# **Name : Aditi Dhepe**

# **Roll No : 2213774**

# Experiment No:4

Develop a web application in PHP using various concepts of object-oriented programming

like Class, Object, Inheritance, Function, Overloading, Constructor, and Destructor

## Theory:

1. **Class Definition (Car and SportsCar):**
   * The **Car** class has three public properties: **$brand**, **$color**, and **$model**.
   * It has a constructor **\_\_construct** that initializes the properties and displays a message when an object is created.
   * A method **displayInfo** is defined to display information about the car.
   * A destructor **\_\_destruct** is implemented to show a message when an object is destroyed.
2. **Inheritance:**
   * The **SportsCar** class extends the **Car** class, inheriting its properties and methods.
   * It introduces an additional property **$speed**.
3. **Constructor Overloading:**
   * The **SportsCar** class demonstrates constructor overloading by calling the parent class constructor using **parent::\_\_construct($brand, $color, $model)** and then setting its own property **$speed**.
4. **Method Overriding:**
   * The **SportsCar** class overrides the **displayInfo** method from the parent class to include information about the speed.

## Source code:

<!-- Develop a web application in PHP using various concepts of object-oriented programming

like Class, Object, Inheritance, Function, Overloading, Constructor, and Destructor

Code: -->

<?php

// Define a basic class 'Car'

class Car

{

    public $brand;

    public $color;

    public $model;

    // Constructor

    public function \_\_construct($brand, $color, $model)

    {

        $this->brand = $brand;

        $this->color = $color;

        $this->model = $model;

        echo "A new $this->color $this->brand $this->model created.<br>";

    }

    // Method to display car information

    public function displayInfo()

    {

        echo "This is a $this->color $this->brand $this->model.<br>";

    }

    // Destructor

    public function \_\_destruct()

    {

        echo "The $this->color $this->brand $this->model is being destroyed.<br>";

    }

}

// Inheriting class 'Car' into 'SportsCar'

class SportsCar extends Car

{

    public $speed;

    // Constructor Overloading within the inherited class

    public function \_\_construct($brand, $color, $model, $speed)

    {

        parent::\_\_construct($brand, $color, $model);

        $this->speed = $speed;

    }

    // Method Overriding

    public function displayInfo()

    {

        echo "This is a $this->color $this->brand $this->model and can speed up to $this->speed mph.<br>";

    }

}

// Creating objects from the classes

$car1 = new Car("Toyota", "red", "Camry");

$car1->displayInfo();

$sportsCar1 = new SportsCar("Ferrari", "red", "458 Italia", 200);

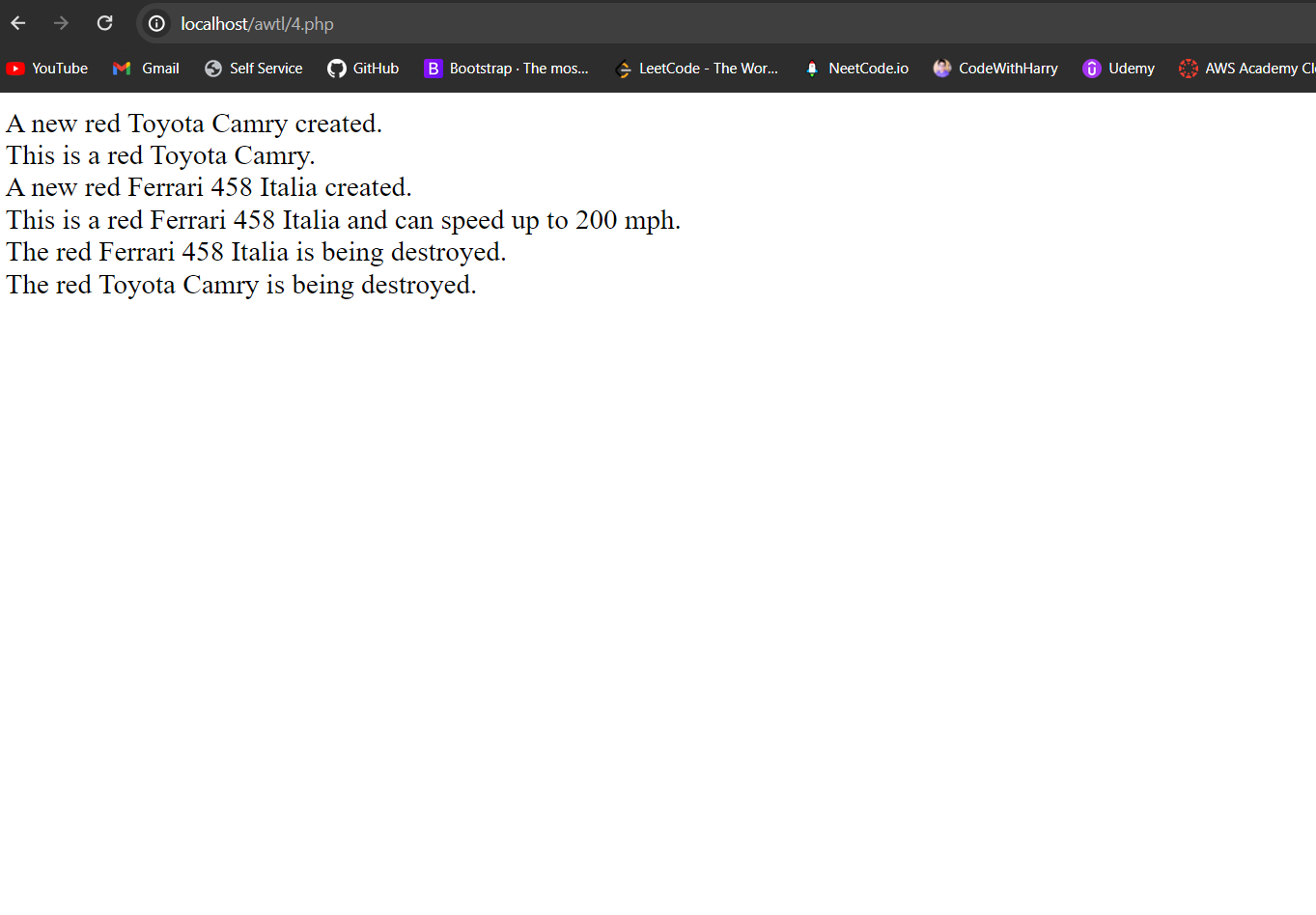
$sportsCar1->displayInfo();

?>

## Conclusion:

## The code successfully illustrates fundamental OOP concepts in PHP. It creates instances of the Car and SportsCar classes, showcasing the use of constructors, methods, inheritance, and method overriding.

## Output (Screenshots):



[Github : https://github.com/howlcat25/AWTL-SEM-6/blob/main/Source%20codes/assign\_4.php](Github%20:%20https://github.com/howlcat25/AWTL-SEM-6/blob/main/Source%20codes/assign_4.php)