

HTML ÉS PHP

(Nagyon) rövid áttekintés

ADATBÁZISRENDSZEREK MŰKÖDÉSI SÉMÁJA



A gyakorlaton:



MySQL

HTML

- ▶ Az Internet legfontosabb szolgáltatása a WWW
- ▶ A weboldalak megjelenését és szerkezetét egy speciális jelölőnyelv, a HTML határozza meg
- ▶ A fenti információkat a szövegbe ágyazott HTML tag-ek segítségével adjuk meg
- ▶ A fájlok kiterjesztése .html

```
<!-- komment -->  
<html>  
  <body>  
    A dokumentum törzse  
  </body>  
</html>
```


STATIKUS VS. DINAMIKUS WEB

- ▶ Eddig arra láttunk példát, hogyan készítsünk HTML jelölésekkel formázott weblapokat
- ▶ Ezek publikálhatók az Interneten (egy szerveren), és az érdeklődők elolvashatják őket
- ▶ Egy (web)alkalmazásnál szükséges a felhasználói interakciók kezelése, a tartalom dinamikus előállítás (például: vásárló online kosarában lévő termékek listája változik a vásárlás folyamán)
- ▶ Ehhez önmagában a HTML nem elég
- ▶ Request-response (kérés-válasz) modell

PHP ALAPOK

- ▶ Objektorientált (vagy inkább többparadigmájú) programozási nyelv
- ▶ Lényegi működése: a kliens oldalról bejövő adatokat fogad, és azok ismeretében a szerver végrehajtja a programunkat, “legyárt” egy weblapot, ami a felhasználónak megjelenik
- ▶ .php kiterjesztéssel mentjük el a programjainkat
- ▶ ezen fájlokban tetszőleges mennyiségű HTML kódot helyezhetünk el, valamint speciális határolók (<?php és ?>) közt a programot (sőt ezek többször váltakozhatnak is)

ez itt a PHP-kód,
az echo parancs kiír egy szöveget

```
<!DOCTYPE html>  
<meta charset=utf-8>  
<title>PHP Test</title>  
<?php  
    echo 'Hello World';  
?>
```

PHP / ADATSZERKEZETEK, VEZÉRLÉS

- ▶ a változók típusaival nem nagyon kell törődnünk

```
$valtozo = 'Egy szoveg';  
$masik = 421;  
$tomb = array('Suzuki', 'Fiat', 'Volvo');  
$atomb = array('zero' => 'nulla', 'one' => 'egy',  
               'two' => 'kettő');  
echo 'Two is called ' . $atomb['two'] . ' in Hungarian';
```

asszociatív tömb:
kulcs-érték párok
megadásával

- ▶ Feltételes vezérlési szerkezet:

```
if (feltétel1) { ... } [elseif (feltétel2) {...}  
else { ...}]
```

sztring
konkatenáció

- ▶ Diszkrét ismétlés (“minden elemre hajtsd végre”):

```
foreach ($tomb as $auto) { echo $auto; }
```

PHP / FÜGGVÉNYEK, OSZTÁLYOK

► **Függvény:**

```
function osszeado($a, $b)
{
    return $a+$b;
}
```

► **Osztály:**

```
class MintaOsztaly
{
    public $var = 'kezdőérték';
    public function ertekKiir()
    {
        echo $this->var;
    }
}
```

A class kulcsszóval definiáljuk az osztályt.
A tagváltozóknak adunk láthatóságot, pl. itt a \$var változó publikus.
Ha egy tagváltozóra hivatkozunk metódusokban, kiírjuk az aktuális objektumpéldányt jelentő \$this-t a változó neve elé.

BEÉPÍTETT FÜGGVÉNYEK PHP-BAN

- ▶ A PHP számos beépített függvénycsomaggal telepíthető a szerverre
- ▶ Ezek számos programozói problémára megoldást szolgáltatnak
- ▶ Példák:
 - ▶ `str_replace($mit, $mire, $miben)` - Megkeresi és lecseréli `$mit` összes előfordulását `$mire` értékével
 - ▶ `$eredmeny = array_merge($tomb1, $tomb2 [, ...])` - Összefésül két v. több tömböt
- ▶ Egy komplexebbnek tűnő probléma egyéni megoldása helyett először érdemes átolvasni a dokumentációt