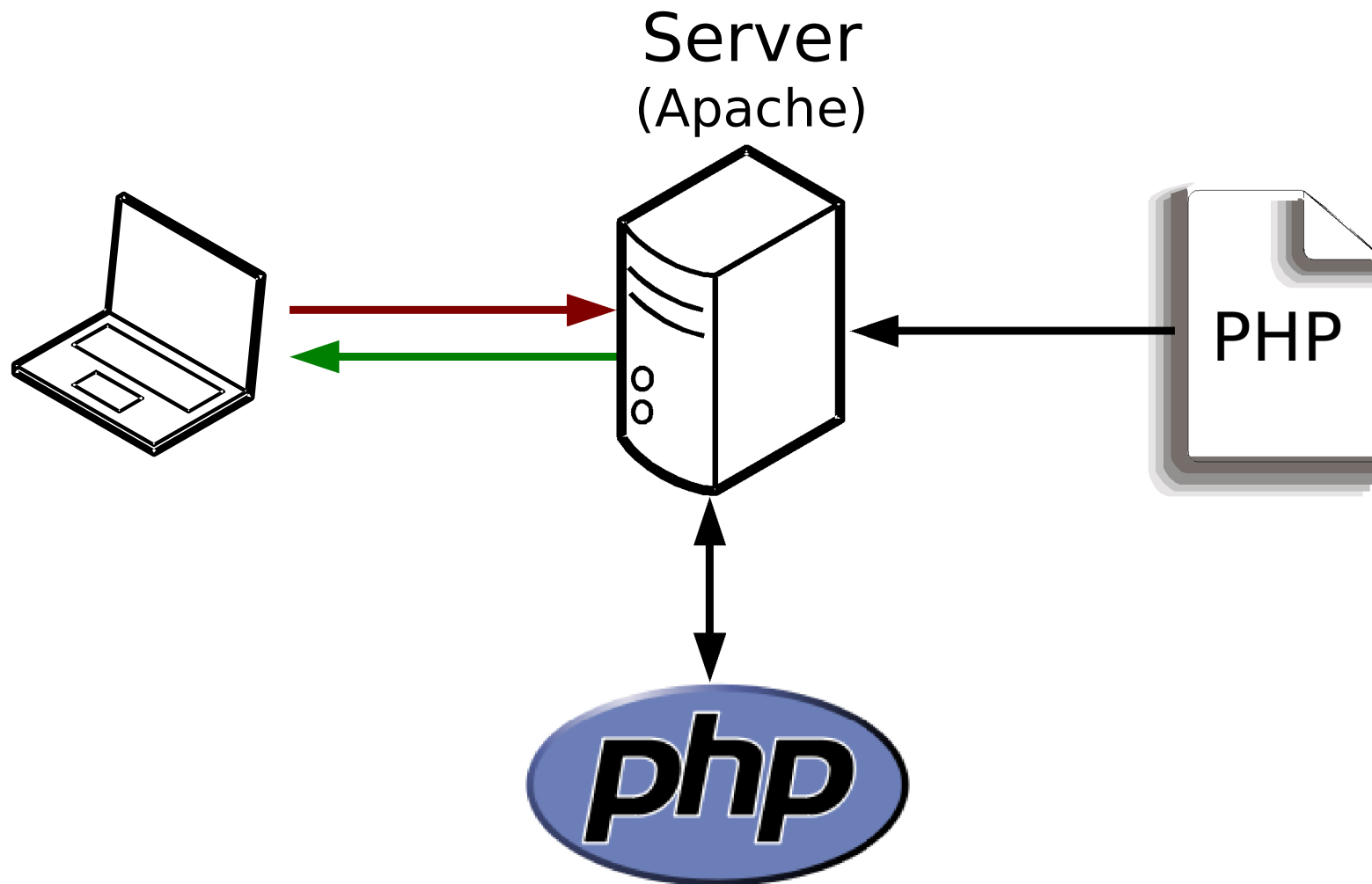
# Općenito o PHP

---

- ▶ Naziv dolazi od "PHP Hypertext Preprocessor"
  - ▶ Rekurzivna skraćenica
- ▶ Ima ekstenziju \*.php
- ▶ Interpreterski jezik
- ▶ Sintaksa slična kao u C-u
- ▶ Izvodi se **na poslužitelju**
  - ▶ Prije izvođenja prevodi se u *bytecode*
  - ▶ Moguće koristiti i kao skriptni jezik opće namjene (npr. konzolna ili grafička aplikacija), ali time se nećemo baviti
- ▶ Ne ovisi o sustavu posjetitelja *web* stranice
- ▶ Generira HTML koji se prosljeđuje posjetitelju (klijentu)
  - ▶ Klijent prikazuje HTML stranicu

# Interakcija PHP-a i klijenta

---



# Hello PHP

---

- ▶ PHP kôd nalazi se unutar specijalnih tagova
  - ▶ `<?php ... ?>` ili `<? ... ?>`
- ▶ Ti tagovi su znak interpreteru (**na poslužitelju**) da je taj dio potrebno izvršiti
- ▶ Sve izvan tih tagova direktno se prosljeđuje klijentu

- ▶ Npr.:

- ▶ Kopirano

```
<?php
    echo 'Izvršeno';
?>
```

# Zadatak (1)

---



- ▶ Preimenujte datoteku "stranica.html" u "stranica.php"
- ▶ Redak "šđčćž" zamijenite s PHP kodom:
  - ▶ 

```
<?php  
    echo 'Hello PHP!';  
?>
```
- ▶ Isprobajte!

# Komenari

---

- ▶ Tri vrste komentara
  - ▶ Jednolinijski
    - ▶ *// ovo je komentar u jednom redu*
    - ▶ *# i ovo je komentar u jednom redu*
  - ▶ Višelinijski
    - ▶ */\* Ovo je komentar u više redova \*/*

# Varijable

---

- ▶ Nije potrebno deklarirati
- ▶ Razlikuju velika i mala slova! (*case sensitive*)
- ▶ Tip se određuje pri inicijalizaciji
  - ▶ Varijabla postaje istog tipa kao i ono što u nju kopiramo
- ▶ Ispred imena varijabli mora biti znak `$`
  - ▶ `$a = 42;`
- ▶ Predlažemo *camel* način imenovanja varijabli
  - ▶ Npr.: `$ovoJeVarijabla`, `$ovoJeDrugaVarijabla`

# Brisanje varijabli

---

- ▶ Pomoću funkcije `unset`
- ▶ Najčešće nije potrebno
- ▶ Primjer:
  - ▶ `unset ( $a ) ;`

# Tipovi varijabli

---

- ▶ Postoje osnovni tipovi poput:
  - ▶ `int`, `integer` (cijeli brojevi)
  - ▶ `float`, `double`, `real` (realni brojevi)
  - ▶ `bool`, `boolean` (logička vrijednost: **true** ili **false**)
- ▶ Ali postoje i napredni tipovi:
  - ▶ `string` (znakovni nizovi)
  - ▶ `array` (polja)
  - ▶ `object` (instanca klase)
  - ▶ `resource` (npr. datoteka ili veza s bazom)
  - ▶ `null` (ništa)

# var\_dump()

---

- ▶ Funkcija koja ispiše varijablu i njen tip
- ▶ Polja ispisuje rekurzivno i to u cijelosti
- ▶ Vrlo korisno za traženje grešaka (*debug*)
- ▶ Primjer:
  - ▶ `var_dump( '42' );`
- ▶ Ispis:
  - ▶ `string(2) "42"`

# Operacije s različitim tipovima podataka

---

- ▶ Automatski se pretvaraju podaci u potreban tip
  - ▶ "1"+2 vraća 3 (tipa integer)
  - ▶ 1/2 vraća 0.5 (tipa float)
  - ▶ 4/2 vraća 2 (tipa integer!!!)
- ▶ Moguće promijeniti tip podatka kao u C-u
  - ▶ `$a = (int)"42"`
  - ▶ Najčešće nije potrebno
- ▶ Ako operacija prelazi raspon `integera`, koristi se `float` za rezultat!

# Provjere tipova

---

- ▶ Postoji niz funkcija za provjeru tipa podatka
  - ▶ `is_string($var)` - da li je `$var` tipa `string`
  - ▶ `is_integer($var)` - da li je `$var` tipa `int`
  - ▶ `is_array($var)` - da li je `$var` polje
  - ▶ ...
- ▶ Više na:
  - ▶ <http://www.php.net/manual/en/ref.var.php>

# Reference

---

- ▶ Operator = kopira sadržaj desne varijable u lijevu
  - ▶ Operacija spora za veći blok podataka
  - ▶ Podatak bi se kopirao kada bi funkcija vraćala referencu
  - ▶ Često nije poželjno kopirati instancu klase
  - ▶ ...
- ▶ Rješenje: reference (jednostavni pokazivači)
- ▶ Primjer:
  - ▶ `$a = 4;`
  - ▶ `$b = &$a;`
  - ▶ `$b = 42;`
  - ▶ `var_dump($a);`
  - ▶ `int(42)`

# Stringovi (1)

---

- ▶ Puno jednostavniji i moćniji nego u C-u
- ▶ Zapis stringova:
  - ▶ Unutar jednostrukih navodnika
    - ▶ Ne podržava posebne znakove (*escape* sekvence), osim \'
    - ▶ Ne podržava ubacivanje varijabli
  - ▶ Unutar dvostrukih navodnika
    - ▶ Podržava posebne znakove
    - ▶ Podržava ubacivanje varijabli
- ▶ Primjeri:
  - ▶ `$a = 'tekst';`
  - ▶ `$b = 'teks\t\'';`
  - ▶ `$c = "tekst";`
  - ▶ `$d = "tekst$varijabla";`
  - ▶ `$e = "te{$varijabla}kst\n";`

## Stringovi (2)

---

- ▶ Moguće pristupiti pojedinom znaku
  - ▶ `$s = "FER";`  
`$s[0] = "L";`  
`var_dump($s[1]);`
  - ▶ `string(1) "E"`
- ▶ Rezultat je opet string
  - ▶ PHP nema poseban tip char

# Spajanje stringova

---

- ▶ Pomoću posebnog operatora: .
  - ▶ `$b = 42;`  
`$a = $b . ' kn';`  
`var_dump($a);`
- ▶ Ili pomoću dvostrukih navodnika
  - ▶ `$a = "$b kn";`  
`var_dump($a);`
- ▶ Ispis (obje varijante):
  - ▶ `string(5) "42 kn"`

# Funkcije za stringove (1)

---

- ▶ `strpos($haystack, $needle, ...)`
  - ▶ Traži poziciju `$needle` u `$haystack`
  - ▶ Razlikovati poziciju `0` i `false` u slučaju da `$needle` nije pronađen!
- ▶ `strlen($string)`
  - ▶ Vraća duljinu stringa `$string` (kao i u C-u)
- ▶ `str_replace($search, $replace, $subject, ...)`
  - ▶ Zamjenjuje `$search` s `$replace` unutar `$subject`
- ▶ `substr($string, $start [, $length])`
  - ▶ Vraća podstring iz `$string` od pozicije `$start`, duljine `$length` (ili do kraja ako nije zadan)
  - ▶ Proučiti detaljnije na [php.net](http://php.net)!

# Funkcije za stringove (2)

---

- ▶ `trim($str, ...)`
  - ▶ Izbacuje praznine s lijeve i desne strane `$str`
- ▶ `chr($ascii)`
  - ▶ Vraća string sa znakom koji ima zadanu ascii vrijednost
- ▶ `ord($string)`
  - ▶ Vraća ascii vrijednost prvog znaka iz `$string`
- ▶ `htmlspecialchars($string)`
  - ▶ Omogućuje ispis znakova rezerviranih za HTML
- ▶ **Više na:**
  - ▶ <http://hr.php.net/manual/en/ref.strings.php>

# Ispis (1)

---

## ▶ echo

- ▶ Prima više argumenata odvojenih zarezom, ne vraća ništa
- ▶ Dio jezika PHP (nije prava funkcija)
  - ▶ nisu potrebne zagrade
- ▶ Najčešće se koristi
- ▶ `echo $var1, $var2, ...;`

## ▶ print

- ▶ Prima samo jedan argument, uvijek vraća 1
- ▶ Također nije prava funkcija
- ▶ `print $var;`

## ▶ printf

- ▶ Kao i u C-u

## Ispis (2)

---

- ▶ Unutar HTML-a moguće je ispisivati podatke pomoću specijalnih tagova
  - ▶ `short_open_tag` u `php.ini` mora biti omogućen
- ▶ Primjer:
  - ▶ `<body>`
    - ...
    - Proizvod vrijedi `<?=$cijena?>` kn.
    - ...
    - `</body>`

# HTML u PHP-u koji je u HTML-u

---

- ▶ Moguće je ispresjecati HTML i PHP
  - ▶ Umjesto echo
  - ▶ Korisno za veće HTML blokove
- ▶ Primjer:
  - ▶ 

```
<?php
if ( $greska ) {
    ?>Došlo je do greške!<?php
}
?>
```

## Zadatak (2)

---



- ▶ Kopirajte stranica.php u smijesko.php
- ▶ U smijesko.php ubacite neku varijablu kojoj je sadržaj:
  - ▶ :) A < B > C :)
- ▶ Ispisati sadržaj te varijable tako da
  - ▶ :) se zamijeni sa slikom smiješka
    - ▶ <http://images.google.com>
    - ▶ str\_replace
  - ▶ <B> ne završi kao tag!
    - ▶ htmlspecialchars

# Uvjetno izvođenje

---

## ▶ Također kao u C-u...

- ▶ `if ($var == "vjestina") // string == string`
- ▶ `if ($var) // nije null, 0, false, prazan string ili prazno polje`
- ▶ `if (!$var) // suprotno od prethodne`
- ▶ `if (isset($var)) // da li je definirana $var`
- ▶ `if ($var == 1) // ako je varijabla = 1`
- ▶ `if ($var > 1) // ako je veća od 1`
- ▶ `if ($var && $var2) // logički operatori kao u C-u`

# Usporedbe različitih tipova

---

- ▶ Dvije vrste usporedbe:
  - ▶ Labava usporedba (`==` i `!=`)
    - ▶ Kao i u C-u, ali podaci se pretvaraju u isti tip
    - ▶ Ako su stringovi s obje strane numerički, pretvaraju se u `integer`!
    - ▶ Npr. `"10"=="1e1"` vraća `TRUE`!
  - ▶ Stroga usporedba (`===` i `!==`)
    - ▶ Provjera **tipa** i vrijednosti
- ▶ Više na:
  - ▶ <http://php.net/manual/en/language.operators.comparison.php>
  - ▶ <http://www.php.net/manual/en/types.comparisons.php>

# Grananje (switch)

---

- ▶ Također kao u C-u...

```
▶ switch ($inp) {  
    case 'a': ... break;  
    case 'b': ... break;  
    ...  
    default: ...  
}
```

- ▶ Koristi **labavu** usporedbu!

# Petlje

---

## ► Sintaksa kao i u C-u

```
for ($i = 0; $i < 10; ++$i) {  
    echo "$i\n";  
}
```

```
$i = 10;  
while ($i-->0) {  
    echo "$i\n";  
}
```

```
$i = 10;  
do {  
    echo "$i\n";  
    $i--;  
} while ($i > 0);
```

# break i continue

---

- ▶ Dodatno omogućuju skakanje iz više petlji

- ▶ Zamjena za goto

- ▶ Primjer:

- ▶ 

```
for ($i = 0; $i < strlen($str); ++$i) {  
    switch ($str[$i]) {  
        case "\n":  
            break 2; // izađe iz for  
        case ' ':  
            break; // izađe iz switch  
        default:  
            echo $str[$i];  
    }  
}
```

- ▶ Ispisuje sve znakove osim razmaka iz prvog retka iz `$str`

## Zadatak (3)

---



- ▶ Napisati PHP skriptu "tabmnoz.php" koja generira tablicu množenja 1..10.
- ▶ Koristiti:
  - ▶ `<table border="1">`

# Zadatak (4)

---



- ▶ Napisati PHP skriptu "ponavljanje.php" koja prebrojava ponavljanje znakova a, b i c ispred znaka z unutar neke varijable.
- ▶ Primjer varijable:
  - ▶ **Abecedanejdedozjerimaž**
  - ▶ 4