Конструктор объекта

Давайте рассмотрим следующий код:

```php
<?php
  // Класс с публичными свойствами name и age:
  class User
  {
    public $name;
    public $age;
  }

  // Создаем объект класса:
  $user = new User;

  // Записываем данные:
  $user->name = 'Коля';
  $user->age = 25;

  // Прочитываем данные:
  echo $user->name; // выведет 'Коля'
  echo $user->age; // выведет 25
?>
```

В данном коде не очень удобно то, легко можно забыть записать данные в какое-нибудь свойство объекта, особенно если этих свойств много.

Было бы удобно этот код:

```php
<?php
  // Создаем объект класса:
  $user = new User;

  // Записываем данные:
  $user->name = 'Коля';
  $user->age = 25;
?>
```

Заменить на вот этот:

```php
<?php
  $user = new User('Коля', 25); // создадим объ
    ект, сразу заполнив его данными
?>
```

То есть сделать так, чтобы поля объекта заполнялись при его создании - в этом случае мы никак не сможем забыть задать значения этих полей.

Для решения проблемы нам поможет так называемый *метод-конструктор* с названием **__construct** (в начале два символа подчеркивания).

Суть в следующем - если в коде класса существует метод с таким названием - он будет вызываться в момент создания объекта, смотрите пример:

```php
<?php
  class User
  {
    public $name;
```

```
    public $age;

    // Конструктор объекта:
    public function __construct()
    {
      echo '!!!';
    }
  }

  $user = new User; // выведет '!!!'
?>
```

Конструктор в общем-то такой же метод, как и все остальные и может принимать параметры, смотрите на примере:

```
<?php
  class User
  {
    public $name;
    public $age;

    public function __construct($var1, $var2)
    {
      echo $var1 + $var2; // найдем сумму параметров
    }
  }

  $user = new User(1, 2); // выведет 3
?>
```

Итак, давайте переделаем наш код, применив конструктор:

```
<?php
  class User
  {
    public $name;
    public $age;

    // Конструктор объекта:
    public function __construct($name, $age)
    {
      $this->name = $name; // запишем данные в с
        войство name
      $this->age = $age; // запишем данные в с
        войство age
     }
  }

  $user = new User('Коля', 25); // создадим объ
    ект, сразу заполнив его данными

  echo $user->name; // выведет 'Коля'
  echo $user->age; // выведет 25
?>
```

Задача 7.1

Сделайте класс **Employee**, в котором будут следующие публичные свойства - **name** (имя), **age** (возраст), **salary** (зарплата). Сделайте так, чтобы эти свойства заполнялись в методе **___construct** при создании объекта.

Задача 7.2

Создайте объект класса **Employee** с именем 'Вася', возрастом 25, зарплатой 1000.

Задача 7.3

Создайте объект класса **Employee** с именем 'Петя', возрастом 30, зарплатой 2000.

Задача 7.4

Выведите на экран сумму зарплат Васи и Пети.