OOP in PHP

#6 Session

IEEE-AZHAR
STUDENT BRANCH

CONTENTS OF THIS SESSION

Here's what you'll learn at this Session:

- What is OOP
- Classes and objects
- Access Modifiers: public vs. private
- $this keyword
- Construct and Destruct

IEEE-AZHAR
STUDENT BRANCH

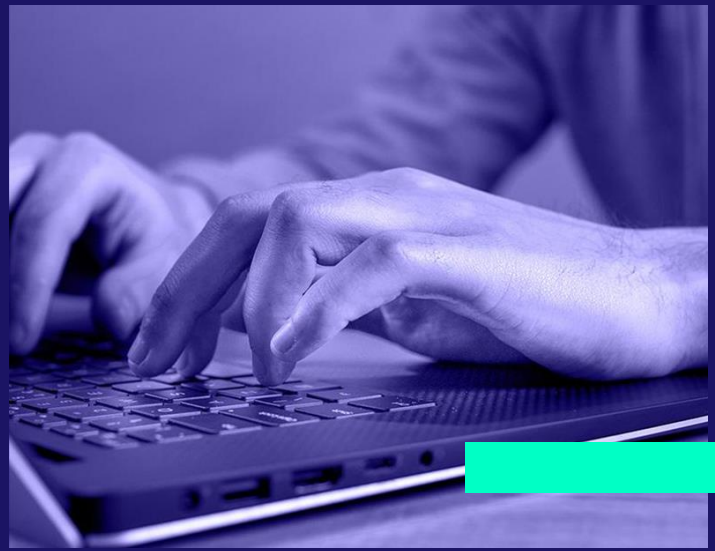
TABLE OF CONTENTS

IEEE-AZHAR
STUDENT BRANCH

What is OOP?

OOP focuses on the objects that are required to be manipulated instead of logic.

- It makes development and maintenance easier
- It provides data hiding
- It provides ability to simulate real-world
- less memory and organized
- reusable



IEEE-AZHAR
STUDENT BRANCH

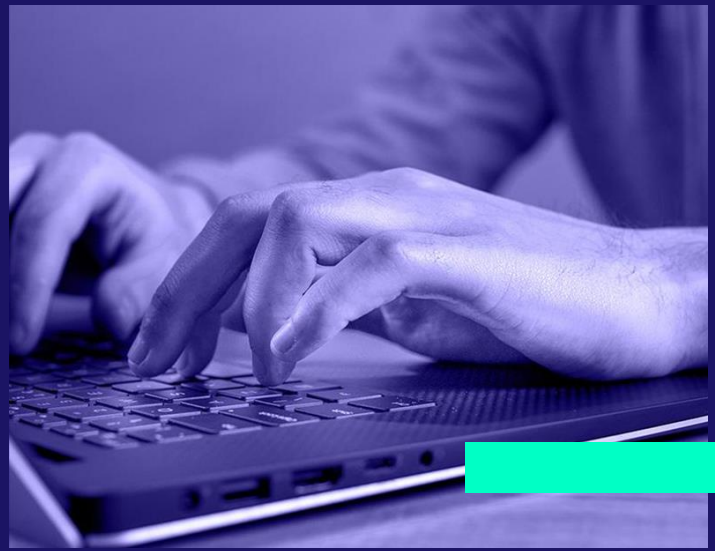
OOP language follow 4 principles:

1-ENCAPSULATION: We can hide direct access to data by using private key

2-ABSTRACTION: It is a process of hiding implementation details and showing only functionality to the user.

3-INHERITANCE: It is used to define the relationship between two classes.

4-: POLYMORPHISM: It is an ability of object to behave in multiple form.

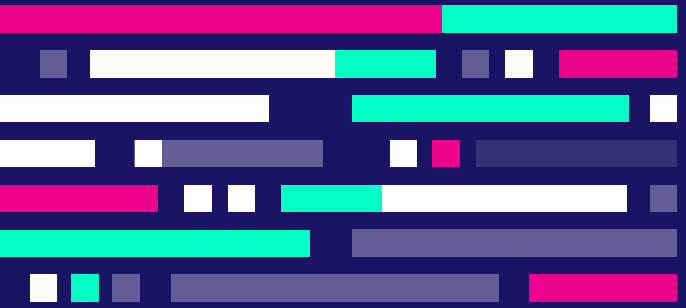


IEEE-AZHAR
STUDENT BRANCH

01

Classes and Objects_

IEEE-AZHAR
STUDENT BRANCH

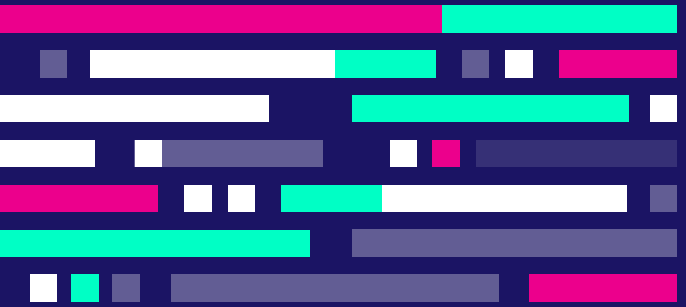


Objects?

If you look at the world around you, you'll find many examples of tangible objects: lamps, phones, computers, and cars.

IEEE-AZHAR
STUDENT BRANCH

PHP Access Modifiers



public - the property or method can be accessed from everywhere. This is default

protected - the property or method can be accessed within the class and by classes derived from that class

private - the property or method can ONLY be accessed within the class

IEEE-AZHAR
STUDENT BRANCH

In PHP, a Class starts with the Class sign, followed by the name of the Class

IEEE-AZHAR
STUDENT BRANCH

#Class Example

```php
<?php

class ClassName
{

}
?>
```

IEEE-AZHAR
STUDENT BRANCH

#Object Example

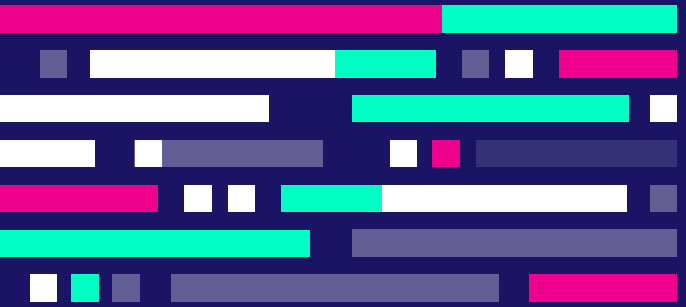
```php
<?php

$object = new ClassName();

?>
```

#Add properties to a class

```php
<?php

class BankAccount
{

    public $accountNumber;
    public $balance;

}
?>
```

IEEE-AZHAR
STUDENT BRANCH

Classes?



In the real world, you can find many same kinds of objects. For example, a bank has many bank accounts. All of them have account numbers and balances.

These Objects are created from the same blueprint. In object-oriented terms, we say that an individual bank account is an instance of a Bank Account class.

IEEE-AZHAR
STUDENT BRANCH

#Access a public property

```php
<?php

  $object->property;

?>
```

IEEE-AZHAR
STUDENT BRANCH

#set a public property

```php
<?php

  $object->property = 500;

?>
```

IEEE-AZHAR
STUDENT BRANCH

#Add method to a class

```php
<?php

class ClassName
{
   public function methodName(params)
   {
     // implementation
    }
?>
```

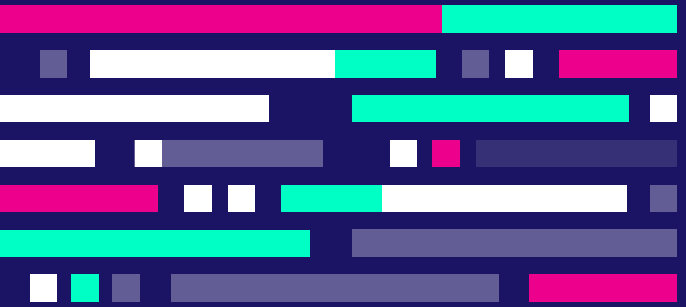
IEEE-AZHAR
STUDENT BRANCH

#Access to method in object

```php
<?php

  $object->method(arguments);

?>
```

IEEE-AZHAR
STUDENT BRANCH

$this keyword



The $this Variable reference to the Current object of the class

IEEE-AZHAR
STUDENT BRANCH

#$this keywords example

```php
<?php
class User
{
    public $name = “ahmed”;

    public function getName()
    {
        return $this->name;
    }
?>
```

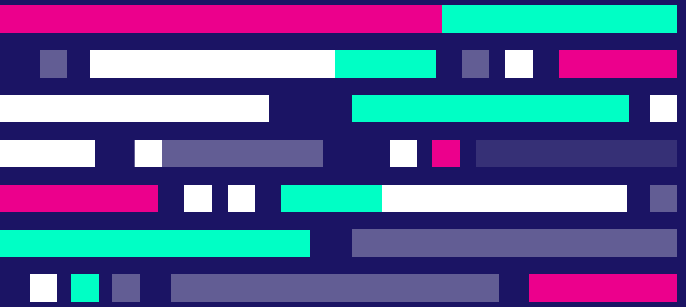
IEEE-AZHAR
STUDENT BRANCH

02

Constructor and Destructor

IEEE-AZHAR
STUDENT BRANCH

Constructor



When you create an instance of the class, PHP automatically calls the constructor method.

IEEE-AZHAR
STUDENT BRANCH

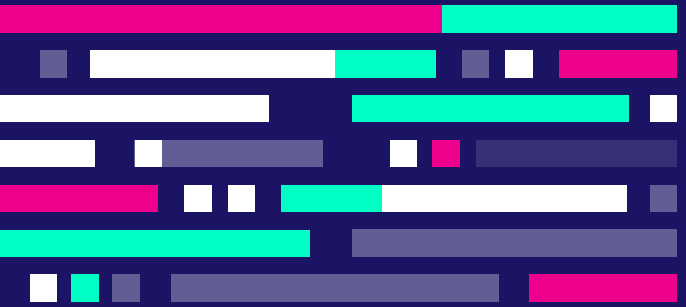
#constructer Syntax

```php
<?php
class User
{
    public function __construct($params)
    {
        // implementation
    }
?>
```

When you create an instance of the class,
PHP automatically calls the constructor method

IEEE-AZHAR
STUDENT BRANCH

Destructor



The destructor is automatically invoked before an object is deleted. It happens when the object has no reference or when the script ends.

IEEE-AZHAR
STUDENT BRANCH

#destructer Syntax

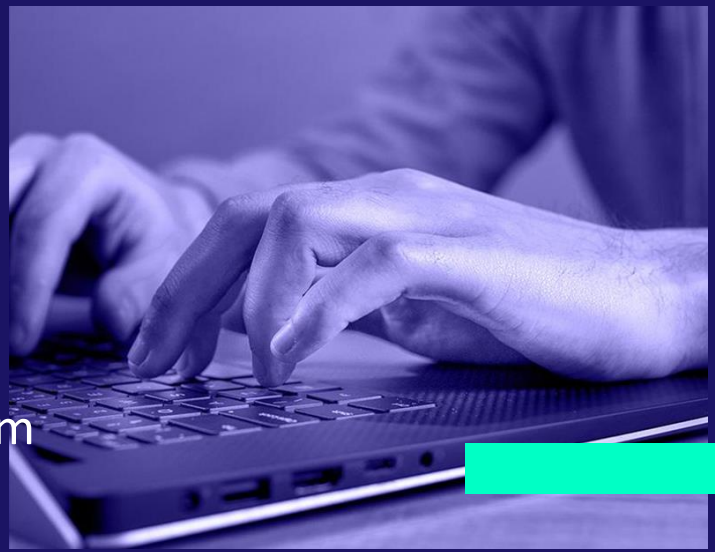
```php
<?php
class User
{
    public function __destruct()
    {
        // implementation
    }
?>
```

PHP automatically invokes the destructor when the object is deleted or the script is terminated.

IEEE-AZHAR
STUDENT BRANCH

Objects and Classes Summary

- Objects have properties and methods.
- A class is a blueprint for creating objects.
- Properties represent the object's state, and methods represent the object's behavior. Properties and methods have visibility.
- Use the new keyword to create an object from a class.
- The $this variable references the current object of the class.
- We can use Constructor and Destructor in Class



IEEE-AZHAR
STUDENT BRANCH

RESOURCES

- PHP Manual: php.net
- Tutorials Point: tutorialspoint.com/php
- PHP Tutorial: phptutorial.net
- W3 School: w3schools.com/php

IEEE-AZHAR
STUDENT BRANCH

Thanks for your attention ☺

IEEE-AZHAR
STUDENT BRANCH