web разработка

ООП
Статични методи и свойства

Съдържание

- Статични методи и свойства
- Константи
- Getters, Setters

Статични свойства

Статични свойства

Свойства, принадлежащи на класа, общи за всички обекти от този клас.

Много често статични са свойствата, съхраняващи връзката към базата данни.

```
class redCar {

static public color = ‘red’;

}
```

Статични свойства

Извеждане на екран

```
echo RedCar::$color;
```

Обръщане към статичните свойства на класа в самия клас

```
public function get_static(){

        echo self::$color;

}
```

Статични свойства

Поведение на статичните свойства в класовете наследници

```
Class Toyota extends RedCar {

        static public $color = ‘blue’;

        public function get(){
                echo self::$color;
        }
}

$toyota = new Toyota();
$toyota = get_static();
$toyota = get();
```

```
Class Toyota extends RedCar {
        static public $color = ‘blue’;
        public function get(){
                echo static::$color;
}
}

$toyota = new Toyota();
$toyota = get_static();
$toyota = get();
```

Статични методи

Статични методи

Методи, принадлежащи на класа, общи за всички обекти от този клас.

Работят само със статични свойства

```
class String {

static public $str = ‘hello’;

static public fucntion st_method() {

        echo self::$str;

    }

}
```

Извикваме ги -

```
String::st_method();
```

Late Static Binding

Late Static Binding

```
class A {
    static public $name = "A";

    static public function getName () {
        return self::$name;
    }

    static public function getNameLateStaticBinding () {
        return static::$name;
    }
}
```

```
class B extends A {
    static public $name = "B";
}
```

Late Static Binding

```
// Output: A
echo A::getName();

// Output: A
echo B::getName();

// Output: B
echo B::getNameLateStaticBinding();
```

B::getName() извежда стойността на променливата $name от класа A, тъй като **self** сочи към класа, в който е дефиниран съответния метод.

За да изведем стойността на променливата за текущия клас, за който я извикваме, използваме **static**.

Константи

Константи

Константите са постоянни величини, които дефинираме в началото на скрипта.

Не ги променяме, а само ги извикваме.

Как ги дефинираме

- Процедурно програмиране

define(‘PAGE’, ‘index’);

- Прието е имената им да се изписват изцяло с главни букви

Константи

В ООП имаме възможност да работим с константа/и на класа

Как ги дефинираме

```
class Page {

const NUMBER = 1;

}
```

Отпечатайте константата NUMBER

NUMBER е константа на класа, не на отделния обект от този клас.

```
Page::NUMBER
```

Или в класа

```
public get_const() {

echo self::NUMBER

}
```

Константи

Константата е видима и в класовете наследници

```
Class Index extends Page {
        public function __construct(){
        echo self::PAGE;
    }
}
$index = new Index();
//вижте резултата в браузъра
```

Константи

приложение

- Постоянни стрингове, които трябва да извеждаме на екрана
- Настройки, включително и нужните за базата данни

HOST =

DATABASE =

PASSWORD =