

Practical-8
OOP with PHP -2

Practical – 8 OOP with PHP-2

1. Create a class Course having property coursename, no_of_year with a constructor, and display() method for the display course information. Create a class named Student which inherits a Course having stud_id, stud_name, array of marks(3 subject). Class contains the constructor and is calling the parent constructor. Class contains method caltotal() which returns the total of 3 subject marks. It contains display method which shows all the information about the students: Name,id,marks,total,course name,no_of_year.

Code:

8_1.php

```
<?php
class Course {
    public $coursename;
    public $no_of_year;

    public function __construct($coursename, $no_of_year) {
        $this->coursename = $coursename;
        $this->no_of_year = $no_of_year;
    }

    public function display() {
        echo "Course Name: " . $this->coursename . "<br>";
        echo "No. of Years: " . $this->no_of_year . "<br>";
    }
}

class Student extends Course {
    public $stud_id;
    public $stud_name;
    public $marks;

    public function __construct($coursename, $no_of_year,
        $stud_id, $stud_name, $marks) {
        parent::__construct($coursename, $no_of_year);
        $this->stud_id = $stud_id;
        $this->stud_name = $stud_name;
        $this->marks = $marks;
    }

    public function caltotal() {
        return array_sum($this->marks);
    }
}
```

Practical – 8 OOP with PHP-2

```
public function display() {
    parent::display();
    echo "Student ID: " . $this->stud_id . "<br>";
    echo "Student Name: " . $this->stud_name . "<br>";
    echo "Marks: " . implode(", ", $this->marks) .
    "<br>";
    echo "Total Marks: " . $this->caltotal() . "<br>";
}
}

$student = new Student("MCA",2, "22MAPOG010", "Kaushal
    Tarpara", [80, 85, 90]);

$student->display();

?>

<?php
    echo "<hr>Today is " . date("d/m/Y") . "<br>";
    echo "The time is " . date("h:i:sa");
?>
```

Output:



Course Name: MCA
No. of Years: 2
Student ID: 22MAPOG010
Student Name: Kaushal Tarpara
Marks: 80, 85, 90
Total Marks: 255

Today is 17/09/2023
The time is 07:52:52am

Practical – 8 OOP with PHP-2

2. Create a class named shape having a method area() which calculates the area of the shape and returns the area.

- a. Create a class circle which inherits Shape and overrides the area() method and returns the area of the circle.**
- b. Create a class square which inherits the Shape class and have an area() method to return the area of the square.**
- c. Create a class Rectangle which inherits the Shape class and have an area() method which returns the area of the Rectangle.**
- d. Create the object of class shape ,circle,square and rectangle class and display area of each. [override the area() method]**

Code:

8_2.php

```
<?php
class Shape {
    public function area() {
        // overridden in the child classes
    }
}

class Circle extends Shape {
    private $radius;

    public function __construct($radius) {
        $this->radius = $radius;
    }

    public function area() {
        return pi() * pow($this->radius, 2);
    }
}
```

```
    }  
}  
  
class Square extends Shape {  
    private $side;  
  
    public function __construct($side) {  
        $this->side = $side;  
    }  
  
    public function area() {  
        return pow($this->side, 2);  
    }  
}  
  
class Rectangle extends Shape {  
    private $length;  
    private $width;  
  
    public function __construct($length, $width) {  
        $this->length = $length;  
        $this->width = $width;  
    }  
}
```

Practical – 8 OOP with PHP-2

```
        public function area() {
            return $this->length * $this->width;
        }
    }

    $shape = new Shape();
    $circle = new Circle(5);
    $square = new Square(4);
    $rectangle = new Rectangle(3, 6);

    echo "Area of Shape: " . $shape->area() . "<br>";
    echo "Area of Circle: " . $circle->area() . "<br>";
    echo "Area of Square: " . $square->area() . "<br>";
    echo "Area of Rectangle: " . $rectangle->area() . "<br>";

?>

<?php
    echo "<hr>Today is " . date("d/m/Y") . "<br>";
    echo "The time is " . date("h:i:sa");

?>
```

Output:

Practical – 8 OOP with PHP-2



Area of Shape:

Area of Circle: 78.539816339745

Area of Square: 16

Area of Rectangle: 18

Today is 17/09/2023

The time is 07:53:32am

Practical – 8 OOP with PHP-2

3. Create an abstract class Employee having a constructor for setting name, year of joining, date of birth and department of employee. Create an appropriate method to display the details of the Employee in well designed HTML format. Create an abstract method calculate_salary() in Employee class.
 - a. Create a class Manager inherits the Employee class. It should include properties like basic salary, DA ,tax amount, HRA etc. property. Create the constructor for setting the values. Implement the calculate_salary() (basic+DA+HRA – tax amount).
 - b. Create a class worker which inherits the Employee class. Create a constructor which sets the property wages per hour,worked hour. Implement calsal() method (wages per hour*worked hour) .
 - c. Create appropriate methods in each class to display the well formatted details in HTML.
 - d. Write a php program which create multiple object of Manager and Worker class, and display the name, designation and salary of each

Code:

8_3.php

```
<?php

abstract class Employee {

    protected $name;

    protected $yearOfJoining;

    protected $dateOfBirth;

    protected $department;


    public function __construct($name, $yearOfJoining,
    $dateOfBirth, $department) {

        $this->name = $name;

        $this->yearOfJoining = $yearOfJoining;

        $this->dateOfBirth = $dateOfBirth;

        $this->department = $department;

    }

}
```


Practical – 8 OOP with PHP-2

```
public abstract function calculate_salary();

public function display() {

    echo "<h2>Employee Details</h2>";
    echo "<p>Name: " . $this->name . "</p>";
    echo "<p>Year of Joining: " . $this->yearOfJoining .
"</p>";
    echo "<p>Date of Birth: " . $this->dateOfBirth .
"</p>";
    echo "<p>Department: " . $this->department . "</p>";

}

}

class Manager extends Employee {

    private $basicSalary;

    private $DA;

    private $taxAmount;

    private $HRA;

    public function __construct($name, $yearOfJoining,
$dateOfBirth, $department, $basicSalary, $DA, $taxAmount,
$HRA) {

        parent::__construct($name, $yearOfJoining,
$dateOfBirth, $department);
```

Practical – 8 OOP with PHP-2

```
$this->basicSalary = $basicSalary;

$this->DA = $DA;

$this->taxAmount = $taxAmount;

$this->HRA = $HRA;
}

public function calculate_salary() {
    return $this->basicSalary + $this->DA + $this->HRA -
    $this->taxAmount;
}

public function display() {
    parent::display();
    echo "<p>Designation: Manager</p>";
    echo "<p>Salary: " . $this->calculate_salary() .
    "</p>";
}
}

class Worker extends Employee {
    private $wagesPerHour;
    private $workedHour;

    public function __construct($name, $yearOfJoining,
    $dateOfBirth, $department, $wagesPerHour, $workedHour) {
```

Practical – 8 OOP with PHP-2

```
        parent::__construct($name, $yearOfJoining,
$dateOfBirth, $department);

        $this->wagesPerHour = $wagesPerHour;

        $this->workedHour = $workedHour;

    }

    public function calculate_salary() {

        return $this->wagesPerHour * $this->workedHour;

    }

    public function display() {

        parent::display();

        echo "<p>Designation: Worker</p>";

        echo "<p>Salary: " . $this->calculate_salary() .
"</p>";

    }

}

$manager1 = new Manager("Kauhsal ", 2010, "2002-01-01",
"Devlpoment", 5000, 1000, 500, 2000);

$manager2 = new Manager("Anuj", 2012, "1999-05-10",
"Marketing", 6000, 1200, 600, 2500);

$worker1 = new Worker("Aman", 2015, "1998-03-15",
"Production", 10, 40);

$worker2 = new Worker("Kripal", 2018, "2002-07-20",
"Operations", 12, 35);
```

Practical – 8 OOP with PHP-2

```
$manager1->display();
$manager2->display();
$worker1->display();
$worker2->display();
?>

<?php
    echo "<hr>Today is " . date("d/m/Y") . "<br>";
    echo "The time is " . date("h:i:sa");
?>
```

Output:

Employee Details

Name: Kauhsal

Year of Joining: 2010

Date of Birth: 2002-01-01

Department: Devlpoment

Designation: Manager

Salary: 7500

Employee Details

Name: Anuj

Year of Joining: 2012

Date of Birth: 1999-05-10

Department: Marketing

Designation: Manager

Salary: 9100

Employee Details

Name: Aman

Year of Joining: 2015

Date of Birth: 1998-03-15

Department: Production

Designation: Worker

Salary: 400

Employee Details

Name: Kripal

Year of Joining: 2018

Date of Birth: 2002-07-20

Department: Operations

Designation: Worker

Salary: 420

Today is 17/09/2023

The time is 08:24:14am

Practical – 8 OOP with PHP-2

4 . Create a class Item with Property : Item name,Item no Method : display -> display item name and ino

a. Create a class Category which inherits Item

Property : category name[e.g cloth/electronics/kids toys.. etc] , subcategory[eg mobile, laptop, jeans,t shirt] , price

Method: displayItem() to display category, subcategory and price and getPrice() to return the price of item

b. Create a class purchase which inherits Item

Property : purchase id, total amount , quantity

Method :

- calculate_order_amount() : calculate and return the total amount (qty*price)
- display_purchase() : display the all details of Item and category.

c. Write a php script which creates the necessary constructor and creates an object of purchase class. Calculate the total order amount. It should also display the details purchas

Code:

8_4.php

```
<?php

class Item {
    protected $itemName;
    protected $itemNo;

    public function __construct($itemName, $itemNo) {
        $this->itemName = $itemName;
        $this->itemNo = $itemNo;
    }

    public function display() {
        echo "Item Name: " . $this->itemName . "\n";
        echo "Item No: " . $this->itemNo . "\n";
    }
}

class Category extends Item {
    protected $categoryName;
    protected $subcategory;
    protected $price;
```

Practical – 8 OOP with PHP-2

```
        public function __construct($itemName, $itemNo,
$categoryName, $subcategory, $price) {
            parent::__construct($itemName, $itemNo);
            $this->categoryName = $categoryName;
            $this->subcategory = $subcategory;
            $this->price = $price;
        }

        public function displayItem() {
            parent::display();
            echo "Category: " . $this->categoryName . "\n";
            echo "Subcategory: " . $this->subcategory . "\n";
            echo "Price: $" . $this->price . "\n";
        }

        public function getPrice() {
            return $this->price;
        }
    }

    class Purchase extends Item {
        protected $purchaseId;
        protected $totalAmount;
        protected $quantity;

        public function __construct($itemName, $itemNo,
$purchaseId, $quantity) {
            parent::__construct($itemName, $itemNo);
            $this->purchaseId = $purchaseId;
            $this->quantity = $quantity;
            $this->totalAmount = $this->calculateOrderAmount();
        }

        public function calculateOrderAmount() {
            // Assuming that the price is set in the Category
class
            $category = new Category($this->itemName, $this-
>itemNo, '', '', 0);
            $price = $category->getPrice();
            return $price * $this->quantity;
        }

        public function displayPurchase() {
```

Practical – 8 OOP with PHP-2

```
        parent::display();
        echo "Purchase ID: " . $this->purchaseId . "\n";
        echo "Quantity: " . $this->quantity . "\n";
        echo "Total Amount: $" . $this->totalAmount . "\n";
    }
}

$purchase = new Purchase("Laptop", 12345, 1, 2);

$purchase->displayPurchase();
?>
<?php
    echo "<hr>Today is " . date("d/m/Y") . "<br>";
    echo "The time is " . date("h:i:sa");
?>
```

Output:



Item Name: Laptop Item No: 12345
Purchase ID: 1 Quantity: 2 Total Amount: \$0

Today is 17/09/2023
The time is 08:29:09am
