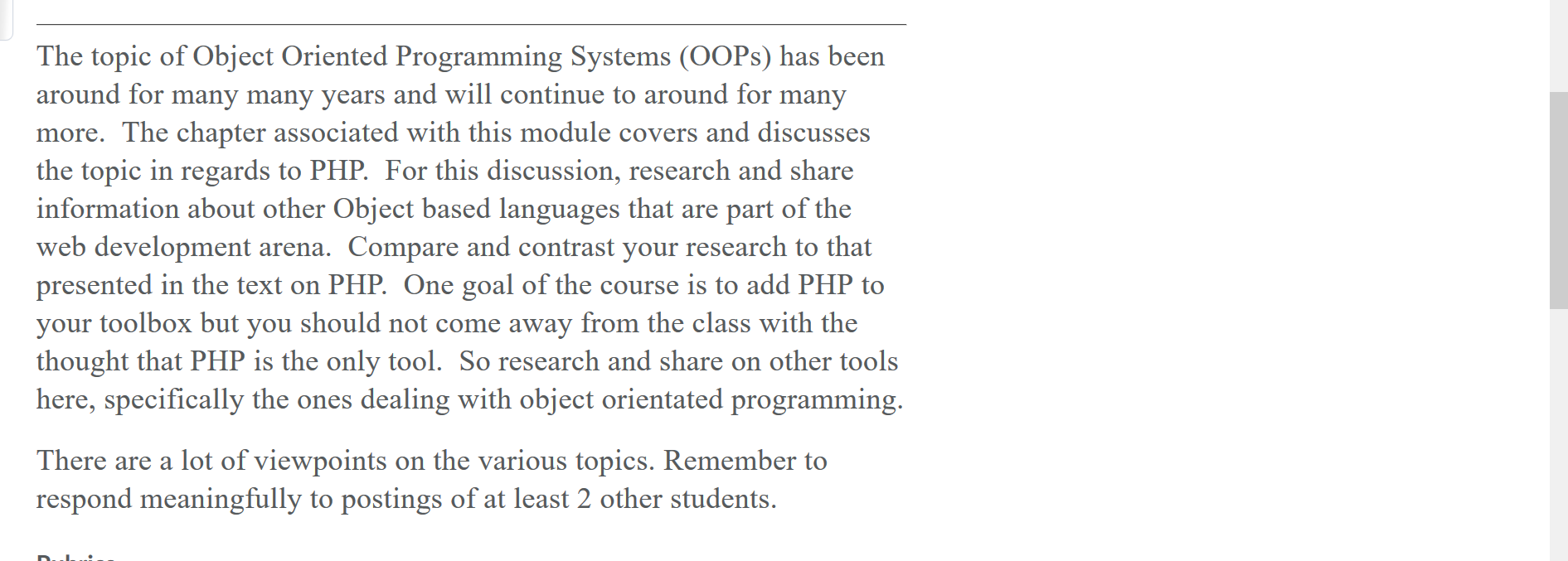
|  |  |
| --- | --- |
| Course Name | ITD 3243 – Server Side Programming |
| Instructor | Mark L. Pranger |
| Student Name | Sharyl Hammer |
| Due date | April 5, 2018 |
| Grade | <grade earned here> |
| Grading Comments | <instructor comments here> |



After reading the chapter which introduces how to create and use objects in PHP, the syntax for creating an object is very similar to C#. The concepts of using a class as the blue print of the object is the same in C#. The scope of the function which uses private, public and protected are the same. They both use a constructor which is a special method which is executed when a new object is created from a class.

When using a constructor in PHP the function has a special keyword that is \_construct. C# in comparison does not use a keyword function or \_construct to define a constructor. PHP uses a default constructor if one is not defined. C# uses a default constructor which is an empty constructor or a defined constructor. The code of how to define the default constructor which does not have parameters passed to it is below.

PHP Constructor – Default Constructor

public function\_construct()  
{

}

C# - public Product() – Empty Constructor

{

}

The comparison of the code with a constructor that uses a parameter. PHP uses the $this to refer to the current object and below in comparison is C# which uses this keyword to refer to the current object. The syntax is different but how it is being used is the same. The PHP constructor uses an object access operator which is the -> symbol which accesses the properties of the object. The C# uses the . which accesses the properties of the object. The properties in PHP are the $id and $name. The properties in C# are the code, description and price. These property values are being initialized to instance variables.

The constructor with parameters in PHP

Public function \_construct($id, $name) {

$this -> id = $id;

$this -> name = $name;

}

The constructor with parameters in C#

Public Product (string code, string description, decimal price )

{

this. Code = code;

this.Description = description;

this. Price = price;

}