

# 5. PHP (priečinky, súbory, regulárne výrazy)

13.3.2006

# Obsah

- Ďalšie užitočné funkcie
- Porovnávanie
- Priečinky, súbory, obrázky
- Regulárne výrazy
- `phpinfo()`

# Ďalšie užitočné funkcie

názov funkcie	popis
rand([min, max])	Vráti náhodné číslo z intervalu 0 a <b>RAND_MAX</b> , alebo z intervalu min - max
str_replace(čo, čím, kde)	Nahradí všetky výskyty <b>čo</b> reťazcom <b>čím</b> v <b>kde</b>
strrpos(kde, čo [, zac])	Vráti prvú pozíciu <b>čo</b> v reťazci <b>kde</b> zprava, alebo FALSE
strrchr(kde, čo)	Vráti podreťazec od pozície <b>čo</b> v reťazci <b>kde</b> zprava do konca reťazca, alebo FALSE. Vyhľadáva len znaky.
strstr(kde, čo)	Vráti podreťazec od pozície <b>čo</b> v reťazci <b>kde</b> zľava do konca reťazca, alebo FALSE.

# Alternatívne zápisy príkazov

```
if (podmienka) {  
    príkazy1;  
} else {  
    príkazy2;  
}
```

```
if (podmienka) :  
    príkazy1;  
else :  
    príkazy2;  
endif;
```

```
while (podmienka) {  
    príkazy;  
}
```

```
while (podmienka) :  
    príkazy;  
endwhile;
```

```
for (vyraz1; vyraz2; vyraz3) {  
    príkazy;  
}
```

```
for (vyraz1; vyraz2; vyraz3) :  
    príkazy;  
endfor;
```

# Porovnávanie

- == alebo ===
- != alebo !==
- if (0 == false) – správne/nesprávne ?

==	Rovnosť hodnôt
===	Rovnosť hodnôt aj typu – identita
!=	Rôznosť hodnôt
!==	Rôznosť hodnôt a typu – identita

# Galéria obrázkov

- Na stránke sa zobrazia najprv len náhľady obrázkov. Pri kliknutí na niektorý obrázok sa zobrazí veľký obrázok.
- Zadáme vstupný priečinkok s obrázkami.
- Všetky obrázky/súbory majú **rovnaký prefix**.
- Názov malého obrázka bez prefixu je názov veľkého obrázka.

# Funkcia nahlady()

```
function nahlady($prefix, $nazov, $cesta,  
    $alt) {
```

```
    ...
```

```
}
```

```
// napr. volanie funkcie
```

```
nahlady("m", "foto", "obrazky/05-foto",  
    "univerzita v Rakúsku");
```

# Priečinky

- `getcwd()` – vráti aktuálny priečinok
- `chdir(cesta)` – zmení aktuálny priečinok
- `opendir(cesta)` – otvorí priečinok a vráti **`dir_handle`**
- `readdir(dir_handle)` – vráti názov ďalšieho súboru v otvorenom priečinku
- `closedir(dir_handle)` – zatvorí otvorený priečinok

# Súbory

- `file_exists(súbor)` – vráti TRUE, ak súbor existuje
- `is_file(súbor)` – vráti TRUE, ak vstupom je normálny súbor
- `is_dir(súbor)` – vráti TRUE, ak súbor je v skutočnosti priečinok

# Obrázky

- `getimagesize(súbor)` – vráti pole, v ktorom sú informácie o obrázku
  - 0 => šírka
  - 1 => výška
  - 2 => typ
  - 3 => reťazec (šírka + výška), ktorý sa použije pri elemente `<img />`
- `exif_read_data(súbor)` – zistí EXIF hlavičky zo súborov JPEG a TIFF
- `exif_imagetype` – zistí typ obrázka (rýchlejšie ako `getimagesize()` )

# exif\_imagetype() – konštanty

Hodnota	Konštanta
1	IMAGETYPE_GIF
2	IMAGETYPE_JPEG
3	IMAGETYPE_PNG
4	IMAGETYPE_SWF
5	IMAGETYPE_PSD
6	IMAGETYPE_BMP
7	IMAGETYPE_TIFF_II
8	IMAGETYPE_TIFF_MM
9	IMAGETYPE_JPC
10	IMAGETYPE_JP2
11	IMAGETYPE_JPX
12	IMAGETYPE_SWC

# Informácie o súboroch

názov funkcie	popis
filesize(súbor)	Vráti veľkosť súboru v bytoch
filetype(súbor)	Vráti typ súboru
filemtime(súbor)	Vráti čas poslednej úpravy súboru
clearstatcache()	Vymaže cache obsahujúcu stav súborov

# Námet na zlepšenie

- Ak je počet obrázkov v riadku 3, kde 1 z nich je inak otočený, nech je v strede.

# Regulárne výrazy

- Regulárny výraz je vzorka (sada znakov), ktorá popisuje hľadaný reťazec.
- Vo vzorke môžeme použiť bežné ale aj špeciálne znaky.
- Zadávame/popisujeme len znaky, ktoré sa môžu vo výslednom reťazci nachádzať.

- **^xyz** – reťazec, ktorý začína xyz.
- **xyz\$** – reťazec, ktorý končí xyz.
- Znak bodka (.) – reprezentuje jeden ľubovoľný znak (okrem nového riadka)
- **[]** – v hranatých zátvorkách uvádzame množinu znakov (**pre 1 znak**)

# Špeciálne znaky

<code>\t</code>	Znak tabulátora
<code>\n</code>	Znak konca riadka
<code>\.</code>	Znak bodky
<code>\-</code>	Znamienko mínus
<code>\\</code>	Znak spätného lomítka

# Príklady

[asd]	Zodpovedá jednému znaku z množiny a,s,d
[abcdABCD]	Zodpovedá jednému znaku z množiny a,b,c,d,A,B,C,D
[a-dA-D]	Zodpovedá jednému znaku z množiny a,b,c,d,A,B,C,D
[a-zA-Z\._]	Zodpovedá jednému znaku z množiny malých a veľkých písmen, znaku bodky a znaku _

# Ošetrenie viacerých výskytov znakov - použijeme {}

1. {2} znamená, že predchádzajúci znak alebo množina znakov sa musí vyskytovať práve 2 krát
2. {2,} znamená, že predchádzajúci znak alebo množina znakov sa musí vyskytovať aspoň 2 krát
3. {2,4} znamená, že predchádzajúci znak alebo množina znakov sa musí vyskytovať aspoň 2 krát, najviac však 4 krát.

Napr.  $^{[0-9]{5}}\$$  – kontrola PSČ

$^{[0-9]{3} [0-9]{2}}\$$  – PSČ s medzerou

$^{[a-zA-Z0-9\._-]{2,}@[a-zA-Z0-9\._-]{1,}}\$$  – e-mail

# Skratky ku $\{\}$

- $\{0,1\} \Rightarrow$  znak  $?$  – napr.  $[abc]?$
- $\{0,\} \Rightarrow$  znak  $*$  – napr.  $[abc]^*$
- $\{1,\} \Rightarrow$  znak  $+$  – napr.  $[abc]^+$

**$^[a-z]^+[0-9]\$$**

Zodpovedá viacerým (aspoň jednému) malým písmenám na začiatku a jednej cifre na konci reťazca

# Regulárne výrazy v PHP

- Boolovské funkcie **ereg()** alebo **eregi()**
- bool **ereg** (**vzorka**, **ret'azec** [, array regs])
- `ereg("^[0-9]{5}$", $psc)`

# phpinfo()

- Informácie o PHP a webovom serveri
- **PHP Variables**
  - **\_SERVER** (DOCUMENT\_ROOT, PHP\_SELF, REQUEST\_URI)
  - **\_POST**
  - **\_GET**
  - **\_SESSION**

# Testy na cvičeniach

- Začína sa cca 5 min. po oficiálnom začiatku cvičenia.
- Začiatok a koniec riešenia testu stanoví cvičiaci.
- Tí čo prídu neskôr, musia skončiť s ostatnými v časovom limite.
- Dôvod?: Napr. kvôli nekorektnému fungovaniu časovača v MOODLE

**Ďakujem za  
pozornosť 😊**

Priestor na vaše otázky