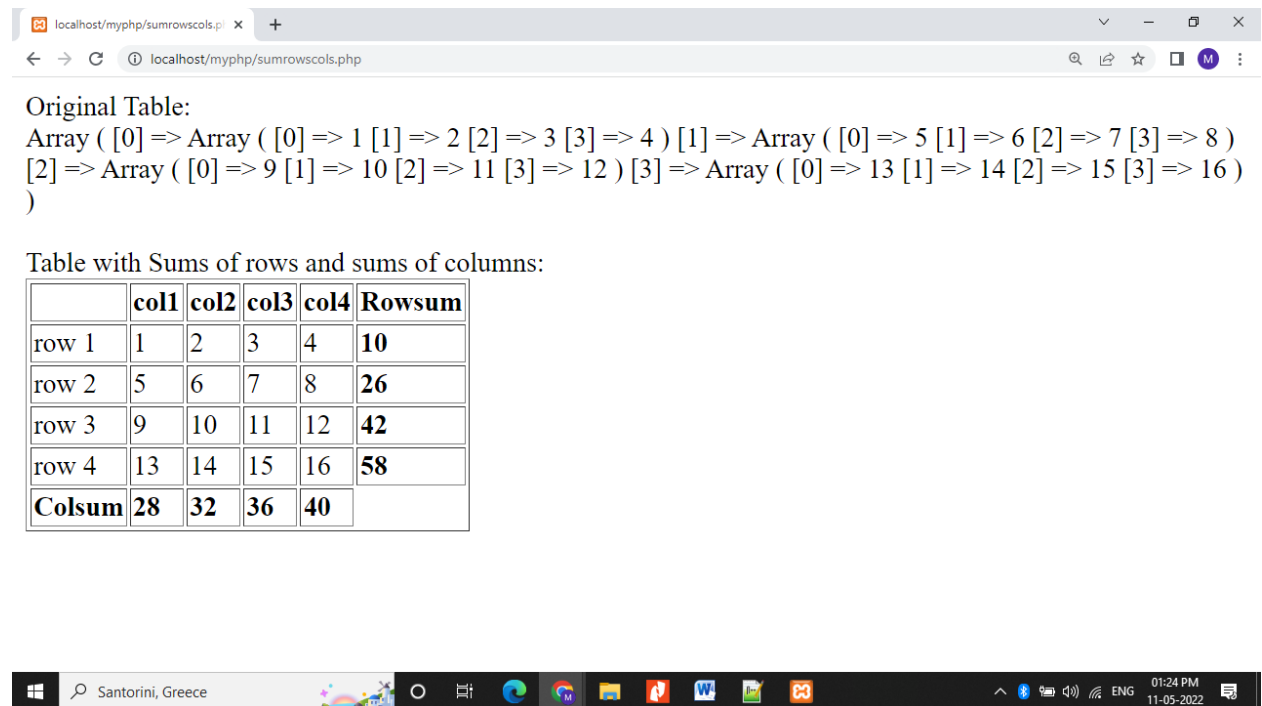


1. Sum of columns and rows in a table

Source code:

```
<?php
    $rows = 4;
    $cols = 4;
    $arr = array(
        array(1,2,3,4),
        array(5,6,7,8),
        array(9,10,11,12),
        array(13,14,15,16)
    );
    echo "Original Table:<br>";
    print_r($arr);
    echo '<br><table border="1">';
    echo "<tr><th><th>col1<th>col2<th>col3<th>col4<th>Rowsum</tr>";
    echo "<br>";
    echo "Table with Sums of rows and sums of columns: <br>";
    for($i=0; $i<$rows; $i++)
    {
        echo "<tr><td>row ".$i+1;
        $rowsum = 0;
        for($j=0; $j<$cols; $j++)
        {
            echo "<td>".$arr[$i][$j];
            $rowsum = $rowsum + $arr[$i][$j];
        }
        echo "<td><b>".$rowsum."</b></tr>";
    }
    echo "<tr><td><b>Colsum</b>";
    for($j=0; $j<$cols; $j++)
    {
        $colsum = 0;
        for($i=0; $i<$rows; $i++)
        {
            $colsum = $colsum + $arr[$i][$j];
        }
        echo "<td><b>".$colsum."</b>";
    }
    echo "</tr></table>";
?>
```

Output



Original Table:
Array ([0] => Array ([0] => 1 [1] => 2 [2] => 3 [3] => 4) [1] => Array ([0] => 5 [1] => 6 [2] => 7 [3] => 8) [2] => Array ([0] => 9 [1] => 10 [2] => 11 [3] => 12) [3] => Array ([0] => 13 [1] => 14 [2] => 15 [3] => 16))

Table with Sums of rows and sums of columns:

	col1	col2	col3	col4	Rowsum
row 1	1	2	3	4	10
row 2	5	6	7	8	26
row 3	9	10	11	12	42
row 4	13	14	15	16	58
Colsum	28	32	36	40	

2. Sum of first N prime Numbers

Source code:

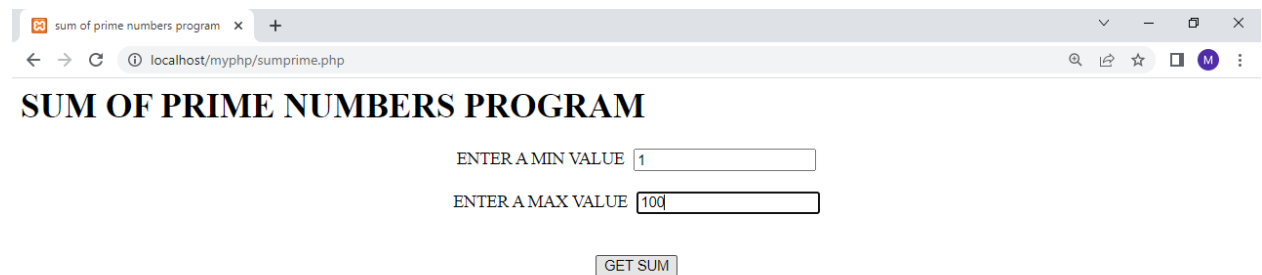
Source Code:

```
<html>
  <head>
    <title> sum of prime numbers program</title>
    <h1>SUM OF PRIME NUMBERS PROGRAM</h1>
  </head>
  <body>
    <form method="POST">
      <center>
        ENTER A MIN VALUE &nbsp;<input type="text" name="number1"><br/> <br/>
        ENTER A MAX VALUE &nbsp;<input type="text" name="number2">
        <br/><br/><br/>
        <input type="submit" value="GET SUM">
      </center>
    </body>
</html>
<?php
  if($_POST)
  {
    echo "<br/><br/><br/><b><center>SUM OF THE PRIME NUMBER IS</b>";
    $min = $_POST['number1'];
    $max = $_POST['number2'];
```

```
$sum = 0;
for($num = $min;$num<=$max;$num++)
{
    $count = 0;
    for($i=2;$i<=sqrt($num);$i++)
    {
        if($num % $i == 0)
        {
            $count++;
            break;
        }
    }
    if($count==0 && $num!= 1)
        $sum = $sum + $num;
}
echo "<b> $sum </b></center>";
}

?>
```

Output:



sum of prime numbers program x

localhost/myphp/sumprime.php

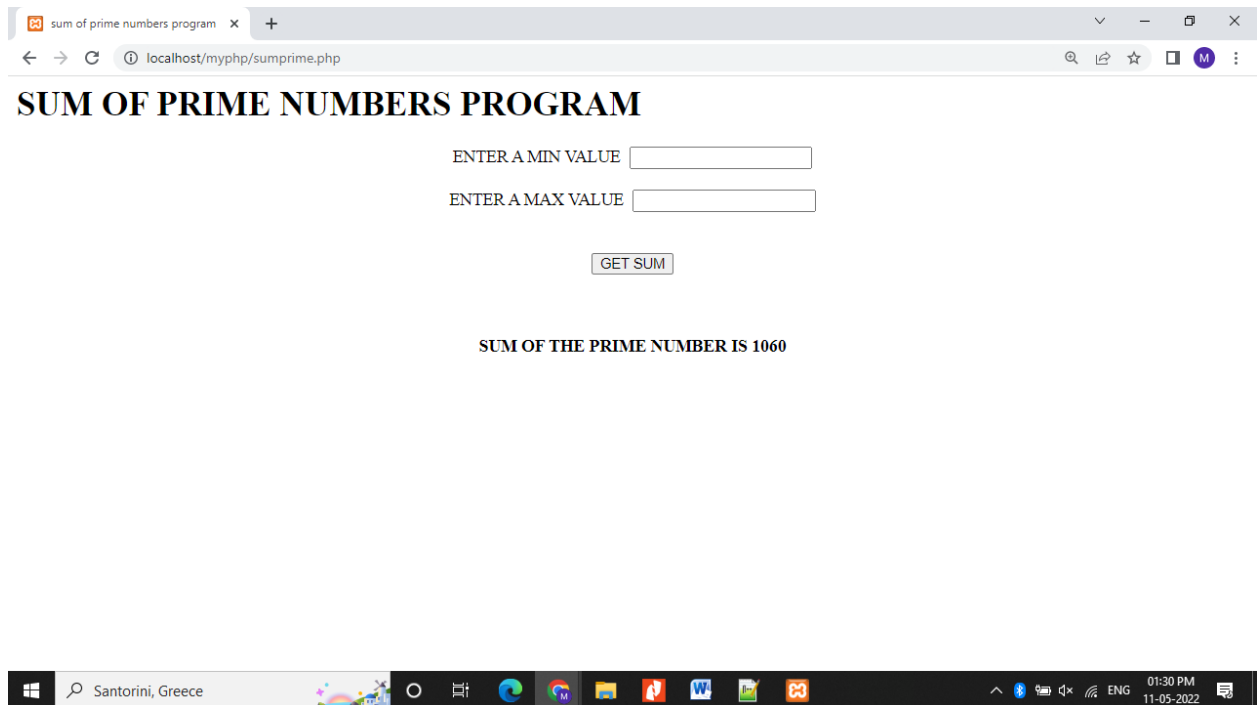
SUM OF PRIME NUMBERS PROGRAM

ENTER A MIN VALUE

ENTER A MAX VALUE

GET SUM





3. Finding an valid Email Address

Source Code:

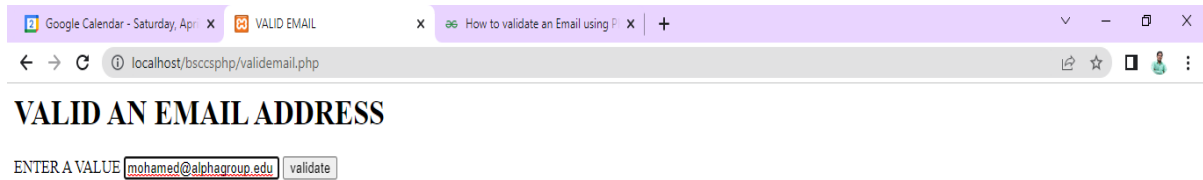
```
<html>
  <head>
    <title>VALID EMAIL</title>
    <h1>VALID AN EMAIL ADDRESS</h1>
  </head>
  <body>
    <form method="POST">
      ENTER A VALUE
      <input type="text" name="EMAIL">
      <input type="submit" value="validate">
    </body>
</html>
<?php
  if($_POST)
  {
    echo "<br/><br/><br/> <br/><br/>";
    $email = $_POST['EMAIL'];
    if (filter_var($email, FILTER_VALIDATE_EMAIL))
    {
      echo("$email is a valid email address");
    }
    else
    {
```

```

        echo("$email is not a valid email address");
    }
}
?>

```

Output:



4. Converting a Number written in words to digits

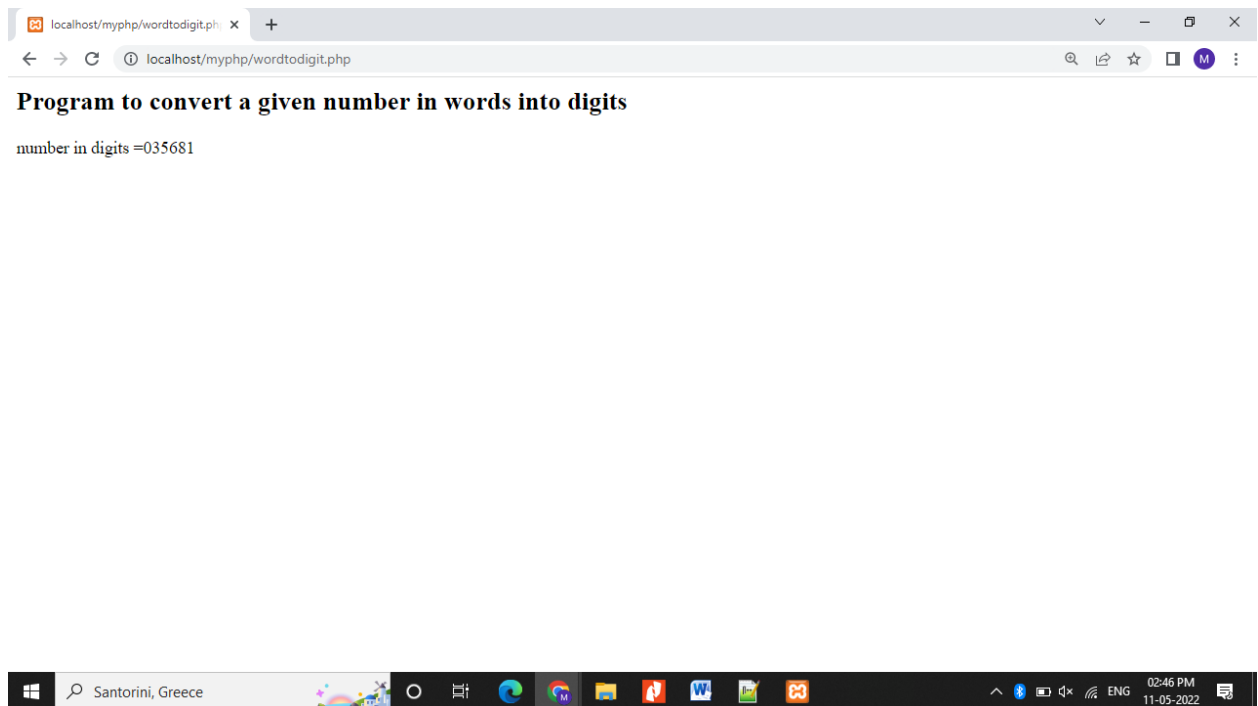
Source Code:

```

<?php
function digit($word)
{
    $warr = explode(' ', $word);
    $result = "";
    foreach($warr as $value)
    {
        switch(trim($value))
        {
            case 'zero'      : $result .= '0'; break;
            case 'one'       : $result .= '1'; break;
            case 'two'       : $result .= '2'; break;
            case 'three': $result .= '3'; break;
            case 'four'      : $result .= '4'; break;
            case 'five'      : $result .= '5'; break;
            case 'six'       : $result .= '6'; break;
            case 'seven': $result .= '7'; break;
            case 'eight': $result .= '8'; break;
            case 'nine'      : $result .= '9'; break;
        }
    }
    return $result;
}
echo "<h2>Program to convert a given number in words into digits</h2>";
echo "number in digits = ".digit("zero three five six eight one");
?>

```

Output:

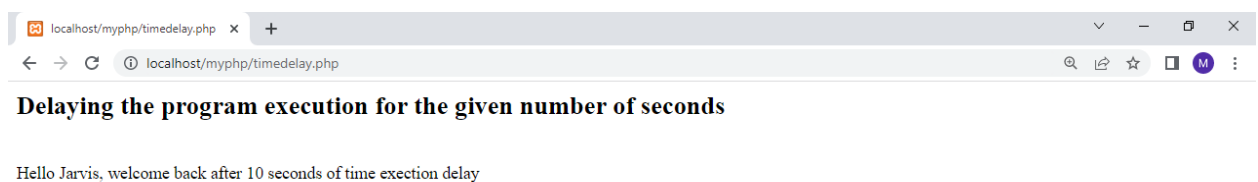


5. Delaying the Program Execution for given number of seconds

Source Code:

```
<?php
    //current time
    echo "<h2>Delaying the program execution for the given number of seconds</h2>";
    sleep(10);
    $username = "Jarvis";
    echo "<br>Hello $username, welcome back after 10 seconds of time execution delay";
?>
```

Output:

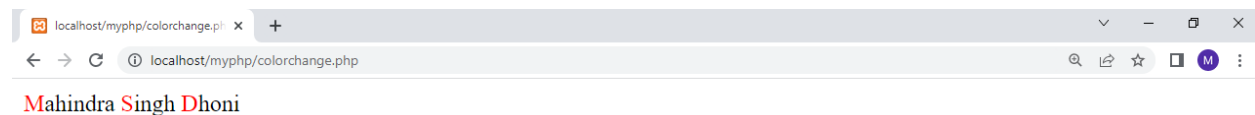


6. Changing the color of the first character of given set of words

Source code:

```
<?php
$txt = "Mahindra Singh Dhoni";
// The preg_replace is used here to replace the
// color of first character of the word
$txt = preg_replace('/(\b[a-z])/i',
'<span style="color:red;">\1</span>', $txt);
// Display the text value
echo $txt;
?>
```

OUTPUT:



7. Generating the Multiplication table of a number

Source code:

```
<html>
  <head>
    <title> MULTIPLICATION TABLE</title>
    <h1><center> MULTIPLICATION TABLE</center></h1>
  </head>
  <body>
    <form method="POST">
      Enter Table Value
      <input type="text" name="number">
      <input type="submit" value="Result">
    </body>
</html>
<?php
  if($_POST)
  {
    $num = $_POST['number'];
    echo "<br/><br/><br/><b>Multiplication table of $num is given
below</b> <br/><br/>";
    for($i = 1; $i <=15; $i++)
    {
```

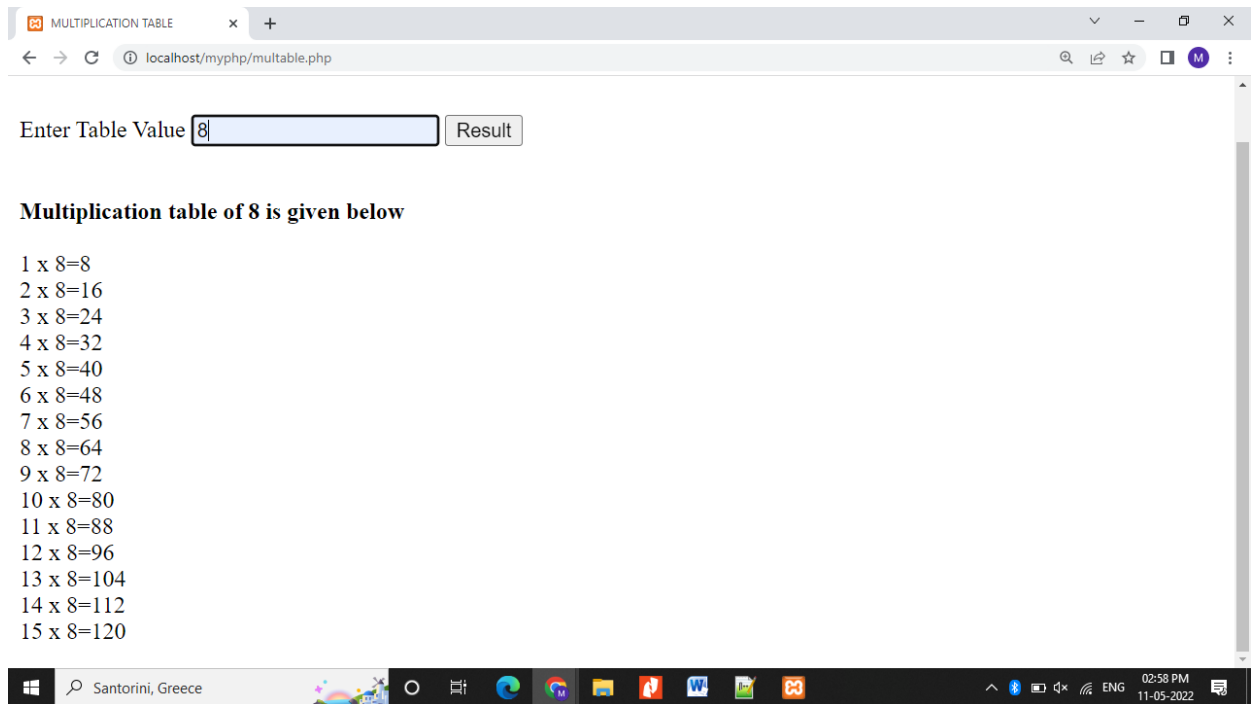
```

        echo "$i x $num=", $i * $num;
        echo "<br/>";
    }
}

?>

```

Output:



8. Calculating the factorial of a number

Source code:

```

<html>
    <head>
        <title> FACTORIAL </title>
        <h1><center> FINDING FACTORIAL OF A GIVEN NUMBER</center></h1>
    </head>
    <body>
        <form method="POST">
            Enter A NUMBER
            <input type="text" name="number">
            <input type="submit" value="Result">
        </body>
</html>
<?php
    if($_POST)
    {

```



```

        echo "<br/><br/>";
        $num = $_POST['number'];
        function fact($num)
        {
            if($num == 0 || $num == 1)
                return 1;
            else
                return $num * fact($num - 1); //recursive function call
        }
        echo "factorial=",fact($num); //function call
    }
?>

```

Output:



9. Reading a file, reversing its contents and writing it to a new File

Source Code:

angry.txt

hai, are you angry on me ?

Fileread.php

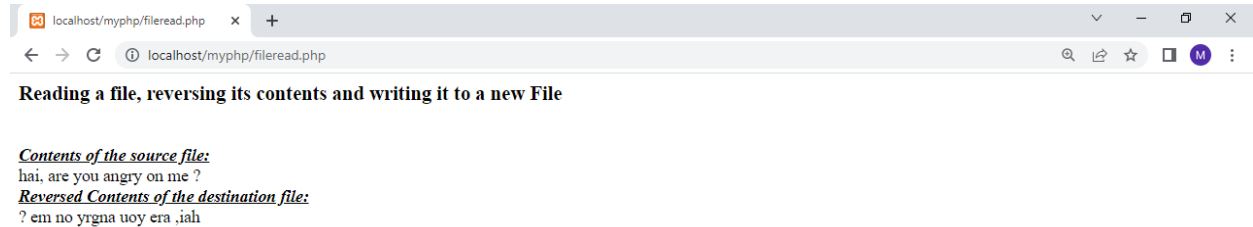
```

<?php
echo "<h3>Reading a file, reversing its contents and writing it to a new File</h3>";
$inputfile=fopen("angry.txt","r")or die("Unable to open file for reading");
$originaldata=file_get_contents("angry.txt");
echo "<br><b><i><u>Contents of the source file:</b></i></u><br>".$originaldata;
$reverseddata=strrev($originaldata);
$newfile=fopen("destination.txt","w") or die("Unable to open file for writing");
file_put_contents("destination.txt",$reverseddata);
$data =file_get_contents("destination.txt");
echo "<br><b><i><u>Reversed Contents of the destination
file:</b></i></u><br>".$data;
fclose($inputfile);

```

```
fclose($newfile);  
?>
```

Output:



10. Creating a Student Marksheet and inserting records in Database

Source Code:

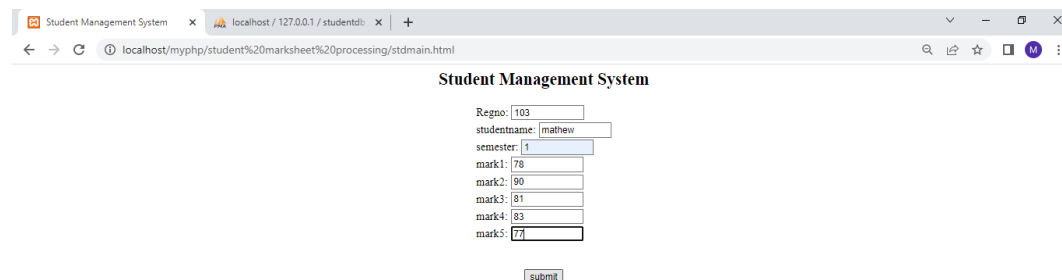
stdmain.html

```
<html>  
  <head>  
    <title> Student Management System </title>  
    <center> <h2> Student Management System </h2></center>  
    <form name="std" method="post" action="stdprocess.php">  
      <table border="0" align="center">  
        <tr>  
          <td> Regno: <input type="text" name="regno" size="10">  
        <tr>  
          <td> studentname: <input type="text" name="studname" size="10">  
        <tr>  
          <td> semester: <input type="text" name="sem" size="10">  
        <tr>  
          <td> mark1: <input type="text" name="mark1" size="10">  
        <tr>  
          <td> mark2: <input type="text" name="mark2" size="10">  
        <tr>  
          <td> mark3: <input type="text" name="mark3" size="10">  
        <tr>  
          <td> mark4: <input type="text" name="mark4" size="10">  
        <tr>  
          <td> mark5: <input type="text" name="mark5" size="10">  
        </table>  
        <br><br>  
        <center>  
          <input type="submit" name="submit" value="submit">  
        </center>  
      </form>  
</html>
```

stdprocess.php

```
<?php
    $regno = $_POST['regno'];
    $studentname = $_POST['studname'];
    $semester = $_POST['sem'];
    $mark1 = $_POST['mark1'];
    $mark2 = $_POST['mark2'];
    $mark3 = $_POST['mark3'];
    $mark4 = $_POST['mark4'];
    $mark5 = $_POST['mark5'];
    //database connection
    $conn = new mysqli("localhost","root","","studentdb");
    if($conn->connect_error)
        die('connection failed : '.$conn->connect_error);
    else
    {
        $stmt = $conn->prepare("insert into
sss(regno,studname,sem,mark1,mark2,mark3,mark4,mark5) values(?,?,?,?,?,?,?,?)");
    $stmt-
>bind_param("isiiiiii",$regno,$studentname,$semester,$mark1,$mark2,$mark3,$mark4,$mark5);
        $stmt->execute();
        echo "successfull";
        $stmt->close();
        $conn->close();
    }
?>
```

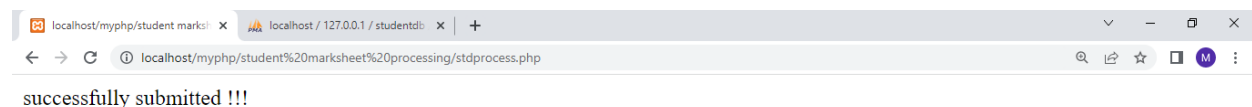
Output:



Student Management System

Regno: 103
studentname: mathew
semester: 1
mark1: 78
mark2: 90
mark3: 81
mark4: 83
mark5: 77

submit



successfully submitted !!!

SELECT * FROM `sss`							
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]							
<input type="checkbox"/> Show all Number of rows: 25 Filter rows: <input type="text" value="Search this table"/>							
+ Options							
regno	studname	sem	mark1	mark2	mark3	mark4	mark5
101	suhail	1	65	56	85	85	86
102	naveen	1	56	85	96	52	41
103	mathew	1	78	90	81	83	77

11. Retrieve and print all the email addresses from the Document using XML

Source Code:

email.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<company>
  <employee>
    <department> Computer Science </department>
    <empname> Mohamed Suhail </empname>
    <empid> AGASC075 </empid>
    <email> mohamed@alphagroup.edu </email>
  </employee>
  <employee>
    <department> Computer Science </department>
    <empname> Naveen Kumar </empname>
    <empid> AGASC018 </empid>
    <email> naveenkumar@alphagroup.edu </email>
  </employee>
  <employee>
    <department> Computer Science </department>
    <empname> Mathew Isaac </empname>
    <empid> AGASC074 </empid>
    <email> mathewisaac@alphagroup.edu </email>
  </employee>
  <employee>
    <department> Computer Science </department>
    <empname> Tony Santhosh </empname>
    <empid> AGASC023 </empid>
    <email> tony@alphagroup.edu </email>
  </employee>
</company>
```

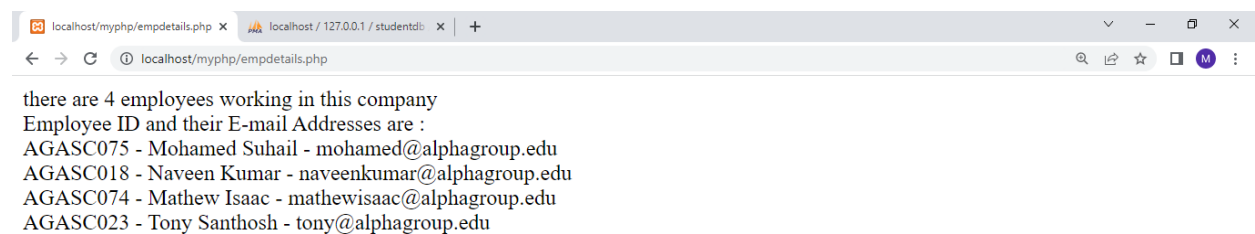
empdetails.php

```

<?php
    $xml = simplexml_load_file("email.xml")
    or die("unable to load XML");
    echo "there are ".count($xml->employee)." employees working
    in this company";
    echo "<br> Employee ID and their E-mail Addresses are :";
    foreach($xml->employee as $semp)
    {
        echo "<br>".$semp->empid;
        echo " - ".$semp->empname;
        echo " - ".$semp->email;
    }
?>

```

Output:



12. Different ways to retrieve text value using DOM

Source Code:

tree.xml

```

<?xml version="1.0"?>
<tree>
<person type="grandpa"/>
<person type="grandma"/>
<children>
<person type="pa"/>
<person type="ma"/>
<children>
<person type="bro">
<name>John</name>
</person>
<person type="sis">
<name>Jane</name>
</person>
</children>
</children>

```

</tree>

domdoc.php

```
<?php
    $doc = new DOMDocument();
    $doc->preserveWhiteSpace = false;
    $doc->load('tree.xml');
    echo "first method=".$doc->firstChild->childNodes->item(2)->childNodes->
item(2)->childNodes->item(0)->childNodes->item(0)->childNodes->item(0)->nodeValue;
    echo "<br>";
    echo "second method=".$doc->getElementsByTagName('name')->item(0)-
>nodeValue;
    echo "<br>";
    echo "third method=".$doc->getElementsByTagName('person')->item(4)-
>childNodes->item(0)->nodeValue;
    echo "<br>";
?>
```

Output:

