



# INTERNETOVÉ TECHNOLOGIE – SERVER SIDE P2

## 2009-02-24

### PHP – PERSONAL HOME PAGE:

#### Základy syntaxe:

- ✓ Sekvence příkazů oddělených středníkem
- ✓ PHP ignoruje oddělovače (tab, space, enter)
- ✓ Není „Case Sensitive“ – na velikosti písmen záleží pouze v názvech proměnných (Camel Case)

#### Komentáře v PHP:

- ✓ /\* Víceřádkový komentář – typ C \*/
- ✓ //Jednořádkový komentář  
// typ C++
- ✓ # Jednořádkový komentář  
# Typ Bourne Shell

#### Proměnné:

- ✓ Názvy proměnných začínají znakem dolaru, např. \$a
- ✓ Název proměnné může obsahovat libovolná písmena, číslice nebo podtržítka
- ✓ Název proměnné však nesmí začínat číslicí
- ✓ Proměnné jsou automaticky deklarovány v okamžiku jejich prvního použití

#### Datové typy:

- ✓ Primární datové typy:
  - Celá čísla – integer
  - Reálná čísla – double
  - Řetězce – string
- ✓ Složené datové typy:
  - Pole – array
  - Objekty – object

#### Typ pole:

- ✓ Jedna proměnná může obsahovat několik hodnot různých typů.
- ✓ Jednotlivé hodnoty jsou přístupné pomocí indexů, které se zapisují do hranatých závorek
- ✓ Příklady:  
\$a[0]="pondělí";                      \$a[]="pondělí";                      \$a=Array("pondělí","úterý");  
\$a[1]="úterý";                      \$a[]="úterý";

#### Asociativní pole:

- ✓ V asociativních polích je jako index pole použit textový řetězec
- ✓ Příklady:  
\$mail["Pavel"]="jerzi@pef.czu.cz";  
\$mail["Havlicek"]="havlicek@pef.czu.cz";
- ✓ Kromě uvozovek lze používat také apostrofy, co je v uvozovkách, je i vykonáváno (můžeme do uvozovek vložit i název proměnné a místo názvu se jako hodnota uloží hodnota té proměnné a ne její název), potřebujeme-li uložit i znaménko dolar, je vhodný apostrof

#### Vícerozměrná pole:

- ✓ Pro každý rozměr stačí přidat nové hranaté závorky s dalším indexem.
- ✓ Příklad: \$m[2][3]



### Objekt:

- ✓ Typ, který může obsahovat libovolný počet proměnných a funkcí.
- ✓ PHP podporuje pouze jednoduchou dědičnost a konstruktory

### Příklad:

### Operátory:

- ✓ () řízení priorit
- ✓ [] přístup k indexu pole
- ✓ ! logická negace
- ✓ ++ -- inkrementace, dekrementace
- ✓ @ potlačení chybových hlášení
- ✓ \* / % násobení, dělení, modulo (zbytek po dělení)
- ✓ + - . sčítání, odčítání, operátor pro spojování řetězců

### Řídící struktury:

- ✓ Příkazy pro větvení programu:
  - If (else) – buď a nebo
  - Switch (case) – přepínání více možností
- ✓ Příkazy cyklu:
  - While
  - Do/while
  - For

### If:

- ✓ Provedení příkazu pokud je splněna určitá podmínka.
- ✓ Příklad:  
//Pokud je a větší než b, skript vypíše text: a je větší než b  
If (\$a>\$b) ...

### If – else:

### Switch – case:

- ✓ Na základě hodnoty jednoho výrazu se provádí jedna větev skriptu

### While:

- ✓ Vytvoření smyčky, která opakovaně provádí nějaké operace, dokud platí určitá podmínka.

### Do – while:

- ✓ Pracuje podobně jako while, podmínka je však umístěna na konci cyklu
- ✓ Rozdíl: cyklus podmínkou na konci proběhne minimálně jednou

### Cyklus for:

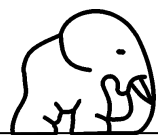
- ✓ For(inicializační\_výraz;podmíněný\_výraz;iterační\_výraz) tělo\_cyklu;
- ✓ Inicializační výraz

### Booleovské hodnoty proměnných:

- ✓ Každá proměnná má přiřazenu booleovskou hodnotu (true nebo false)
- ✓ Integer a real jsou false jen pokud je jejich hodnota rovna nule
- ✓ String má hodnotu false jen pokud je prázdný
- ✓ Pole má hodnotu false...

### Vkládání souborů:

- ✓ Vkládané soubory často obsahují předdefinované funkce, konstanty, standardizované záhlaví a zápatí atd.
- ✓ Příkaz require „soubor.xxx“; - pomocí require je každý soubor možné vložit v rámci jednoho skriptu pouze jednou



- ✓ Příkaz include „soubor.xxx“ – je možné ho použít pro opakované vkládání jednoho souboru (např. v rámci cyklu);

### Dynamické proměnné:

- ✓ Jméno proměnné lze chápat jako dynamickou proměnnou
- ✓ Příklad:  
\$pole = "IDproduktu";  
\$\$pole = "123A";  
// totéž jako: \$IDproduktu = 123A;  
echo \$pole." je ".\$pole;  
// vypíše: IDproduktu je 123A;

### Webové proměnné:

- ✓ PHP automaticky vytváří proměnné pro všechna data přijatá v http požadavku, to se týká metod GET, POST, cookies i proměnných prostředí.
- ✓ Příklad: po obdržení této URL adresy: <http://www.nekde.cz/dp.php?jmeno=Pavel&znamka=1>
- ✓ V souboru dp.php můžeme pracovat s proměnnými \$jmeno a \$znamka,

### Konstanty:

- ✓ Slouží k uchovávání hodnot, které nelze během chodu skriptu měnit.
- ✓ Pro definování konstanty se používá funkce define();

### Propojení s DB:

- ✓ PHP podporuje řadu databází přímo přes jejich API (MySQL, MsSQL, Oracle, PostgreSQL, Informix atd.)
- ✓ K řadě databází je také možno přistupovat...

### Propojení PHP s MySQL:

- ✓ Práci s MySQL musíme zahájit připojením k serveru pomocí funkce:
- ✓ Po připojení do DB je nutné vybrat databázi, se kterou budeme pracovat:
- ✓ Poté již můžeme klást dotazy nad tabulkami této databáze:

### Globální proměnné:

- ✓ V PHP existují předdefinované proměnné – jedno i vícerozměrná pole
- ✓ Vše obsahuje pole \$GLOBALS
- ✓ Dvě možnosti přístupu – první uvedená je deprecated od verze PHP 4:
  - Jménem parametru, např. \$HESLO – dnes se většinou nevyskytuje
  - Jménem parametru, např. \$\_POST["heslo"]
  - Jménem parametru, např. \$GLOBALS["POST"]["heslo"]

### Příklad:

```
<?php
echo "Ahoj světe!";
$a = 1;
$b = 2;
$c = $a + $b;
echo "c je $c"; //c je 3
echo 'c je $c'; //c je $c
?>
```

```
<form action="posli.php" method="post">
Text: <input name="text" />
E-mail: <input name="email" />
<input type="submit" value="Odeslat" />
</form>
```



```
<?php
$text = $_POST["text"];
$email = $_POST["email"];
mail($email, "Predmet", "$text", "From: lohr..");
?>
```