

EXPERIMENT-1

1.1 OBJECTIVE:

To accept a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.

1.2 PROGRAM LOGIC:

1. Create a HTML file.
2. Read a number in one text field and display that number name in another text field. 3. Include the JavaScript to convert number into words.

1.3 PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and Mysql.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL
localhost:3306/directory name/filename

1.4 SOURCE CODE:

```
<html>
<head>
<script type="text/javascript">
var th = ['','thousand','million', 'billion','trillion'];
var dg=['zero','one','two','three','four','five','six','seven','eight','nine'];
var
tn=['ten','eleven','twelve','thirteen','fourteen','fifteen','sixteen','seventeen','eighteen','nineteen'];
var tw=['twenty','thirty','forty','fifty','sixty','seventy','eighty','ninety'];

function change()
{
    var numString=document.getElementById('anumber').value;
    var output=towords(numString);
    document.getElementById('aresult').value=output;
}

function towords(s)
{
    s = s.toString();
    if (s != parseFloat(s))
        return 'not a number';
    var x = s.indexOf('.');
    if (x == -1)
        x = s.length;
    if (x > 3)
        return 'too big';
    var n = s.split("");
    var str = "";
    var sk = 0;
    for (var i=0; i < x; i++)
    {
```

```

        if ((x-i)%3==2)
        {
            if (n[i] == '1')
            {
                str += tn[Number(n[i+1])] + ' '; i++;
                sk=1;
            }
            else if (n[i]!=0)
            {
                str += tw[n[i]-2] + ' '; sk=1;
            }
        }
        else if (n[i]!=0)
        {
            str += dg[n[i]] + ' ';
            if ((x-i)%3==0)
                str += 'hundred ';
            sk=1;
        }
        if ((x-i)%3==1)
        {
            if (sk)
                str += th[(x-i-1)/3] + ' ';
            sk=0;
        }
    }
    return str;
}
</script>
</head>
<body>
<form>
    enter a number<input type="text" id="anumber">
    <input type="button" value='convert to words' onClick="change()">
    <input type="text" size="40" id="aresult">
</form>
</body>
</html>

```

EXPERIMENT-2

OBJECTIVE:

To display the number lines ,words and characters from an HTML page that has one input, which can take multi-line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with a white space and lines are separated with new line character.

PROGRAM LOGIC:

1. Create an HTML File to read the multi-line text using text area field.

2. Once the user clicks the submit button it should show the number of characters ,words and lines entered in the text area.
3. Include the JavaScript code to count the number of characters, words and lines.

PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and Mysql.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL
localhost:3306/directory name/filename

SOURCE CODE:

```
<html>
<head>
    <script type="text/JavaScript"> function count()
    {
        var str=document.getElementById('atext').value;
        var result="";
        result+='The number of characters are '+str.length+'\n';

        var arr=str.split(' ');
        result+='The number of words are '+arr.length+'\n';

        var a=str.split('\n');
        result+='The number of lines are '+a.length; alert(result);
    }
    </script>
</head>

<body>
<form>
    <textarea rows='7' cols='70' id='atext'> </textarea>
    <input type="submit" value="submit" onclick="count()">
</form>
</body>

</html>
```

EXPERIMENT-3

OBJECTIVE:

To print the capital of country from HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size)

PROGRAM LOGIC:

1. Create an HTML file from which select the country from the selection box.
2. Once the user selects the country it should display the selected country's capital.
3. Include the JavaScript display the capital for selected country.
4. Create the CSS file which includes the properties like color, bold and font size.

PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and Mysql.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL
localhost:3306/directory name/filename

SOURCE CODE:

```
<html>
<title>Contries</title> <head>
<script type="text/JavaScript">
function OnDropDownChange(dropDown)
{
var selectedValue = dropDown.options[dropDown.selectedIndex].value;
document.getElementById("txtSelectedCapital").innerHTML = selectedValue;
}
</script>
</head>

<body>
<form action = "">
<select name = "Countries" onChange="OnDropDownChange(this);">
<option value="">--Select a country--</option>
<option value="New Delhi">India</option>
<option value="Wellington">New Zealand</option>
<option value="Paris">France</option>
<option value="Athens">Greece</option>
<option value="Madrid">Spain</option>
</select>

<h1 style="color:green;font-family:Times New Roman;font-size:200%;" id="txtSelectedCapital"
type="text"></h1>
</form>
</body>
</html>
```

EXPERIMENT-4

OBJECTIVE:

To write a program that parses an XML document using DOM and SAX parsers.

PROGRAM LOGIC:

1. Create a .xml file that has to be parsed.
2. Using DOM parser
 - a. Get a document builder using document builder factory and parse the xml file to create a DOM object
 - b. Get a list of User elements from the DOM
 - c. For each User element id get the name, age and qualification.
3. Using SAX Parser
 - a. Create a Sax parser and parse the xml

- b. In the event handler create the User object
- c. Print out the data

PROCEDURE:

To execute a java program we require setting a class path: 1.C:\set path= C:\Program Files\Java\jdk1.6.0\bin;. ; 2.C:\javac Parse.java
C:\java Parse

SOURCE CODE:

DOM:

Student.xml

```
<?xml version="1.0"?>
<student>
  <Roll_No>10</Roll_No>
  <Personal_Info>
    <Name>parth</Name>
    <Address>pune</Address>
    <Phone>1234567890</Phone>
  </Personal_Info>
  <Class>Second</Class>
  <Subject>Maths</Subject>
  <Marks>100</Marks>

  <Roll_No>20</Roll_No>
  <Personal_Info>
    <Name>AnuRadha</Name>
    <Address>Bangalore</Address>
    <Phone>90901233</Phone>
  </Personal_Info>
  <Class>Fifth</Class>
  <Subject>English</Subject>
  <Marks>90</Marks>

  <Roll_No>30</Roll_No>
  <Personal_Info>
    <Name>Anand</Name>
    <Address>Mumbai</Address>
    <Phone>90901256</Phone>
  </Personal_Info>
  <Class>Fifth</Class>
  <Subject>English</Subject>
  <Marks>90</Marks>
</student>
```

Parse.java

```
import java.io.*;
import javax.xml.parsers.*;
import org.w3c.dom.*;
import org.xml.sax.*;

public class Parse
```

```

{
    public static void main(String[] arg)
    {
        try
        {
            System.out.println("enter the name of xml document");
            BufferedReader input=new BufferedReader(new InputStreamReader(System.in));
            String file_name=input.readLine();
            File fp=new File(file_name);
            if(fp.exists())
            {
                try
                {
                    DocumentBuilderFactory Factory_obj=DocumentBuilderFactory.newInstance();
                    DocumentBuilder builder=Factory_obj.newDocumentBuilder();
                    InputSource ip_src=new InputSource(file_name);
                    Document doc=builder.parse(ip_src);
                    System.out.println(file_name+" is well-formed!");
                }
                catch(Exception e)
                {
                    System.out.println(file_name+" isn't well-formed!"); System.exit(1);
                }
            }
            else
            {
                System.out.print("file not found!");
            }
        }
        catch(IOException ex)
        {
            ex.printStackTrace();
        }
    }
}

```

SAX:

EmployeeDetail.xml:

```

<?xml version="1.0"?>
<EmployeeDetail>
    <Employee>
        <Emp_id>E-001</Emp_id>
        <Emp_Name>revathy</Emp_Name>
        <Emp_E-mail>revathy@yahoo.com</Emp_E-mail>
    </Employee>
    <Employee>
        <Emp_id>E-002</Emp_id>
        <Emp_Name>vinod</Emp_Name>
        <Emp_E-mail>vinod2@yahoo.com</Emp_E-mail>
    </Employee>

```

```
<Employee>
  <Emp_id>E-001</Emp_id>
  <Emp_Name>deepak</Emp_Name>
  <Emp_E-mail>deepak3@yahoo.com</Emp_E-mail>
</Employee>
</EmployeeDetail>
```

SAXParserCheck.java:

```
import org.xml.sax.*;
import org.xml.sax.helpers.*;
import java.io.*;

public class SAXParserCheck
{
    public static void main(String[] args) throws IOException
    {
        BufferedReader bf=new BufferedReader(new InputStreamReader(System.in));
        System.out.print("enter XML file name :");
        String xmlfile=bf.readLine();
        SAXParserCheck par=new SAXParserCheck(xmlfile);
    }

    public SAXParserCheck(String str)
    {
        try
        {
            File file=new File(str); if(file.exists())
            {
                XMLReader reader=XMLReaderFactory.createXMLReader();
                reader.parse(str);
                System.out.println(str+" is well-formed!");
            }
            else
            {
                System.out.println("File not found:"+str);
            }
        }
        catch(SAXException sax)
        {
            System.out.println(str+" isn't well-formed");
        }
        catch(IOException io)
        {
            System.out.println(str+" isn't well-formed");
        }
    }
}
```

EXPERIMENT-5

OBJECTIVE:

To validate the user login using PHP and database.

PROGRAM LOGIC:

1. Create the user database and table along with the user name and password as the attributes.
2. Create an HTML File to retrieve user name and password.
3. Create PHP file to retrieve the data from the database and HTML File.
4. Authenticate the user name and password if it is the valid user the display as successful login otherwise display failure message.

PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and MySQL.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL
localhost:3306/directory name/filename

SOURCE CODE:

Login.html

```
<html>
<body>
<form enctype="multipart/form-data" action="database.php" method="post">
username:<input type="text" name="username"> <br>
password:<input type="password" name="password" maxlength="10"> <br>
<input type="submit" name="submit">
</form>
</body>
</html>
```

database.php

```
<?php
$name=$_REQUEST['username'];
$pass=$_REQUEST['password'];
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = 'root';
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
{
    die('Could not connect: ' . mysql_error());
}

$sql = 'SELECT username, password FROM user_detail';
mysql_select_db('user_login');

$retval = mysql_query( $sql, $conn );
```



```

if(! $retval )
{
    die('Could not get data: ' . mysql_error());
}

while($row = mysql_fetch_array($retval, MYSQL_ASSOC))
{
    if($name==$row['username'] && $pass==$row['password'])
    {
        echo("login suceses");
        return true;
    }
    else
    {
        echo("invalid username and paasword ");
        return false;
    }
}
?>

```

EXPERIMENT-6

OBJECTIVE:

To validate the user login using PHP and XML.

PROGRAM LOGIC:

1. Create the XML file with the tags user name and password.
2. Create an HTML File to retrieve user name and password.
3. Create PHP file to retrieve the data from the XML and HTML File.
4. Authenticate the user name and password if it is the valid user the display as successful login otherwise display failure message.

PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and Mysql.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL
localhost:3306/directory name/filename

SOURCE CODE:

Login.html

```

<html>
<body>
<form action="database.php" method="post">
username:<input type="text" name="username"> <br>
password:<input type="password" name="password" maxlength="10"> <br>
<input type="submit" name="submit"> </form>
</body>

```

```
</html>
```

Database.php

```
<?php
$name=$_REQUEST['username'];
$pass=$_REQUEST['password'];
$file="file.xml";
$xml=simplexml_load_file($file);

if($name==$xml->username && $pass==$xml->password)
{
    echo("login suceses");
    return true;
}
Else
{
    echo("invalid username and paasword ");
    return false;
}
?>
```

File.xml

```
<?xml version="1.0"?>
<user-detail>
    <username>raghu</username>
    <password>siet</password>
</user-detail>
```

EXPERIMENT-7

OBJECTIVE:

To develop simple calculator web application.

PROGRAM LOGIC:

1. Create an HTML file to read two variables and operator.
2. Create an PHP file to compute the arithmetic operation.
3. Dispaly the result.

PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and Mysql.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL localhost:90/directory name/filename

SOURCE CODE:

cal.html

```
<html>
<body>
    <form method="POST" action="process.php">
```

```
NO1:<input type="text" name="num1" id="num1"> <br>
NO2:<input type="text" name="num2" id="num2"> <br>
```

```
<select name = "func">
  <option value="+">Add [+]</option>
  <option value="-">Subtract[-]</option>
  <option value="*">Multiple[*]</option>
  <option value="/">Divide[/]</option>
  <option value="%">modulus</option>
</select>
```

```
<input type="submit" name="submit" value="Submit" >
```

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

Process.php

```
<?php
$num1 = ($_POST['num1']);
$num2 = ($_POST['num2']);
$func = ($_POST['func']);
if(is_numeric($num1) &&is_numeric($num2) )
{
    if($func != null)
    {
        switch($func)
        {
            case "+" :
                $result= $num1 + $num2;
                break;
            case "-" :
                $result= $num1 - $num2;
                break;
            case "*" :
                $result= $num1 * $num2;
                break;
            case "/" :
                $result= $num1 / $num2;
                break;
            case "%" :
                $result= $num1 % $num2;
                break;
        }
        echo("calculation result: ". $result);
    }
}
?>
```

EXPERIMENT-8

OBJECTIVE:

To validate age attribute, if age is less than 18 then display a message the user is not authorized to visit this site otherwise a welcome message should be displayed using PHP.

PROGRAM LOGIC:

1. Create an HTML file to read the age and username.
2. Create a PHP file to check the user age.
3. If age>18 then display a message, welcome to the site otherwise display not authorized to this site.

PROCEDURE:

To execute a PHP program:

1. Open XAMPP control Panel then start Apache Server and MySQL.
2. Open Notepad++ and Save the php program in htdocs folder of XAMPP.
3. To run the php file open the browser and type the following URL
localhost:3306/directory name/filename

SOURCE CODE:

Login.html

```
<html>
<body>
<form action="age.php" method="post">
username:<input type="text" name="username"> <br>
password:<input type="number" name="age" maxlength="10"> <br>
<input type="submit" name="submit">
</form>
</body>
</html>
```

Age.php

```
<?php
    $name=$_REQUEST['username'];
    $age=$_REQUEST['age'];
    if($age<=18)
    {
        echo("hello!".$name.".not authorized to visit this site");
    }
    else
    {
        echo("hello!".$name.".welcome to this site");
    }
?>
```