Запись на курсы по HTML, CSS, JavaScript, PHP, фреймворкам и CMS, а также: помощь в поиске работы и заказов, стажировка на реальных проектах→

урок 23 из 107

Beрстка JavaScript PHP NodeJs Vue React Laravel WordPress AJAX Парсинг

Бесплатные курсы по React для новичков. Начало 4-го ноября→ Конкурс CSS картинок. Тема: Хэллоуин. Призовой фонд: 100\$. Подробности→

## Передача объектов по ссылке

Пусть у нас дан вот такой класс **User**:

```
1
    <?php
2
      class User
3
4
        public $name;
5
        public $age;
7
        public function __construct($name, $age)
8
9
          $this->name = $name;
10
          $this->age = $age;
11
12
13 | ?>
```

Пусть мы создаем объект этого класса:

```
1 <?php
2 $user = new User('Коля', 30);
3 ?>
```

Представьте теперь следующую ситуацию: вы хотите значение переменной **\$user** присвоить какой-нибудь другой переменной, например **\$test**.

Если речь шла не об объектах, а о *примитивах*, то есть о строках, числах, массивах и тп, то в переменную **\$test** попала бы копия значения переменной **\$user**.

Что значит копия: это значит, что изменения любой из переменной в дальнейшем не меняли бы значение другой переменной.

Посмотрим на примере:

С объектами все по-другому - при записи в другую переменную объекты не копируются, а *передаются по ссылке*: это значит, что обе эти переменные значением имеют один и тот же объект.

Это имеет важное следствие: если поменять какие-то свойства объекта с помощью одной переменной, во второй переменной появятся эти же изменения:



```
er:

8 | e cho $user->name; // выведет 'Вася'!

9 | ?>
```

Учтите, что речь идет именно об объектах. Если вы в какую-то переменную запишите данные из свойства объекта - они скопируются, а не передадутся по ссылке:

```
1
   <?php
2
     $user = new User('Коля', 30);
3
     $name = $user->name; // запишем в переменную $na
4
       те текст 'Коля'
5
     $ name = 'Вася'; // поменяли переменную $name, но
       $user->name не поменялось
6
     // Проверим - выведем свойство name из переменной $us
     e cho $user->name; // выведет 'Коля'
9 ?>
1
   <?php
     $user = new User('Коля', 30);
2
3
4
     $test = $user; // и $test, и $user ссылаются на
       один и тот же объект
5
     $ test = 123; // теперь $test не ссылается, а $
       user - ссылается
6 ?>
```

Объект существует в памяти компьютера до тех пор, пока на него ссылается хоть кто-нибудь. В примере выше, если что-нибудь записать и в переменную **\$user** - на наш объект больше не будет ссылаться ни одна переменная и этот объект удалится из памяти.

## Задача 23.1

Сделайте класс **Product** (продукт), в котором будут следующие свойства: **name** (название продукта), **price** (цена).

## Задача 23.2

Создайте объект класса **Product**, запишите его в переменную **\$product1**.

## Задача 23.3

Присвойте значение переменной **\$product1** в переменную **\$product2**. Проверьте то, что обе переменные ссылаются на один и тот же объект.



