

# Objektum orientált PHP

készítette: Sztanka-Tóth Tamás,  
Sánta Zsolt

# Mi is az az objektum?

- Az objektum: Változók és függvények összessége
- Egységek: Class
  - változó
  - függvény
- Az osztályon belül 3 féle képen látható függvény vagy osztályt adhatunk meg:
  - public:
    - ez az alapértelmezett, az osztályon kívül is látható
  - private
    - a legvédettebb, csak az osztályon belül érhető el
  - protected
    - az osztályon belül, illetve az osztály leszármazottjából érhető el.

# A \_\_construct beépített függvény

- Attól függően fut le, hogy hány változóval hívjuk meg az osztályt

- Pl.:

```
class pelda1{
    private $a;
    function __construct($a){
        echo "egy változóval hívtad meg az osztályt";
    }
    function __construct(){
        echo "változó nélkül hívtad meg az osztályt";
    }
}
```

# Setter, getter függvények

- Hogyan tudunk, private változót az osztályunkon kívül is használni?
- Írunk egy public függvényt, amely segítségével ez lehetséges lesz.

- Pl.:

```
class pelda2{  
private $a;
```

```
public setA($x){  
$this->a=$x;  
}
```

```
public getA(){  
return $this->a;  
}
```

```
}
```

# Osztály származtatása

- A gyerek osztály, a szülő osztálynak minden változóját és függvényét örökli ( ha azok nem private jelölésűek, illetve nem örökli még a `__construct` függvényeket )
- Persze a gyerek osztálynak lehet megadni új függvényeket ill. változókat is.

- Ennek segítségével nem kell mindent többször megírni

- A kulcsszó az `extends`

- Pl.:

```
class pelda3{  
protected $a;  
  
...  
}
```

```
class pelda4 extends pelda3{  
private $b; // $a-t és $b-t is tartalmazza az osztály  
}
```

# \_\_constructor származtatása

- A \_\_constructor függvény alpból nem lesz lemásolva, de tudjuk származtatni a parent:: segítségével
- A következő féleképpen:

```
class a{
    __construct(){
        ...
    }
}
class b extends a{
    __construct(){
        parent::__construct();
    }
}
```

# Konstansok az osztályokban

- A php lehetővé teszi hogy konstansokat deklaráljunk
- Amikor értéket adunk nekik, nem kell a "\$" jel
- Pl.:

```
class pelda5{  
    const valami = 1;  
    function showconstant(){  
        echo self::valami;  
    }  
}
```
- Az osztály példányosítása nélkül is tudjuk használni az osztályon kívül. Pl.:

```
echo pelda5::valami;
```

források:

- Paksy Patrik mysql és php prezentáció
- [www.php.net](http://www.php.net)
- [www.w3schools.org](http://www.w3schools.org)

készítette: Sánta Zsolt, Sztanka-Tóth Tamás