Name:Vishal Singh

Class:Sybca(sci)

Roll nos:12939

Assignment :4

<html>

<body>

<center>

<h2> Calculate Area and Volume of Cylinder</h2>

<form action="Action\_1.php" method="POST">

<h3>   Enter radius : <input type=text name=r><br>  </h3>

<h3>   Enter height : <input type=text name=h><br><br>  </h3>

<input type=submit value=submit name=Show>

<input type=reset value=Reset name=Reset>

</form>

</center>

</body>

<html>

<?php

$r    =   $\_POST['r'];

$h   =   $\_POST['h'];

define('PI','3.14');

interface shape

{

    function area($r,$h);

    function vol($r,$h);

}

class cylinder implements *shape*

{

     function area($r,$h)

        {

           $area  =  2\*PI\*$r\*($r+$h);

            echo "<h3>The area of cylinder is :$area</h3>";

        }

    function vol($r,$h)

    {

        $vol  =  PI\*$r\*$r\*$h;

        echo "<h3>The volume of cylinder is :$vol</h3>";

    }

}

$c   =  new *cylinder*;

$c->area($r,$h);

$c->vol($r,$h);

?>

<html>

  <head>

    <style>

      body

      {

        background:#f4a9a8;

      }

    </style>

  </head>

  <body>

    <form action="Action\_2.php" method="POST">

      <center>

        <h2>Select any one shape below</h2>

        <label>Enter the height of triangle::</label><br><br>

        <input type="text" name="h1" id="h1"><br>

        <br>

        <label>Enter the base of triangle::</label><br><br>

        <input type="text" name="b1" id="b1"><br>

        <br><br>

        <label>Enter the side of square::</label><br><br>

        <input type="text" name="a1" id="a1"><br>

        <br><br>

        <label>Enter the length of rectangle::</label><br><br>

        <input type="text" name="l1" id="l1"><br>

       <br><br>

        <label>Enter the breadth of rectangle::</label><br><br>

        <input type="text" name="b2" id="b2"><br>

        <br><br>

        <label>Enter the radius of circle::</label><br><br>

        <input type="text" name="r1" id="r1"><br>

<br><br><br>

        <label ><h2>Select a shape for doing operation::</h2> </label><br><br>

        <input type="radio" name="op" value=1>TRIANGLE <br><br>

        <input type="radio" name="op" value=2>SQUARE <br><br>

        <input type="radio" name="op" value=3>RECTANGLE <br><br>

        <input type="radio" name="op" value=4>CIRCLE <br><br>

        <br><br><br>

        <input type="submit" value="submit">

        <input type="reset" value="reset">

<?php

define('pi',3.14);

class shape

{

   function area\_rect($l1,$b2)

   {

      return($l1\*$b2);

   }

   function area\_circle($r1)

   {

       return (pi\*$r1\*$r1);

   }

}

class triangle extends *shape*

{

    function area\_triangle($b1,$h1)

    {

    return($b1\*$h1)/2;

    }

}

class square extends *shape*

{

    function area\_square($a1)

    {

    return ($a1\*$a1);

    }

}

$ob=new *shape*();

$op=$\_POST['op'];

switch($op)

 {

    case 1 :   $h1=$\_POST['h1'];

                $b1=$\_POST['b1'];

                $ob1=new *triangle*();

                $a=$ob1->area\_triangle($h1,$b1);

                echo "Area of triangle is ::.$a";

                        break;

    case 2 :   $a1=$\_POST['a1'];

                $ob2=new *square*();

                $a=$ob2->area\_square($a1);

                echo "Area of square is ::.$a";

                        break;

    case 3 :   $l1=$\_POST['l1'];

                $b2=$\_POST['b2'];

                $a=$ob->area\_rect($l1,$b2);

                echo "Area of rectangle is ::.$a.";

                    break;

    case 4 :   $r1=$\_POST['r1'];

                $a=$ob->area\_circle($r1);

                echo "Area of circle is ::.$a.";

                    break;

 }

?>

<!DOCTYPE html>

<html>

    <head>

    </head>

    <body>

        <center>

        <form action="Action\_3.php" method="POST">

        <label >Teacher Code :: </label>

        <br>

        <input type="text" name="code" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher Name :: </label>

        <br>

        <input type="text" name="name" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher Qualification :: </label>

        <br>

        <input type="text" name="qual" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher Acc\_nos :: </label>

        <br>

        <input type="text" name="Acc\_nos" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher Joining date :: </label>

        <br>

        <input type="date" name="date" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher Basic pay :: </label>

        <br>

        <input type="text" name="basic\_pay" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher Earning :: </label>

        <br>

        <input type="text" name="earn" placeholder="comma separated values">

        <br><br><br>

        <label >Teacher deducation :: </label>

        <br>

        <input type="text" name="dedu" placeholder="comma separated values">

        <br><br><br>

        <label >Your Choice :: </label>

       <br><br>

        <input type="radio" name="op" value=1>For Master table <br><br>

        <label >To sort an entry :: </label><br>

        <br>

        <input type="radio" name="op" value=2>Sort <br><br>

        <label >To search an entry :: </label><br>

        <input type="text" name="search" ><br>

        <input type="radio" name="op" value=3>Search an entry <br><br>

        <input type="radio" name="op" value=4>Display salary of all

        teachers <br><br>

        <input type="submit" value="SUBMIT">

        <input type="reset" value="RESET"><br>

        </form>

    </center>

    </body>

</html>

<!-- Write PHP script to demonstrate the concept of introspection for examining

object.

 -->

 <?php

Class A

{

    private $x=10,$y;

    function \_\_construct($a)

    {

        $this->y=$a;

        echo "<br>";

        echo "constructor A is called!!";

        echo "<br>";

    }

    function displayA()

    {

        echo "x=$this->x,y=$this->y";

    }

}

Class B extends *A*

{

   private  $p,$q;

   function \_\_construct($l,$m,$n)

   {

       parent:: \_\_construct($l);

       echo "<br>";

       $this->p=$m;

       $this->q=$n;

       echo "<br>";

       echo "constructor of class B is called!!!";

       echo "<br>";

   }

   function displayB()

   {

    echo "<br>";

       echo "p=$this->p,q=$this->q";

       echo "<br>";

   }

}

$a=50;

$b=40;

$c=70;

$obj=new *B*($a,$b,$c);

$obj->displayA();

$obj->displayB();

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "FUNCTION EXAMING CLASSES<br>";

echo "CLASS A EXISTS OR NOT :: ";

echo class\_exists('A');

echo "<br>";

echo "CLASS C EXISTS OR NOT :: ";

echo class\_exists('C');

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "GET EXISTING ALL CLASSES<br>";

$all=get\_declared\_classes();

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "GET METHODS OF CLASS A<br>";

print\_r(get\_class\_methods('A'));

echo "<br>";

echo "GET METHODS OF CLASS B<br>";

print\_r(get\_class\_methods('B'));

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "GET VARIABLES OF CLASS A<br>";

var\_dump(get\_class\_vars('A'));

echo"<br>";

echo "GET VARIABLES OF CLASS B<br>";

var\_dump(get\_class\_vars('B'));

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "Parent of class B<br>";

echo "<br>".get\_parent\_class($obj);

echo "<br>".get\_parent\_class('B');

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "Parent of class A<br>";

echo "<br>".get\_parent\_class('A');

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo"<br>"."Function examin of OBJECT"."<br>";

echo"<br> obj is object or not";

echo is\_object($obj);

echo "<br>";

echo "obj1 is object or not<br>";

echo is\_object($obj1);//obj1 is not any object of any class

echo "<br>";

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "<br> GET OBJECT CLASS NAME";

echo get\_class($obj);

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "METHOD displayA() exists or not";

echo method\_exists($obj,'displayA');

echo "<br>";

echo "METHOD displayC() exists or not";

echo method\_exists($obj,'displayC');

echo "<br>"."\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"."<br>";

echo "GET VARIABLE BY USING OBJECT OF CLASS B";

print\_r(get\_object\_vars($obj));

echo"<br>";

?>

<html>

    <body >

    <center>

            <form action="<?php echo $\_SERVER['PHP\_SELF'];?>" method="POST" enctype="multipart/form-data">

               ID : <br><input type="text" name="id" placeholder=" comma separated values"><br><br>

               Name : <br><input type="text" name="name" placeholder="comma separated values"><br><br>

               Department : <br><input type="text" name="department" placeholder="comma separated values"><br><br>

               Salary : <br> <input type="text" name="salary" placeholder="comma separated values"><br><br>

               Bonus : <br><input type="text" name="bonus" placeholder="comma separated values"><br><br>

                <input type="submit" name="submit">

                <input type="reset" name="reset">

            </form>

            </center>

    <?php

        class Employee

        {

            protected $id;

            protected $name;

            protected $department;

            protected $salary;

            function \_\_construct($id,$name,$department,$salary)

            {

                $this->id = $id;

                $this->name = $name;

                $this->department = $department;

                $this->salary = $salary;

            }

            function \_\_destruct()

            {

               // echo "Object is detsroyed!!!";

            }

            function display()

            {

                echo "ID : ".$this->id."<br>";

                echo "Name : ".$this->name."<br>";

                echo "Department : ".$this->department."<br>";

                echo "Salary : ".$this->salary."<br>";

            }

        }

        class Manager extends *Employee*

        {

            private $bonus;

            function \_\_construct($id,$name,$department,$salary,$bonus)

            {

                parent::\_\_construct($id,$name,$department,$salary);

                $this->bonus = $bonus;

            }

            function \_\_destruct()

            {

               // echo "Object is detsroyed!!!";

            }

            function display()

            {

                parent::display();

                echo "Bonus : ".$this->bonus."<br>";

            }

            function gettotalsalary()

            {

                return $this->bonus + $this->salary;

            }

        }

        if($\_SERVER['REQUEST\_METHOD']=='POST')

        {

            $id = explode(' , ',$\_POST['id']);

            $name = explode(' , ',$\_POST['name']);

            $department = explode(' , ',$\_POST['department']);

            $salary = explode(' , ',$\_POST['salary']);

            $bonus = explode(' , ',$\_POST['bonus']);

            $emp = array();//empty array

            for($i=0;$i<6;$i++)

            {

                $emp[$i] = new *Manager*($id[$i],$name[$i],$department[$i],$salary[$i],$bonus[$i]);

            }

            $maxsalary = $emp[0]->gettotalsalary();

            $index = 0;

            $temp = 0;

            for($i=1;$i<6;$i++)

            {

                $temp = $emp[$i]->gettotalsalary();

                if($temp > $maxsalary)

                {

                    $maxsalary = $temp;

                    $index = $i;

                }

            }

            $emp[$index]->display();

        }

    ?>

    </body>

</html>

<html>

    <body>

            <form action="<?php echo $\_SERVER['PHP\_SELF'];?>" method="POST" enctype="multipart/form-data">

                Code : <input type="text" name="code" placeholder="comma separated values"><br><br>

                Name : <input type="text" name="name" placeholder="comma separated values"><br><br>

                Designation : <input type="text" name="designation" placeholder="comma separated values"><br><br>

                Account Number : <input type="text" name="accno" placeholder="comma separated values"><br><br>

                Join Date : <input type="text" name="joindate" placeholder="comma separated values"><br><br>

                Basic : <input type="text" name="basic" placeholder="comma separated values"><br><br>

                Earning : <input type="text" name="earning" placeholder="comma separated values"><br><br>

                Deduction : <input type="text" name="deduction" placeholder="comma separated values"><br><br>

                <input type="submit" name="submit">

            </form>

        <?php

        class employee

        {

            protected $ecode;

            protected $ename;

            protected $edesignation;

            function \_\_construct($ecode,$ename,$edesignation)

            {

                $this->ecode = $ecode;

                $this->ename = $ename;

                $this->edesignation = $edesignation;

            }

            function \_\_destruct(){}

            function display()

            {

                echo "Code : ".$this->ecode."<br>";

                echo "Name : ".$this->ename."<br>";

                echo "designation : ".$this->edesignation."<br>";

            }

        }

        class emp\_acc extends *employee*

        {

            protected $acccno;

            protected $jdate;

            function \_\_construct($ecode,$ename,$edesignation,$accno,$jdate)

            {

                parent::\_\_construct($ecode,$ename,$edesignation);

                $this->accno = $accno;

                $this->jdate = $jdate;

            }

            function \_\_destruct(){}

            function display()

            {

                parent::display();

                echo "accno : ".$this->accno."<br>";

                echo "join date : ".$this->jdate."<br>";

            }

        }

        class emp\_sal extends *emp\_acc*

        {

            private $basic;

            private $earning;

            private $deduction;

            function \_\_construct($ecode,$ename,$edesignation,$accno,$jdate,$basic,$earning,$deduction)

            {

                parent::\_\_construct($ecode,$ename,$edesignation,$accno,$jdate);

                $this->basic = $basic;

                $this->earning = $earning;

                $this->deduction = $deduction;

            }

            function \_\_destruct(){}

            function getsal()

            {

                $salary = $this->basic + $this->earning - $this->deduction;

                return $salary;

            }

            function display()

            {

                parent::display();

                echo "basic : ".$this->basic."<br>";

                echo "earning : ".$this->earning."<br>";

                echo "deduction : ".$this->deduction."<br>";

            }

        }

        if($\_SERVER['REQUEST\_METHOD']=='POST')

        {

                $code = $\_POST['code'];

                $name = $\_POST['name'];

                $designation = $\_POST['designation'];

                $accno = $\_POST['accno'];

                $joindate = $\_POST['joindate'];

                $basic = $\_POST['basic'];

                $earning = $\_POST['earning'];

                $deduction = $\_POST['deduction'];

                $codearr = explode(',',$code);

                $namearr = explode(',',$name);

                $designationarr = explode(',',$designation);

                $accnoarr = explode(',',$accno);

                $joindatearr = explode(',',$joindate);

                $basicarr = explode(',',$basic);

                $earningarr = explode(',',$earning);

                $deductionarr = explode(',',$deduction);

                $emp = array();

                $size = sizeof($codearr);

                for($i=0;$i<$size;$i++)

                {

                    $emp[$i] = new *emp\_sal*($codearr[$i], $namearr[$i], $designationarr[$i], $accnoarr[$i], $joindatearr[$i], $basicarr[$i], $earningarr[$i], $deductionarr[$i]);

                }

                $maxsal = $emp[0]->getsal();

                $temp = 0;

                $maxindex = 0;

                for($i=1;$i<$size;$i++)

                {

                    $temp = $emp[$i]->getsal();

                    if($temp > $maxsal)

                    {

                        $maxsal = $temp;

                        $maxindex = $i;

                    }

                }

                echo "<br>Employee having maximum salary : <br>";

                echo $emp[$maxindex]->display();

                $minsal = $emp[0]->getsal();;

                $temp = 0;

                $minindex = 0;

                for($i=1;$i<$size;$i++)

                {

                    $temp = $emp[$i]->getsal();

                    if($temp < $minsal)

                    {

                        $minsal = $temp;

                        $minindex = $i;

                    }

                }

                echo "<br><br>Employee having minimum salary : <br>";

                echo $emp[$minindex]->display();

        }

        ?>

    </body>

</html>