Visibility of attributes and methods (public, protected and private).

In a class we can define our attributes and methods as public, protected or private (public, protected or private) depending on the visibility we want them to have in the rest of the code. In previous articles we only used public attributes and methods.

Let's see the differences between public, protected and private:

Public: we can access properties and methods from anywhere, from the current class, classes that inherit from the current class, and from other classes.

Protected: the attribute or method can be accessed from the class that defines it and from any other that inherits from this class.

Private: attributes or methods are only accessible from the class that defines them.

We can see an example of this in the Car class:

<? php

// We define the class

class Car {

/ \* Protected attributes accessible in classes that inherit from this \* /

protected $ model;

protected $ color;

protected $ speed;

/ \* Constructor, always has to be public \* /

public function \_\_construct ($ model, $ color, $ speed = 0) {

$ this-> model = $ model;

$ this-> color = $ color;

$ this-> speed = $ speed;

}

/ \* Private methods can only be called within this class \* /

private function getColor () {

return $ this-> color;

}

private function setColor ($ color) {

$ this-> color = $ color;

}

private function speed up () {

$ this-> speed ++;

}

private function stop () {

$ this-> speed--;

}

private function getSpeed ​​() {

return $ this-> speed;

}

public function showInfo () {

// We call other methods

$ info = "<h1> Car information: </h1>";

$ info. = "Model:". $ this-> model;

$ info. = "<br/> Color:". $ this-> getColor ();

$ info. = "<br/> Speed:". $ this-> getSpeed ​​();

return $ info;

}

}

?>