

UNIT 3 DELIVER EXERCISES OOP

(3 points exercise 2 and 3.5 points exercise 1 and 3)

1. Create a class "Person" with a property "Name" and its get and set methods.

Create a child class of Person called "Student" with these properties:

- License Number
- An Array with 3 positions containing 3 marks (1 per trimester)

It will also inherit from Person the 'name' property. The properties must be private, so you will need to provide set and get methods:

- Create a method with 2 parameters (mark and trimester). The method will save the mark in the required position.
- Create a method with 1 parameter (trimester). The method will return the mark pertaining to the required trimester.
- Create a method to return the average score.

Use the class "Student" to create a couple of objects, and do the following tasks:

Fill their properties. You can use a form to request its values.

List the students with their names, trimester marks and average score.

2. Create an Employee class and define its name and salary as attributes. An initialize method will also be defined that will receive the name and salary as data. Finally, create a method that prints the employee name and a message whether or not to pay taxes (if the salary exceeds 3000€, the employee will pay taxes). You should instantiate a couple of Employees

3. We want to create a class to implement and show a formatted HTML table.

The class, named Table, will have the following properties:

- An array with the titles of the table.
- A constant with the background color of the first row of the table:
`private const bgcolorheader="blue";`
- A static property with the background color of the table (all the other rows)
`private static $bgcolorall;`
- A two dimensional array which includes one data array for every row in the table.

The class Table needs some methods too:

- The property will be private, so we will need the setcolor method.
- A method to draw the header (first row) of the table.
- A method to draw the rest of the rows.
- A method called "drawTable" that shows the table on the screen using the two methods above (drawHeader and drawRows).

Once the class is ready, we will be able to do this in our main php file:

```
//Change the bgcolor of all our tables, create the setcolor method
Table::setcolor("white");

$titles=array("First Name","Last name","Email","Phone"); $data=array(
    array("Jose","González","josegonzalez@gmail.com","962324234"),
    array("Alba","Sanz","albasanz@gmail.com","666123123"),
    array("Ana","Blanes","anablanes@gmail.com","677656755"));
$table = new Table($titles, $data);
$table->drawTable();
```

And then the web will look something like this:

First Name	Last name	Email	Phone
Jose	González	josegonzalez@gmail.com	96234234
Alba	Sanz	albasanz@gmail.com	666123123
Ana	Blanes	anablanes@gmail.com	677656755