



WEB разработка

**PHP- Обектно-ориентирано
програмиране
въведение**

Съдържание

- Какво е ООП?
- Класове и обекти (classes & objects)
- \$this
- Методи и атрибути/свойства (methods& properties)
- Конструктор & Деструктор
- Обхват (scope)
- get/set
- ООП?

Класове и обекти

Class

New

Класове и обекти

Класът е тип структура /шаблон/.

Има свойства/properties/ и методи/methods/.

```
class Substance {  
    var $color;  
    var $name;  
    ....  
    function melts() {  
    }  
    function evaporates(){  
    }  
    ....  
}
```

Класове и обекти - 2

Обект на клас /инстанция на клас, instance of a class/

```
$stone          = new Substance();
```

```
$air            = new Substance();
```

```
....
```

```
$water         = new Substance();
```

Класове и обекти - 3

И трите обекта - \$stone, \$air, \$water - са инстанции на един и същи клас - Substance - имат едни и същи свойства и методи.

Задачи - на лист хартия -

1. Опишете поне два класа с код - като добавите по три свойства и два метода, характерни за всеки клас; Създайте по два обекта от всеки клас.
2. Направете ООП дизайн за библиотека. Да има поне три класа със свойства и методи, характерни за тях. Създайте по два обекта от всеки клас.

\$this

\$this

\$this - сочи към текущия обект и показва, че работим в неговия контекст

```
class Substance {  
    var $volume;  
  
    function evaporates(){  
  
        $this->$volume-=10;  
  
    }  
}
```


__constructor()

__constructor()

Извиква се при създаване на нов обект на клас

Всеки клас може да има само 1 конструктор

__constructor() - 2

```
class Substance {
    var $volume;
    var $name;
function __construct($vol, $n){
    $this->volume = $vol;
    $this->type = $l_type
}
function evaporates(){
    $this->$volume-=10;
}
function print_info(){...}
}
```

```
$water1 = new Substance(10, 'water');
```

```
$water->print_info()
```

```
//
```

```
Name: water;
```

```
Volume: 10 l;
```

```
$water->evaporates();
```

```
$water->print_info()
```

```
//
```

```
Name: water;
```

```
Volume: 0 l;
```

__destruct()

__destruct()

Извиква се при унищожаване на всички референции на даден обект

Всеки клас може да има само 1 деструктор

__destruct() - 2

```
function __destruct(){  
//close sessions, mysql connection etc  
}
```

Задачи

Опишете с код -

1. Клас с поне две свойства, конструктор, методи /поне два/, които променят или в който участват свойствата.
2. Създайте поне два обекта от този клас.
3. Извикайте методите върху обектите и
4. Отпечатайте в браузъра, резултат от методите.

SCOPE /обхват/

scope

Всеки метод и свойство на клас имат обхват /scope/, определящ кой може да достъпва този метод или свойства

public - може да се достъпва от всякъде;

protected - ...

private - може да се достъпва само вътре в класа

Конструктора и деструктора трябва да са public.

scope - 2

```
class Liquid {  
    public function printObjName() {  
        echo $this->getName();  
    }  
  
    private function getName() {  
        return get_class($this);  
    }  
}
```

scope - 3

```
class Liquid {  
    var $volume;  
    private $basic_ingredient;  
  
    public function __construct($vol, $bin){  
        $this->volume = $vol;  
        $this->basic_ingredient = $bin;  
    }  
}
```

`__get()`

`__set()`

get&set

Магически функции - [детайли](#)

[__get\(\)](#) - използва се за прочитане на недостъпни данни/private properties/;

[__set\(\)](#) - присвояване на стойност

[property_exists](#) ([mixed](#) \$class , string \$property) - проверява дали обекта/класът притежава дадено СВОЙСТВО

Ресурси

ресурси

[Официална документация за PHP ООП](#)

[PHP ООП - видео уроци](#)

[PHP ООП - tutorial](#)

[PHP ООП - упражнения](#)