

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

урок 89 из 107

[Верстка](#) [JavaScript](#) [PHP](#) [NodeJs](#) [Vue](#) [React](#) [Laravel](#) [WordPress](#) [AJAX](#) [Парсинг](#)[Бесплатные курсы по React для новичков. Начало 4-го ноября→](#) [Конкурс CSS картинок. Тема: Хэллоун. Призовой фонд: 100\\$. Подробности→](#)

Псевдонимы классов для пространств имен в ООП на PHP

Пусть у нас есть два класса Data, принадлежащие разным пространствам имен. Пусть в каком-то классе нам нужны объекты обоих этих классов:

```
1  <?php
2      namespace Project;
3
4      class Test
5      {
6          public function __construct()
7          {
8              $data1 = new \Core\Users\Data; // создаем объект
9              $data2 = new \Core\Admin\Data; // создаем объект
10         }
11     }
12  ?>
```

Пусть мы решили упростить вызовы классов через команду **use**. В этом случае нас ждет проблема: оба класса имеют имя Data, а это значит, что у нас будет конфликт имен:

```
1  <?php
2      namespace Project;
3
4      // Будет конфликт имен:
5      use \Core\Users\Data; // подключаем первый класс
6      use \Core\Admin\Data; // подключаем второй класс
7
8      class Test
9      {
10         public function __construct()
11         {
12             $data1 = new Data;
13             $data2 = new Data;
14         }
15     }
16  ?>
```

Для решения этой проблемы существует специальная команда **as**, позволяющая задать подключаемому классу псевдоним - имя, под которым этот класс будет доступен в этом файле.

Давайте сделаем так, чтобы первый класс Data был доступен как UserData, а второй класс Data - как AdminData:

```
1  <?php
2      namespace Project;
3      use \Core\Users\Data as UserData;
4      use \Core\Admin\Data as AdminData;
5
6      class Test
7      {
8          public function __construct()
9          {
10             $data1 = new UserData;
11             $data2 = new AdminData;
12         }
13     }
```

14 | }
?>

Задача 89.1

Упростите следующий код с использованием **use**:

```
1  <?php
2  namespace Project;
3
4  class Test
5  {
6      public function __construct()
7      {
8          $pageController = new \Resource\Controller\Page;
9          $pageModel      = new \Resource\Model\Page;
10     }
11 }
12 ?>
```

Задача 89.2

Упростите следующий код с использованием **use**:

```
1  <?php
2  namespace Project\Data;
3
4  class Test
5  {
6      public function __construct()
7      {
8          $pageController = new \Project\Data\Controller\Page;
9          $pageModel      = new \Project\Data\Model\Page;
10     }
11 }
12 ?>
```

