Northumbria University NEWCASTLE

KF6012 - Web Application Integration

2020

Object Oriented PHP

Lecture Overview

- Object Oriented PHP

Object Oriented PHP

- We should begin by thinking about classes and objects/instances

Class

Instance(s)

Object Oriented PHP

- We should begin by thinking about classes and objects/instances

Class

```
class Student {

}
```

Instance(s)

```
$stu1 = new Student();
$stu2 = new Student();
```

Object Oriented PHP

- We can give our class properties, such as an id.

Class

```
class Student {
        private $id;
}
```

Instance(s)

```
$stu1 = new Student();
$stu2 = new Student();
```

Object Oriented PHP

- We can give our class methods (still called functions!)

Class

```
class Student {
        private $id;

        public function get_id() {
                return $this->id;
        }
}
```

Instance(s)

```
$stu1 = new Student();
$stu2 = new Student();

echo “<p>”. $stu1->get_id() .”</p>”;
echo “<p>”. $stu2->get_id() .”</p>”;
```

Object Oriented PHP

- Common methods in classes are “getters” and “setters”. Below is an example of a getter – get_id()

Class

```
class Student {
        private $id;
        public function get_id() {
                return $this->id;
        }
}
```

Instance(s)

```
$stu1 = new Student();
$stu2 = new Student();

echo “<p>”. $stu1->get_id() .”</p>”;
echo “<p>”. $stu2->get_id() .”</p>”;
```

- Note we’ve not actually set $id to anything yet!

Object Oriented PHP

- Stay organized! Put your classes in separate files in a subfolder, and use index.php to instantiate objects

Class	**Instance(s)**
. / classes / student.class.php	. / index.php

Thanks

- john.rooksby@northumbria.ac.uk