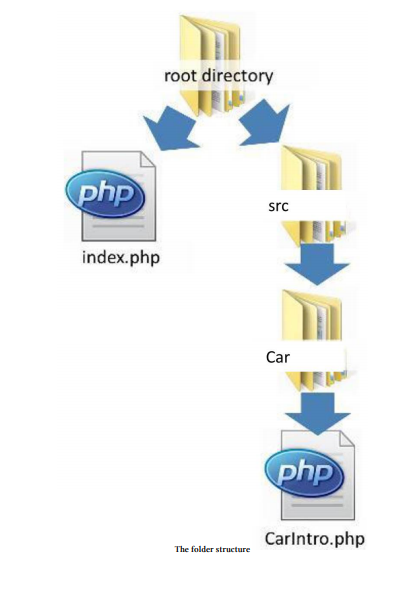
Namespaces

As our projects grow in complexity, we have no choice but to integrate code from different resources. It can be code that is written by other programmers in our team or is already present in code libraries that were developed elsewhere, and can do exactly what we need in a fraction of the time that it would take for developing our own library. In spite of its many benefits, we might encounter some problems when we integrate code from different resources. A huge problem, for example, is that of name collision, which occurs when different classes have the same name or have the same name for their methods. The problem is solved, as of PHP 5.3, by using namespaces.

The directory structure



Now, we can write the CarIntro class within the CarIntro.php file:

class CarIntro {

public function sayHello()

{

return "beep!";

}

}

The fact that we call our classes with the same name as our files. This naming convention would pay off later when we’d like to autoload our classes.

How to define a namespace?

We define the namespace at the very top of the page with the namespace keyword. For example, we will give the class the Acme\Car namespace:

<?php namespace Acme\Car;

class CarIntro {

public function sayHello()

{

return "beep!";

}

}

• It is customary to give to the src directory the Acme namespace.

• You can replace the Acme namespace with your brand name.

• It is a good practice to imitate the directory structure with the namespace. So, every class

within the directory: src/Car will get the Acme\Car namespace.

How to use a class that belongs to a namespace?

After we have namespaced the CarIntro class, we need to some how use it. For instance, we may want to use the class in the index.php file. So, in index.php, we need to do two things: First, we need to include the library and then import the namespace with the use keyword.

<?php

// 1. Require the namespaced file

require "src/Car/CarIntro.php";

// 2. Import the namespace

use Acme\Car\CarIntro();

// 3. Use the class from the context of the namespace

$carIntro = new Acme\Car\CarIntro();

echo $carIntro -> sayHello();

Result:

beep!

How to alias a class from a namespace with a friendly name?

<?php

// 1. Require the file

require "src/Car/CarIntro.php";

// 2. Import the namespace and give it a friendly name

use Acme\Car\CarIntro as Intro;

// 3. Call the class from the namespace with the friendly name

$carIntro = new Intro();

echo $carIntro -> sayHello();

Result:

beep!