



Dr.M.G.R.
Educational and Research Institute
(DEEMED TO BE UNIVERSITY)
(An ISO Certified Institution)
University with Graded Autonomy Status
Maduravoyal , Chennai - 600 095



RECORD NOTEBOOK

BCS18ET3 – PHP/ MYSQL

2022 (ODD SEMESTER)

DEPARTMENT

OF

COMPUTER SCIENCE AND ENGINEERING

NAME : SREE HARI .T

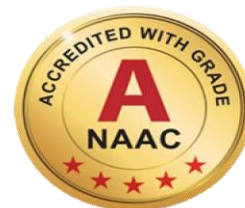
REGISTER NO : 191061101183

COURSE : B.TECH CSE

YEAR/SEM/SEC : III/VI/D



Dr.M.G.R.
Educational and Research Institute
(DEEMED TO BE UNIVERSITY)
(An ISO Certified Institution)
University with Graded Autonomy Status
Maduravoyal , Chennai - 600 095
BONAFIDE CERTIFICATE



Register No.:

191061101183

Subject Name : PHP / MYSQL

Subject Code : BCS18ET3

Department : COMPUTER SCIENCE AND ENGINEERING

Certified that this is the bonafide record of work done by
“SREEHARI.T” of ***“III Year B.Tech. (CSE), Sec- D”*** in the ***“PHP / MYSQL”*** during the year 2022.

Signature of Lab-in-Charge

Signature of Head of Dept.

Submitted for the Practical Examination held on

Internal Examiner

External Examiner

INDEX

Exp.No	Date	Name of the Experiment	Page No.	Staff Signature
1.		Determine whether a number is 'odd' or 'even'		
2.		Switch-Case Statement		
3.		Fibonacci Series Generation		
4.		Determine Factorial of a number		
5.		Class and object		
6.		Inheritance		
7.		Constructor And Destructor		
8.		Rows*Columns multiplication table generation using nested for loops		
9.		Display sorted array values		
10.		Student Registration Form		
11.		Inserting record into a database table using PHP/MySQL		
12.		Tracking the number of visits made to a page		

Ex.No : 1	DETERMINE WHETHER A NUMBER IS 'ODD' OR 'EVEN'
Date:	

AIM:

To write a 'PHP' program to determine whether a given number is 'odd' or 'even'.

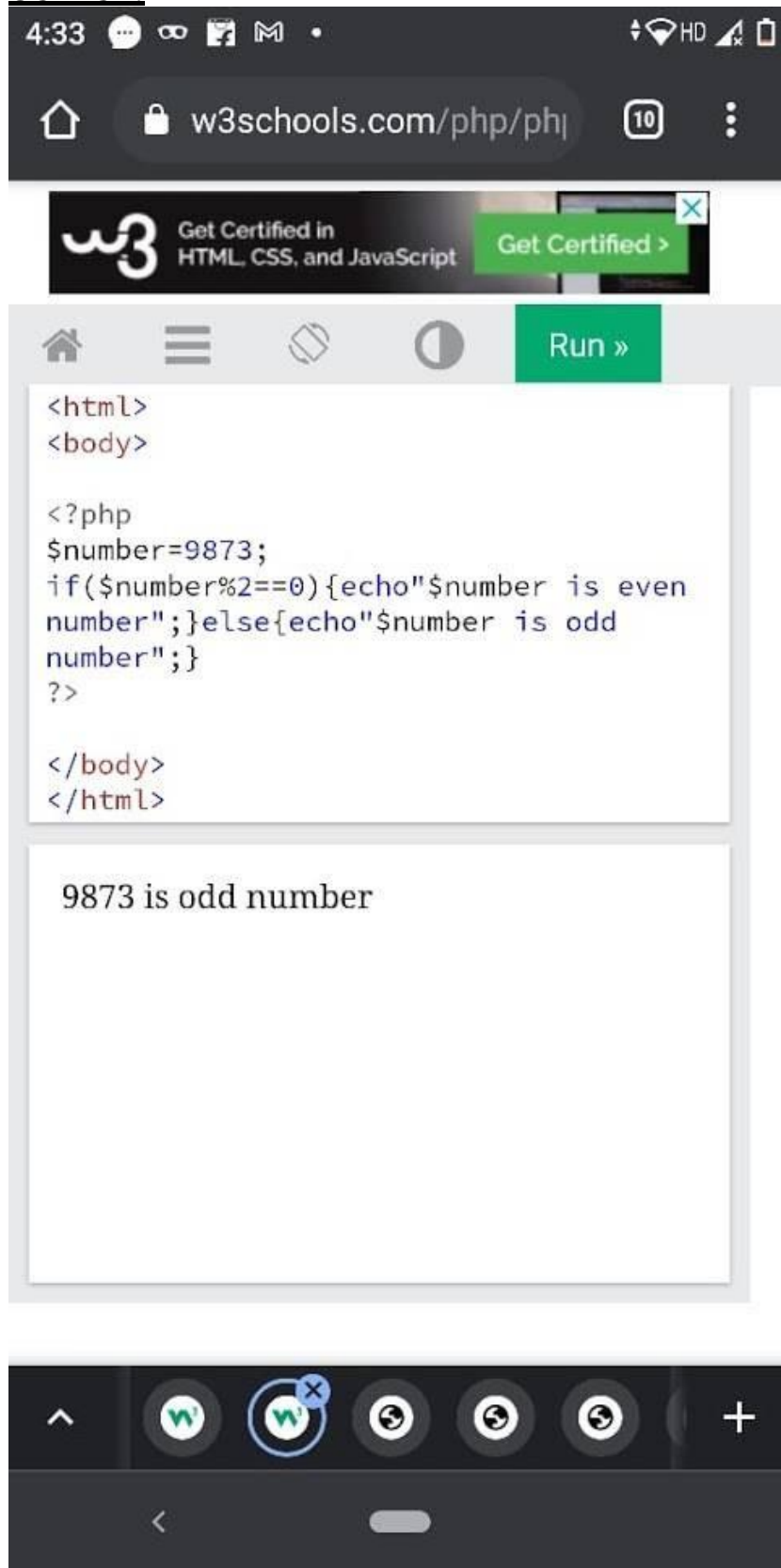
ALGORITHM:

1. Start
2. Read the number from the user.
3. Divide the number by 2.
4. If the remainder is equal to 0, print the number is 'even', else print it is 'odd'.
5. Stop.

PROGRAM:

```
<html>
<body>
<?php
$number 9873;
if ($number%2==0) { echo "$number is number"; }else[echo" $number is odd
number";}
?>
</body>
</html>
```

OUTPUT:



RESULT:

Hence, the 'PHP' program to determine whether a given number is 'odd' or 'even' is executed and the output verified.

Ex.No : 2	SWITCH-CASE STATEMENT
Date:	

AIM:

To write a 'PHP' program to implement the switch-case statement.

ALGORITHM:

1. Start
2. Read the favorite food from the user and assign it to a variable.
3. Compare the value of the variable against the different cases within the switch.
4. Print the message of the case against which the variable is matched.
5. Stop.

PROGRAM:

```
<?php
$favfood = "rice";
switch ($favfood) {
case "Chocolate":
echo "Your favorite food is Chocolate!";
break;
case "Curd":
echo "Your favorite food is Curd!";
break;
case "Tamarind":
echo "Your favorite food is Tamarind!";
break;
case "Briyani":
echo "Your favorite food is Briyani!";
break;
default:
echo "Your favorite food is neither rice, Chocolate, tomato,
```

nor rasam!";

}

?>

191061101183

OUTPUT:



The screenshot shows a mobile browser interface. At the top, the status bar displays the time 4:52, signal strength, HD, and battery level. The address bar shows the URL `tutorialspoint.com/execi`. The page header features the **codingground** logo with the tagline "SIMPLY EASY CODING". Below the header, there are tabs for "Execute", "Share", ".php", and "STDIN". The main code editor area contains the following PHP code:

```
1 <?php
2
3 $favfood = "rice";
4
5 switch ($favfood) {
6
7 case "chocolate":
8
9 echo "Your favorite food is
   chocolate!";
10
11 break;
12
13 case "rice":
14
15 echo "Your favorite food is rice!"
```

Below the code editor, the "Result" section shows the output of the script:

```
$php main.php
Your favorite food is rice!
```

At the bottom of the page, there is an advertisement for "Protect Your Business Today" with the subtext "Keeper Security". The mobile navigation bar at the very bottom shows a back arrow and a home button.

RESULT:

Hence, the 'PHP' program to implement the switch-case statement is executed and the output verified.

Ex.No : 3	FIBONACCI SERIES
Date:	

AIM:

To write a 'PHP' program to generate the Fibonacci series.

ALGORITHM:

1. Start
2. Assign the first two terms of the series with values 0 and 1 respectively.
3. Read the number of terms to be generated into a variable 'n'.
4. Using a loop generate the next terms of the series, by adding the previous two terms till the value 'n'.
5. Display the terms of the Fibonacci series.
6. Stop.

PROGRAM:<?php

```

$a=0;
$b=1;
echo "$a\n";
echo "$b\n";
for($x=2; $x<8;$x++)
{
$c=$a+$b;
$a=$b;
$b=$c;
echo "$c\n";
}
?>

```

OUTPUT:

```
<?php
```

```
$a=0;
$b=1;
echo "$a\n";
echo "$b\n";
for($x=2; $x<8;$x++)
{
    $c=$a+$b;
    $a=$b;
    $b=$c;
    echo "$c\n";
}
```

```
?>
```

```
0 1 1 2 3 5 8 13
```

RESULT:

Hence, the 'PHP' program to generate the Fibonacci series is executed and the output verified.

Ex.No : 4	DETERMINE THE FACTORIAL OF A NUMBER
Date:	

AIM:

To write a 'PHP' program to determine the factorial of a given number.

ALGORITHM:

1. Start
2. Read the number from the user into a variable 'num'.
3. Assign 1 to a variable 'factorial'.
4. Using loop initialize variable 'x' to the number and till it is decremented to 1, multiply factorial with the value 'x'.
5. Print the value of the variable 'factorial'.
6. Stop.

PROGRAM:

```
<?php
$num = 7;
$factorial = 1;
for ($x=$num; $x>=1; $x-)
{
    $factorial = $factorial * $x;
}
echo 'Factorial of $num is $factorial';
```

OUTPUT:

```
<?php
```

```
$num = 7;  
$factorial = 1;  
for ($x=$num; $x>=1; $x--)  
{  
    $factorial = $factorial * $x;  
}  
echo "Factorial of $num is $factorial";
```

```
?>
```

Factorial of 7 is 5040

RESULT:

Hence, the 'PHP' program to determine the factorial of a given number is executed and the output verified.

Ex.No : 5	CLASS AND OBJECT
Date	

AIM:

To write a 'PHP' program to create class and object.

ALGORITHM:

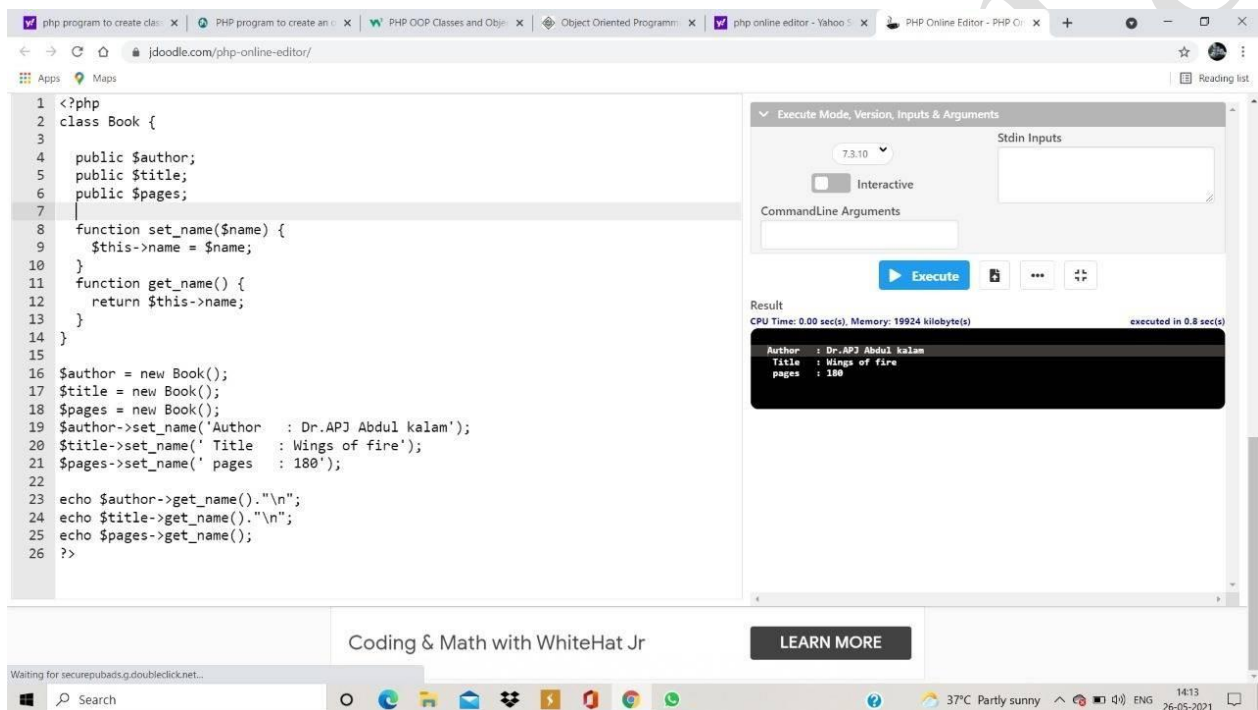
1. Start
2. Create a book class with properties author, title, pages.
3. Create an object for this class.
4. Assign the values for author, title, pages using the object.
5. Display the assigned values.
6. Stop.

PROGRAM:

```
<?php
class Book (
public $author;
public $title;
public $pages;
function set_name($name) {
$this->name = $name;
function get_name() { return $this->name;
}
PHP program to create an
jdoodle.com/php-online-editor/
W PHP OOP Classes and Ob X
Object Oriented Program
$author = new Book(); $title= new
Book(); $pages = new Book();
$author->set_name('Author $title->set_name(" Title
```

```
Spages->set_name(' pages
: Dr.APJ Abdul kalam');
: Wings of fire');
: 180');
echo $author->get_name()."\n";
echo $title->get_name()."\n"; echo Spages->get_name();
?>
```

OUTPUT:



The screenshot shows a web browser window with a PHP online editor. The editor contains a PHP script that defines a 'Book' class with 'set_name' and 'get_name' methods, and then creates three objects: \$author, \$title, and \$pages. The \$author object is set with 'Author : Dr.APJ Abdul kalam', \$title with 'Title : Wings of fire', and \$pages with 'pages : 180'. The script then echoes the get_name() method for each object. The right sidebar shows the execution mode set to 'Execute Mode, Version, Inputs & Arguments' with PHP version 7.3.10. The 'Execute' button has been clicked, and the output is displayed in a black box. The output shows the author, title, and pages information as defined in the code. The bottom of the browser shows a Windows taskbar with various icons and a system tray indicating the date and time as 14:13 on 26-05-2021.

```
1 <?php
2 class Book {
3
4     public $author;
5     public $title;
6     public $pages;
7
8     function set_name($name) {
9         $this->name = $name;
10    }
11    function get_name() {
12        return $this->name;
13    }
14 }
15
16 $author = new Book();
17 $title = new Book();
18 $pages = new Book();
19 $author->set_name('Author : Dr.APJ Abdul kalam');
20 $title->set_name(' Title : Wings of fire');
21 $pages->set_name(' pages : 180');
22
23 echo $author->get_name()."\n";
24 echo $title->get_name()."\n";
25 echo $pages->get_name();
26 ?>
```

Execute Mode, Version, Inputs & Arguments

7.3.10 Stdin Inputs

Interactive

CommandLine Arguments

Execute

Result

CPU Time: 0.00 sec(s), Memory: 19924 kilobyte(s) executed in 0.8 sec(s)

```
Author : Dr.APJ Abdul kalam
Title : Wings of fire
pages : 180
```

Coding & Math with WhiteHat Jr

LEARN MORE

Waiting for securepubads.g.doubleclick.net...

Search

37°C Partly sunny 14:13 26-05-2021

RESULT:

Hence, the 'PHP' program to create class and object is executed and the output verified.

Ex.No : 6	INHERITANCE
Date	

AIM:

To write a 'PHP' program to implement the concept of inheritance.

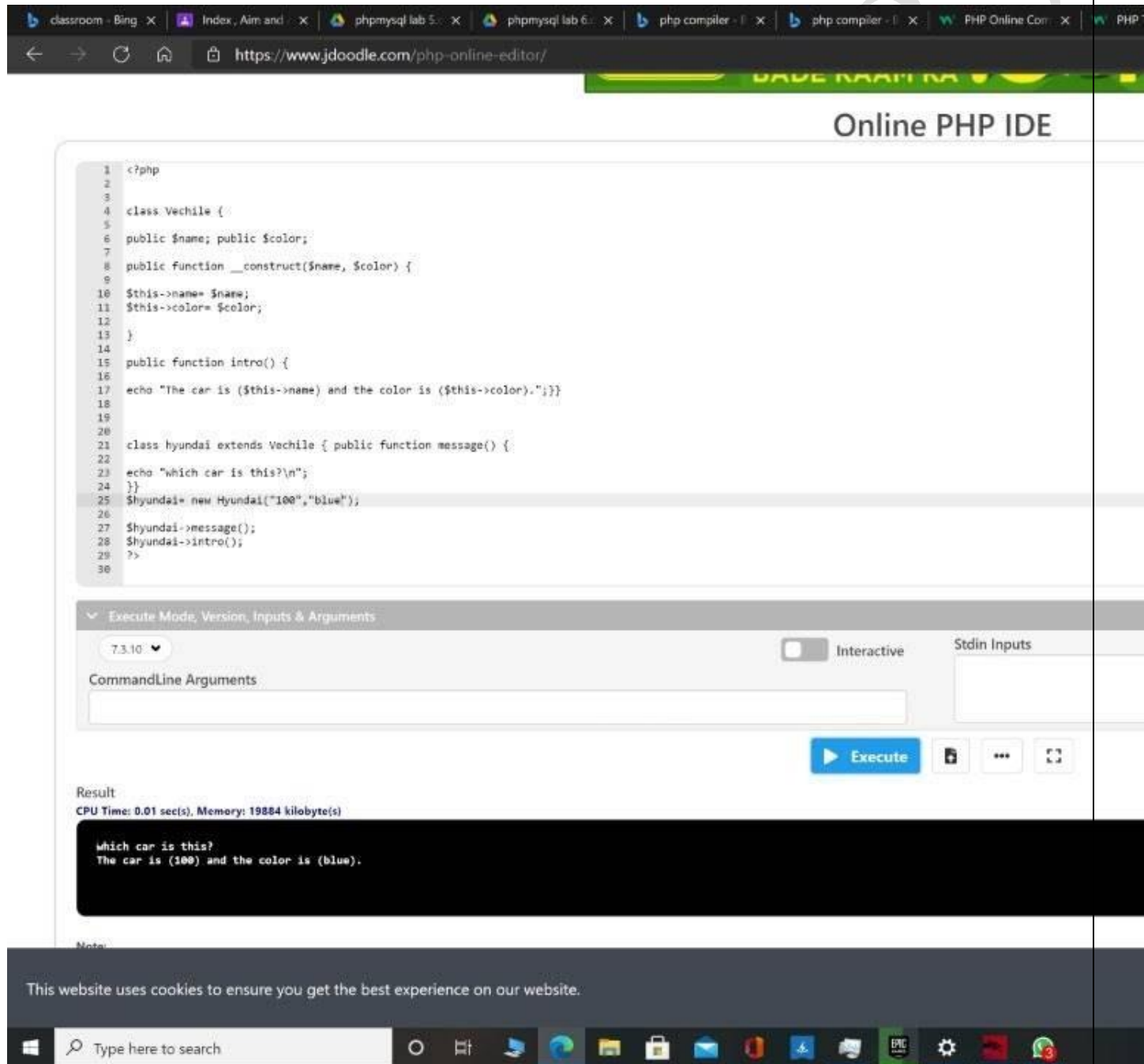
ALGORITHM:

1. Start
2. Create a parent class with property and methods defined.
3. Create a child class which can access the properties and methods of the parent class.
4. Stop.

PROGRAM:

```
<?php
class Vechile (
public $name;
public $color;
public function construct($name, $color) {
$this->name = $name;
$this->color = $color;
}
public function intro() { echo "The car is ($this->name) and the color is ($this->color).";
} }
class hyundai extends Vechile { public function message() {
echo "which car is this?\n";
} }
Shyundai = new Hyundai("120", "red"); Shyundai->message();
Shyundai->intro();
?>
```

OUTPUT:



The screenshot displays a web browser window with multiple tabs open, including 'classroom - Bing', 'Index, Aim and', 'phpmysql lab 5', 'phpmysql lab 6', 'php compiler', and 'PHP Online Com'. The active tab is 'https://www.jdoodle.com/php-online-editor/'. The browser's address bar shows the URL. Below the browser window, the 'Online PHP IDE' interface is visible. The code editor contains the following PHP code:

```
1 <?php
2
3
4 class Vechile {
5
6 public $name; public $color;
7
8 public function __construct($name, $color) {
9
10 $this->name= $name;
11 $this->color= $color;
12
13 }
14
15 public function intro() {
16
17 echo "The car is ($this->name) and the color is ($this->color).";}}
18
19
20
21 class hyundai extends Vechile { public function message() {
22
23 echo "which car is this?\n";
24 }}
25 $hyundai= new Hyundai("100","blue");
26
27 $hyundai->message();
28 $hyundai->intro();
29 ?>
30
```

Below the code editor, the 'Execute Mode, Version, Inputs & Arguments' section is visible. The version is set to '7.3.10'. The 'Interactive' checkbox is checked. The 'CommandLine Arguments' field is empty. The 'Stdin Inputs' field is also empty. The 'Execute' button is highlighted in blue. Below the 'Execute' button, the 'Result' section shows the output of the code execution:

```
Result
CPU Time: 0.01 sec(s), Memory: 19884 kilobyte(s)

which car is this?
The car is (100) and the color is (blue).
```

At the bottom of the browser window, a cookie notice is displayed: 'This website uses cookies to ensure you get the best experience on our website.'

RESULT:

Hence, the 'PHP' program to implement the concept of inheritance is executed and the output verified.

Ex.No : 7	CONSTRUCTOR AND DESTRUCTOR
Date	

AIM:

To write a 'PHP' program to indicate the use of constructor and destructor methods.

ALGORITHM:

1. Start
2. Create a class.
3. Define a construct method, user-defined method and destruct method within the class.
4. Create an object for the class to access the user-defined method.
5. Stop.

PROGRAM:**CONSTRUCTOR**

```
<?php
class Person {
    // first name of person
    private $fname;
    // last name of person
    private $lname;

    // Constructor
    public function __construct($fname, $lname) {
        echo "Initialising the object...<br/>";
        $this->fname = $fname;
        $this->lname = $lname;
    }
}
```

```
// public method to show name
public function showName() {
    echo "My name is: " . $this->fname . " " . $this->lname;
}
}
```

```
// creating class object
$john = new Person("John", "Wick");
$john->showName();
```

?>

DESTRUCTOR

```
<?php
class Person {
    // first name of person
    private $fname;
    // last name of person
    private $lname;

    // Constructor
    public function __construct($fname, $lname) {
        echo "Initialising the object...<br/>";
        $this->fname = $fname;
        $this->lname = $lname;
    }

    // Destructor
    public function __destruct(){
        // clean up resources or do something else
        echo "Destroying Object...";
    }

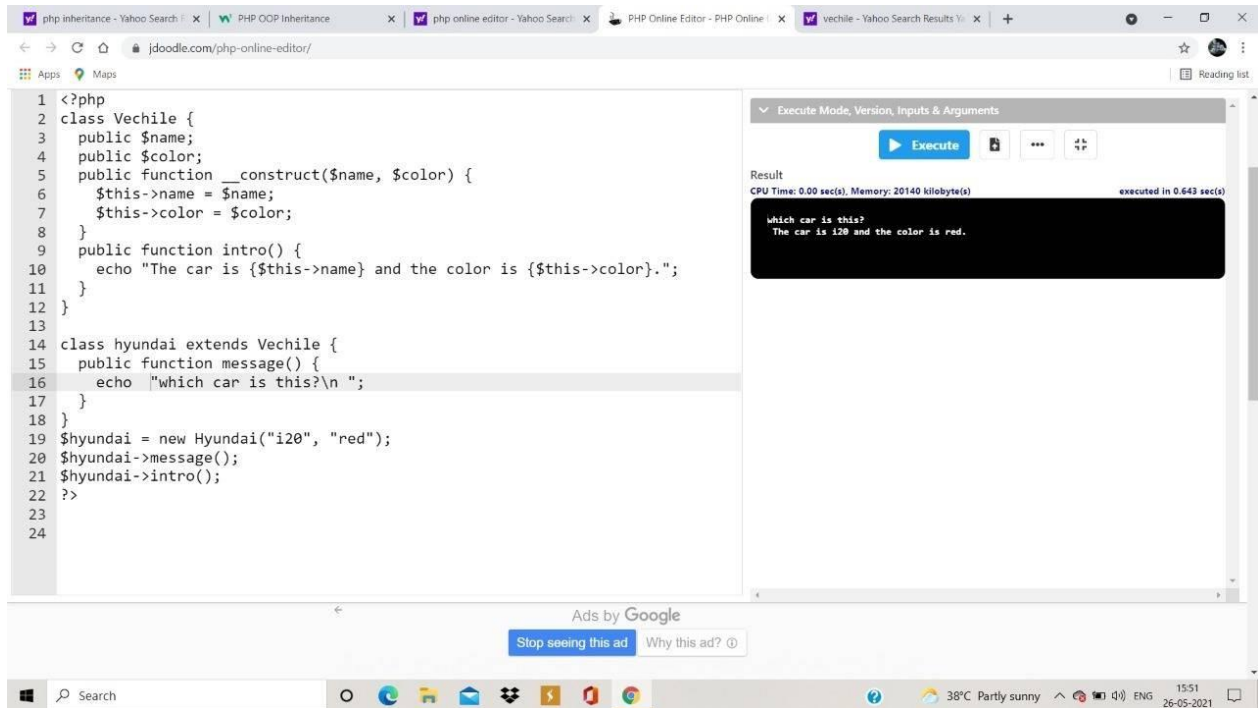
    // public method to show name
    public function showName() {
        echo "My name is: " . $this->fname . " " . $this->lname . "<br/>";
    }
}

// creating class object
$john = new Person("John", "Wick");
```

```
$john->showName();
```

```
?>
```

OUTPUT:



The screenshot shows a web browser window with multiple tabs. The active tab is 'PHP Online Editor - PHP Online'. The address bar shows 'jdoodle.com/php-online-editor/'. The editor contains the following PHP code:

```
1 <?php
2 class Vechile {
3     public $name;
4     public $color;
5     public function __construct($name, $color) {
6         $this->name = $name;
7         $this->color = $color;
8     }
9     public function intro() {
10        echo "The car is {$this->name} and the color is {$this->color}.";
11    }
12 }
13
14 class hyundai extends Vechile {
15     public function message() {
16         echo "which car is this?\n ";
17     }
18 }
19 $hyundai = new Hyundai("i20", "red");
20 $hyundai->message();
21 $hyundai->intro();
22 ?>
23
24
```

The right panel shows the execution results:

Execute Mode, Version, Inputs & Arguments

Execute

Result

CPU Time: 0.00 sec(s), Memory: 20140 kilobyte(s) executed in 0.643 sec(s)

which car is this?
The car is i20 and the color is red.

At the bottom of the browser window, there is an 'Ads by Google' banner with a 'Stop seeing this ad' button and a 'Why this ad?' link. The Windows taskbar is visible at the bottom, showing the search bar, taskbar icons, and system tray with the date '26-05-2021' and time '15:51'.

The screenshot shows a web browser window with multiple tabs. The active tab is 'PHP Online Editor - PHP OI'. The address bar shows 'jdoodle.com/php-online-editor/'. The main content area is divided into two panels. The left panel contains PHP code for a class 'Book' with attributes 'author', 'title', and 'pages', and methods 'set_name' and 'get_name'. The right panel shows the execution results, including the version '7.3.10', the 'Execute' button, and the output: 'Author : Dr.APJ Abdul kalam', 'Title : Wings of fire', and 'pages : 180'. The bottom of the browser shows a Windows taskbar with various icons and a system tray displaying '37°C Partly sunny' and '14:13 26-05-2021'.

```
1 <?php
2 class Book {
3
4     public $author;
5     public $title;
6     public $pages;
7
8     function set_name($name) {
9         $this->name = $name;
10    }
11    function get_name() {
12        return $this->name;
13    }
14 }
15
16 $author = new Book();
17 $title = new Book();
18 $pages = new Book();
19 $author->set_name('Author : Dr.APJ Abdul kalam');
20 $title->set_name(' Title : Wings of fire');
21 $pages->set_name(' pages : 180');
22
23 echo $author->get_name()."\n";
24 echo $title->get_name()."\n";
25 echo $pages->get_name();
26 ?>
```

Execute Mode, Version, Inputs & Arguments

7.3.10 Stdin Inputs

Interactive

CommandLine Arguments

Execute

Result

CPU Time: 0.00 sec(s), Memory: 19924 kilobyte(s) executed in 0.8 sec(s)

```
Author : Dr.APJ Abdul kalam
Title : Wings of fire
pages : 180
```

Coding & Math with WhiteHat Jr

LEARN MORE

Waiting for securepubads.g.doubleclick.net...

Search

37°C Partly sunny 14:13 26-05-2021

RESULT:

Hence, the 'PHP' program to indicate the use of constructor and destructor methods is executed and the output verified.

Ex.No : 8	ROWS * COLUMNS MULTIPLICATION TABLE GENERATION
Date:	

AIM:

To write a 'PHP' program to generate the multiplication table of rows and columns using nested for loops.

ALGORITHM:

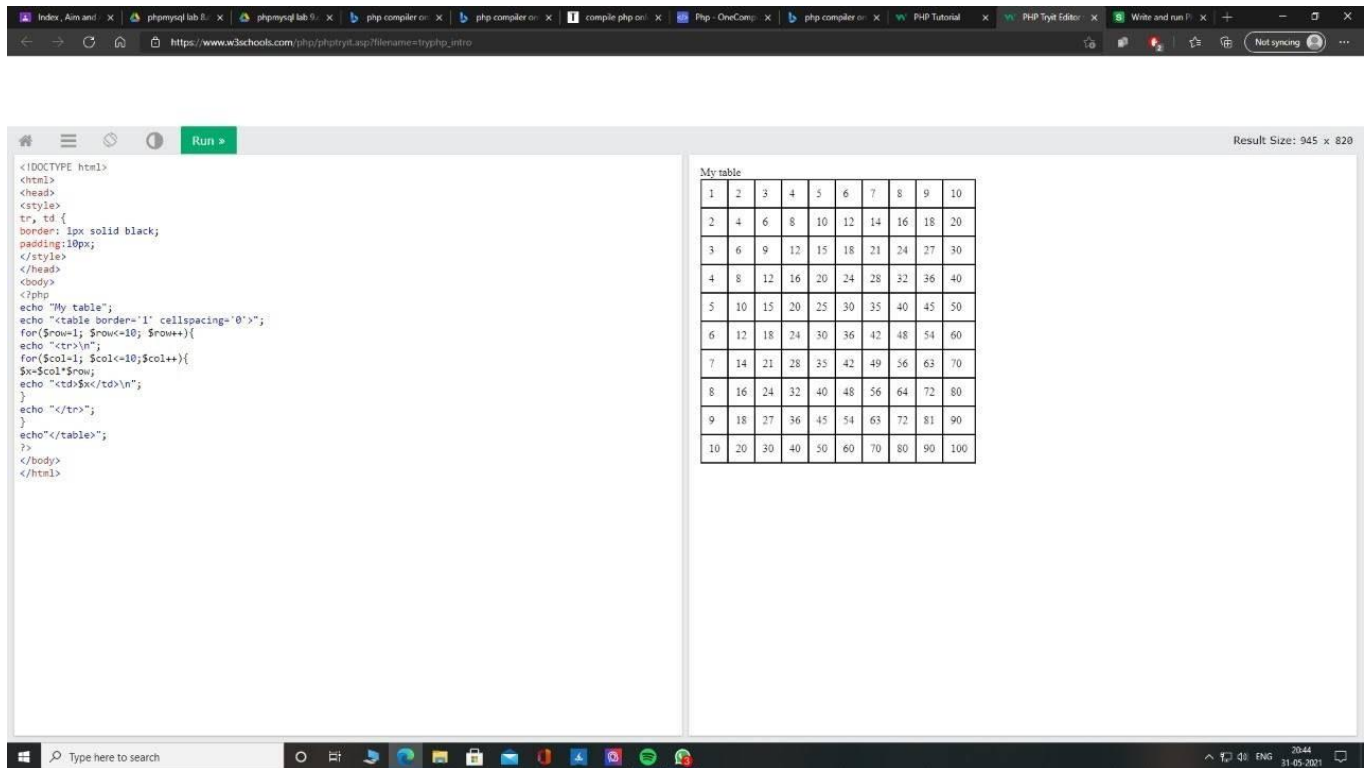
1. Start
2. Read the number of rows and the number of columns into 'row' and 'col' variable respectively.
3. Use an outer for loop for the row and inner for loop for the column.
4. Multiply the row and column values and display them column-wise.
5. Repeat step (4) for the remaining rows.
6. Stop.

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
<style>
tr,td {
border: 1px solid black;
padding:10px;
}
</style>
</head>
<body>
<?php
echo "My table";
echo "<table border='1' cellspacing='0'>"; for ($row=1;$row<=10; $row++)
echo "<tr>\n":
```

```
for($col=1;$col<=10;$col++) {  
$x=$col *$row;  
echo "<td>$x</td>\n"; }  
echo "</tr>";  
echo"</table>";  
?>  
</body>  
</html>
```

OUTPUT:



The screenshot shows a web browser window with the URL https://www.w3schools.com/php/phptryit.asp?filename=tryphp_intro. The browser displays the output of a PHP script, which is a 10x10 multiplication table. The table is titled "My table" and has a black border. The rows and columns are numbered 1 to 10. The table content is as follows:

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

RESULT:

Hence, the 'PHP' program to generate the multiplication table of rows and columns using nested loops is executed and the output verified.

Ex.No : 9	SORTED ARRAY VALUES
Date:	

AIM:

To write a 'PHP' program to create an array of different cities and display them in the sorted order as an unordered list.

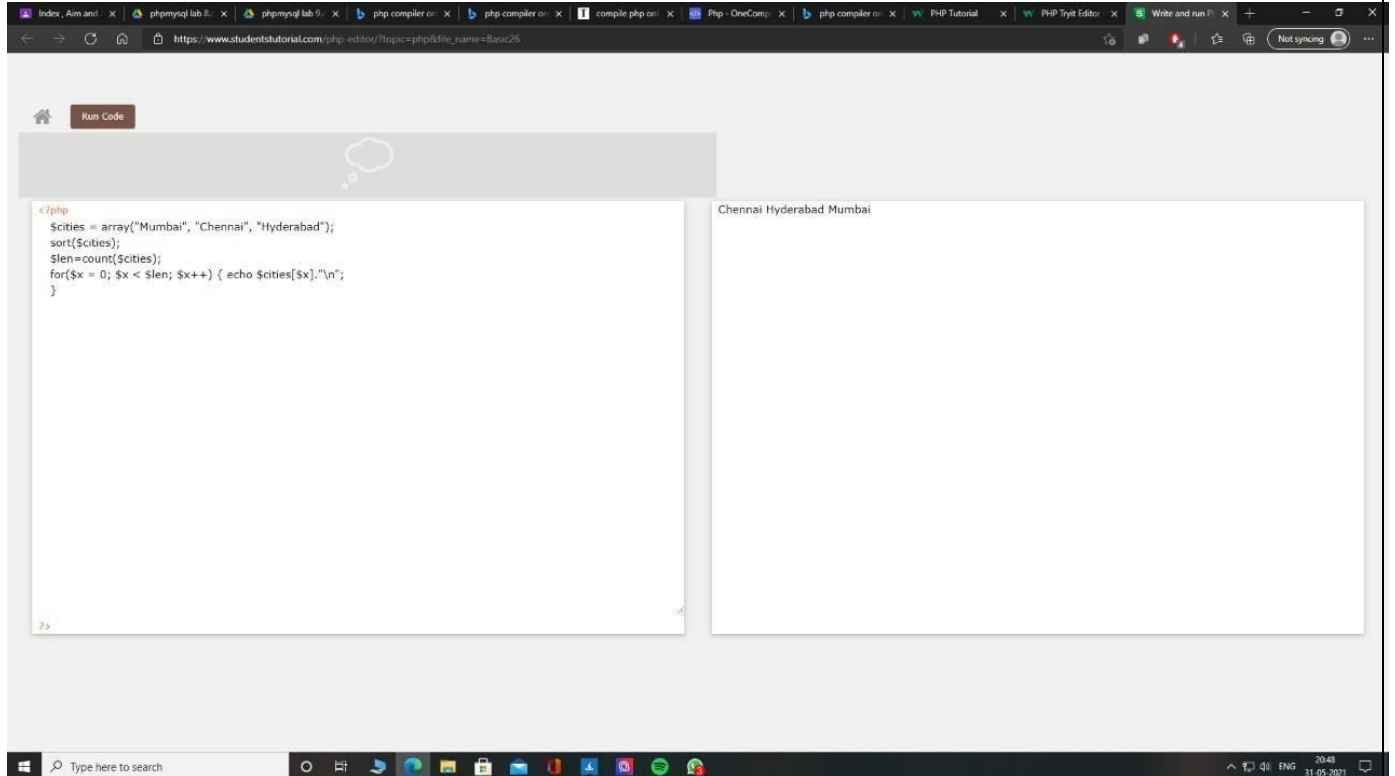
ALGORITHM:

1. Start
2. Initialize an array with names of different cities.
3. Display the values of the original array as an unordered list.
4. Apply the sort function to the original array.
5. Display the values of the sorted array as an unordered list.
6. Stop.

PROGRAM:

```
<?php
$cities = array("Mumbai", "Chennai", "Hyderabad");
sort($cities);
$len
= count($cities);
for($x = 0; $x < $len; $x++) {
echo $cities [$x]."\n";
}
?>
```

OUTPUT:



The screenshot shows a web browser window with multiple tabs. The active tab is titled "https://www.students-tutorial.com/php-editor/?topic=php&file_name=Basic26". The browser's address bar shows the URL. Below the browser window, there is a PHP code editor interface. The code editor has a "Run Code" button at the top left. The code being executed is as follows:

```
<?php
$cities = array("Mumbai", "Chennai", "Hyderabad");
sort($cities);
$len=count($cities);
for($x = 0; $x < $len; $x++) { echo $cities[$x]."\n";
}
```

The output of the code is displayed on the right side of the editor, showing the cities in sorted order: "Chennai Hyderabad Mumbai".

RESULT:

Hence, the 'PHP' program to create an array of different cities and display them in sorted order is executed and the output verified.

Ex.No : 10

Date:

STUDENT REGISTRATION FORM

AIM:

To write a 'PHP' program to design a student registration form and display the submitted details on to another page.

ALGORITHM:

1. Start
2. Design a student registration form in register.php page.
3. Submit the registered details to success.php page and display them.
4. Stop.

PROGRAM:

[6:26 PM, 7/2/2021] Hari: <html> <body>

<form action="target.php" method="get">

Name: <input type="text" name="name">

E-mail: <input type="text" name="email">
 Mobile Phn: <input type="text" name="Mobile-phn">
 <input type="submit">

</form>

</body>

</html>

PHP

<html>

<body>

Welcome <?php echo \$_GET["name"]; ?>

Your email address is: <?php echo \$_GET["email"]; ?>
 Your Mobile number is: ?php echo \$_GET["Mobile-phn"]; ?>

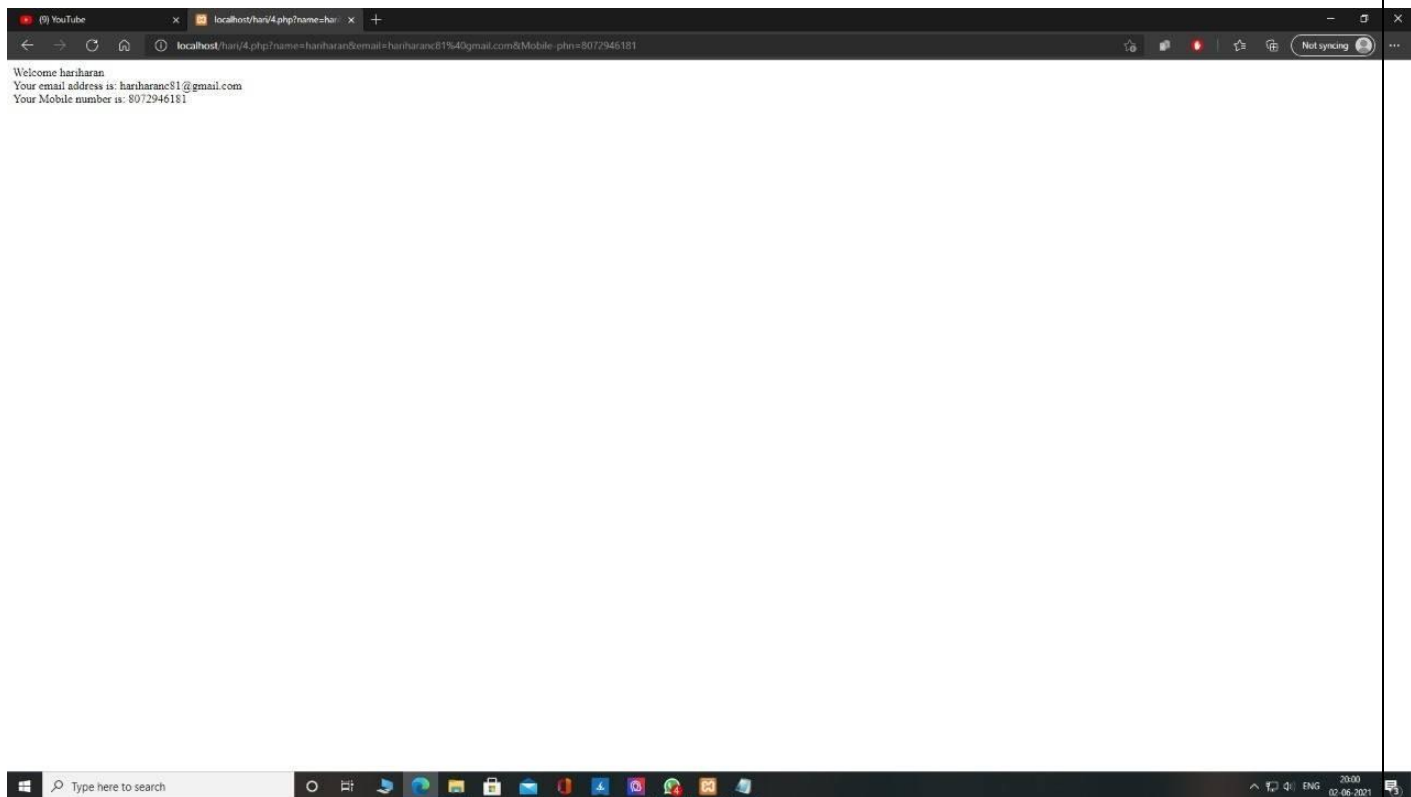
```
</body>  
</html>
```

OUTPUT:



A screenshot of a web browser window displaying a form. The browser's address bar shows 'localhost/hari/3.php'. The form contains three input fields: 'Name' with the value 'harikiran', 'E-mail' with the value 'harikiran87@gmail.com', and 'Mobile Ptn' with the value '9072946181'. Below these fields is a 'Submit' button. The browser's status bar at the bottom indicates 'Not syncing'.





RESULT:

Hence, the 'PHP' program to design a student registration form and display the submitted details on to another page is executed and the output verified.

Ex.No : 11	INSERTING RECORD INTO A TABLE USING PHP/MySQL
Date:	

AIM:

To write a 'PHP/MySQL' program to insert a record into a database table.

ALGORITHM:

1. Start
2. Assign server name, user name, password, dbname , conn variables with the required values.
3. Check if the connection was successful. If not, print connection failed, else, proceed to step (4).
4. Insert a record and assign that SQL query to sql object.
5. Check the execution of the query with the conn object. If successful, print "record inserted successfully" else print "error".
6. Close the connection.
7. Stop.

PROGRAM:

insert.php

```
<html>
<style>
form,input
{
padding:10px;
margin:10px;
}
</style>
<body>
<title>Insert Data</title>
```

```
<form action="Main.php" method="POST">
<input type="text" name="Name" placeholder="Name"><br>
<input type="email" name="Email" placeholder="Email Id"><br>
<input type="mobile" name="Mobile" placeholder="Mobile-number"><br>
<input type="text" name="Department" placeholder="Department"><br>
<input type="submit" name="submit">
</form>
```

```
</body>
```

```
</html>
```

Main.php

```
<?php
```

```
$con = mysqli_connect('localhost','root','');
```

```
if(!$con)
```

```
{
```

```
    echo "Not Connected to server";
```

```
}
```

```
if (!mysqli_select_db($con,'computer'))
```

```
{
```

```
    echo "Database is not selected";
```

```
}
```

```
$Name = $_POST['Name'];
```

```
$Email = $_POST['Email'];
```

```
$Mobile = $_POST['Mobile'];
```

```
$Department = $_POST['Department'];
```

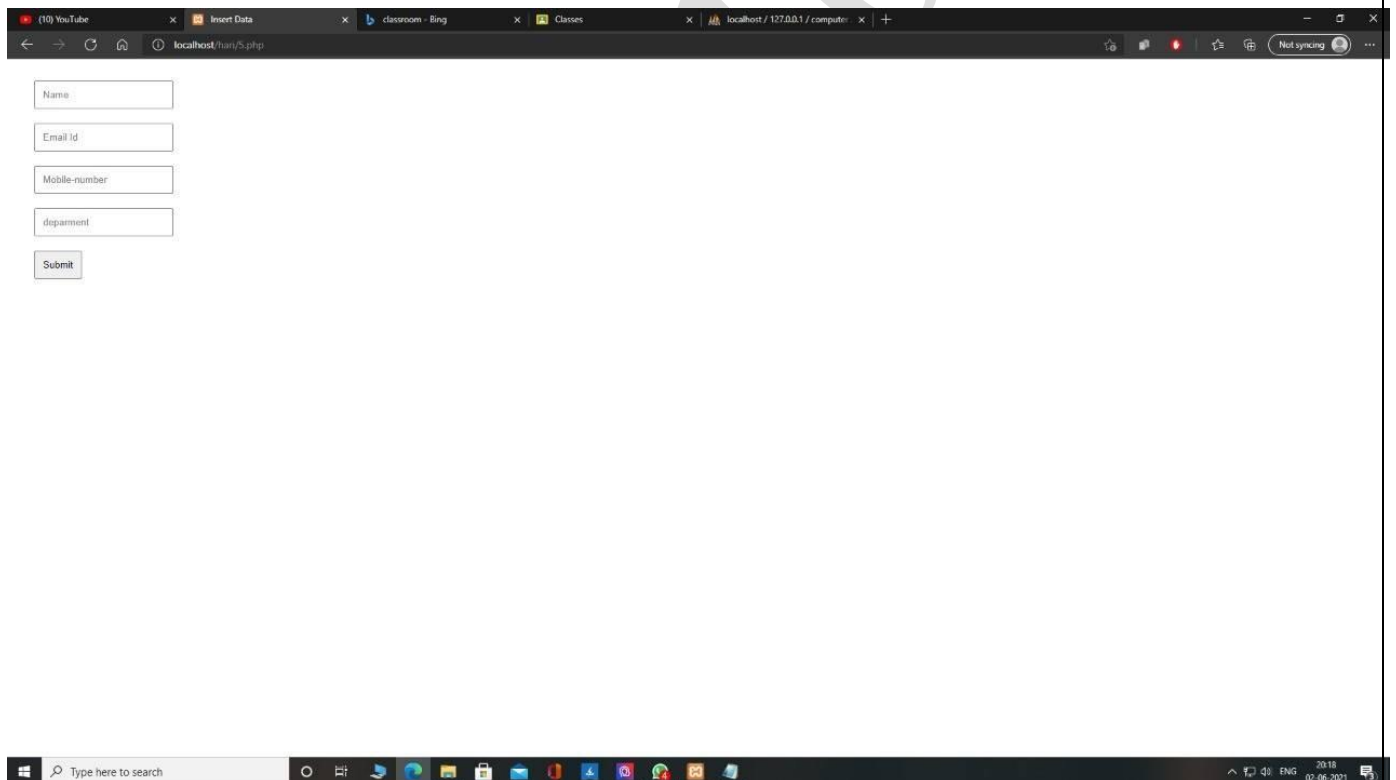
```
$sql="INSERT INTO student (Name,Email,Mobile,Department) VALUES
```

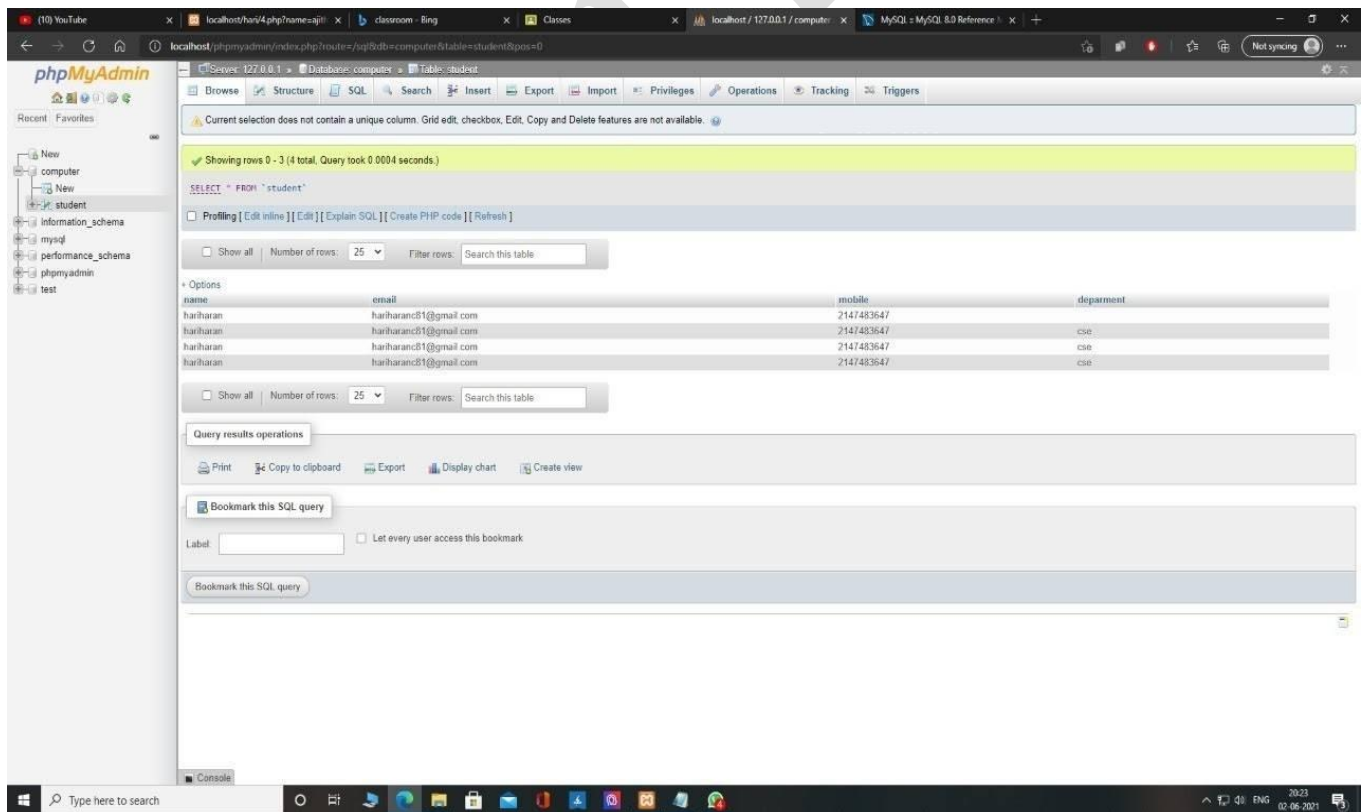
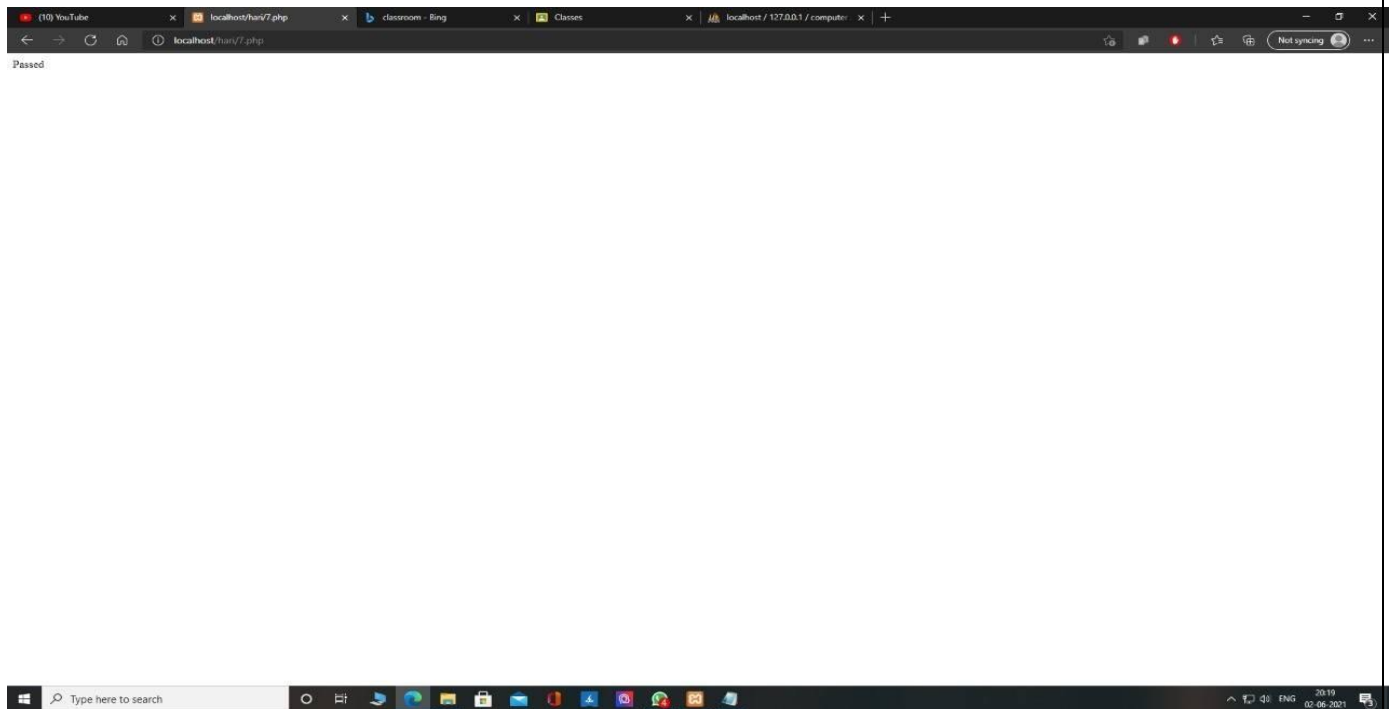
```
(' $Name','$Email','$Mobile','$Department')";
```

```
if (!mysqli_query($con,$sql))
```

```
{  
    echo "Failed";  
}  
else  
{  
    echo "Passed";  
}  
header("refresh:2; url=insert.php");  
?>
```

OUTPUT:





RESULT:

Hence, the 'PHP/MySQL' program to insert a record into a database table is executed and the output verified.

Ex.No : 12	TRACKING THE NUMBER OF VISITS MADE TO A PAGE
Date:	

AIM:

To write a 'PHP' program to track the number of visits made to a page.

ALGORITHM:

1. Start
2. Initialize the session.
3. If the session isset is true, increment the counter value by 1, otherwise, do not alter the initial counter value which was 1.
4. Print the number of times the page was visited.
5. Stop.

PROGRAM:

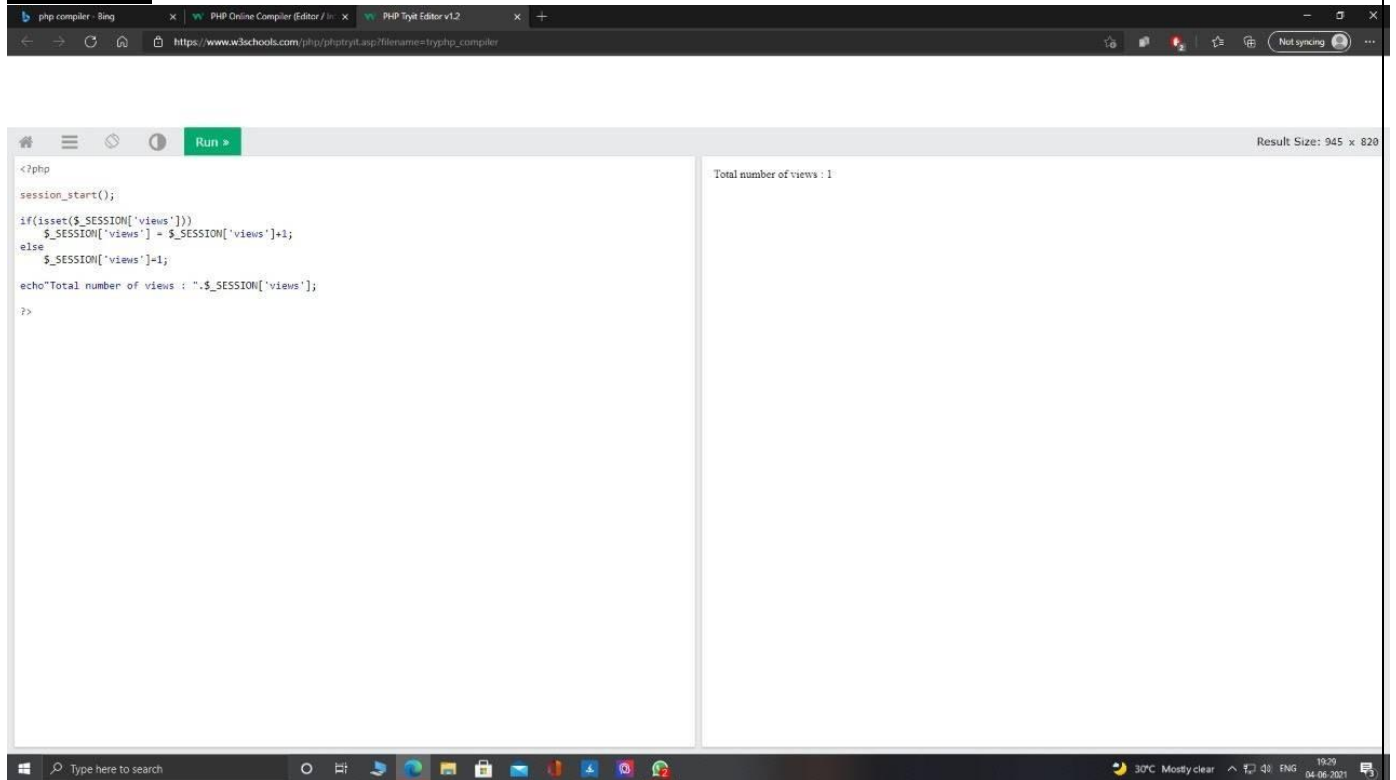
```
<?php
session_start();

if(isset($_SESSION['views']))
    $_SESSION['views'] = $_SESSION['views']+1;
else
    $_SESSION['views']=1;

echo"Total number of views : ".$_SESSION['views'];

?>
```

OUTPUT:



```
<?php
session_start();

if(isset($_SESSION['views']))
    $_SESSION['views'] = $_SESSION['views']+1;
else
    $_SESSION['views']=1;

echo"Total number of views : ".$_SESSION['views'];
?>
```

Total number of views : 1

RESULT:

Hence, the 'PHP' program to track the number of visits made to a page is executed and the output verified.