

Internet and Web Programming
CSE3002
Fall Semester 2020-21

Lab Assignment 4

ISHAAN OHRI
18BCE0265

Question:

Write a PHP program to demonstrate the concept of Session Management and Cookies

1. Create a cookie with a value of username and his email. The cookie should expire in 3 days. Check whether the cookie is present in the website, when the user enters his username, the email should be given by the cookies.
2. Perform the following operations:
 - i. Create a cookie. Check whether the cookie is enabled or not. Print the status.
 - ii. Delete the created cookie before an hour
 - iii. Check whether the cookie is disabled or not. Print the status

Procedure:

HTML:

<!DOCTYPE html>: Tells browser that it is an HTML document

<head>: Container for header elements

<title>: Name for toolbar

<body>: Body element for document

<style>: Styles for the different elements

<p>: Defines a paragraph

<form>: Used to define a form in the HTML page

action: defines the script to be executed when submit button is clicked.

method: defines the method

<input>: Declare input controls that allow users to input data.

type: Defines the type of input text/radio/checkbox

name: Defines the name of the <input> element

id: Unique id to identify

pattern: Defines pattern of input data

required: Compulsory to enter

value: Defines the value of the option in radio/checkbox

PHP:

isset(): Checks whether a variable is set. It means that it has to be declared and is not NULL. Here it is used to check if any POST request is made by the user.

Username and email are retrieved from the data using \$_POST[] method.

Setcookie() method is used to set the cookie for 3 days = 3 * 24 * 60 * 60

Cookie can be automatically deleted after 1 hour by setting time in setcookie function as 60 * 60.

Code:

index.php

```
<html>
<body>
<form action="page.php" method="post">
<p> Username: <input name="username" type="text" value="
<?php
if(isset($_COOKIE["username"])) {
echo $_COOKIE["username"]; } ?>" class="input-field"></p>
<p>Email: <input name="email" type="email" value="
<?php if(isset($_COOKIE["email"])) {
echo $_COOKIE["email"]; } ?>" class="input-field">
</p>
<p><input type="checkbox" name="remember" /> Remember me </p>
<p><input type="submit" value="Login"></span></p>
</form>
<p><a href="page.php"> Go to Login Page </a></p>
</body>
```

```
</html>
```

page.php

```
<?php if(!empty($_POST["remember"])) {  
    setcookie ("username",$_POST["username"],time()+ 86400*3);  
    setcookie ("email",$_POST["email"],time()+ 86400*3);  
    echo "Cookies Set Successfully"; }  
else  
{  
    setcookie("username","");  
    setcookie("email","");  
    echo "Cookies Not Set"; }  
?>
```

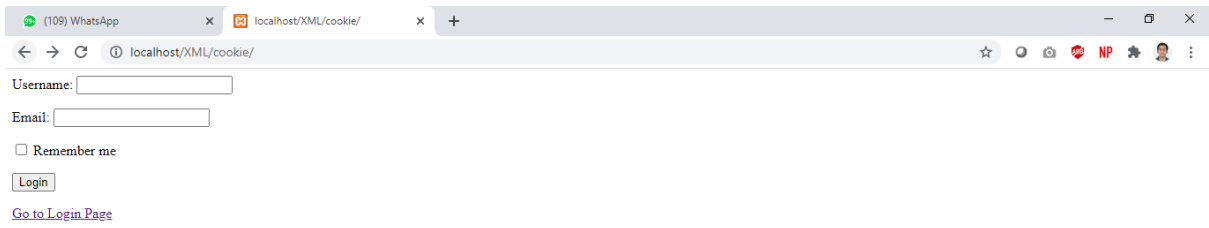
delete.php

```
<!DOCTYPE html>  
<?php  
setcookie("user", "", time() - 3600);  
?>  
<html>  
<body>  
<?php  
echo "Cookie 'user' is deleted.";  
?>  
</body>  
</html>
```

cookie.php

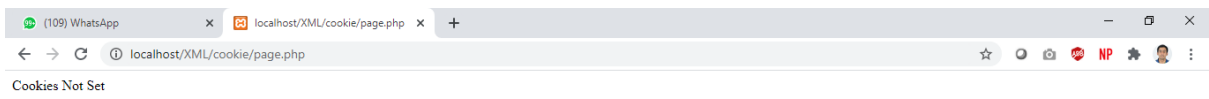
```
<?php  
setcookie("test_cookie", "test", time() + 3600, '/');  
?>  
<html>  
<body>  
<?php  
if(count($_COOKIE) > 0) {  
    echo "Cookies are enabled.";  
} else {  
    echo "Cookies are disabled.";  
}  
?>  
</body>  
</html>
```

Output:



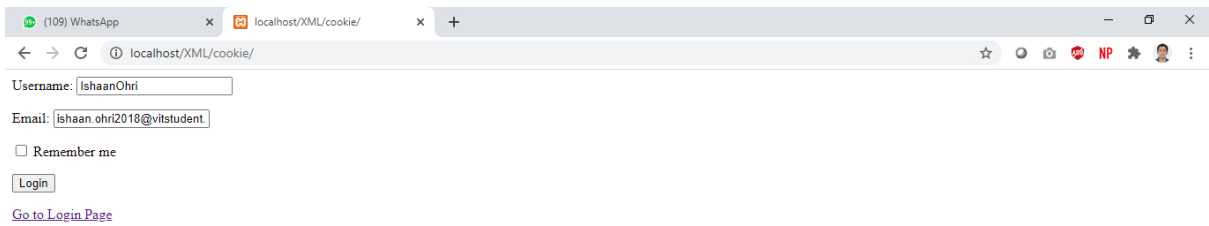
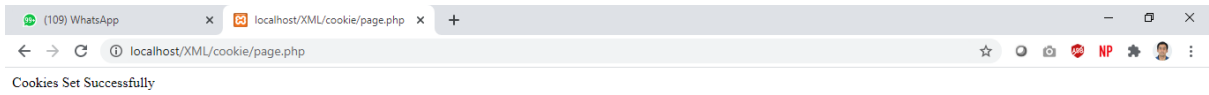
A screenshot of a web browser window with two tabs: "(109) WhatsApp" and "localhost/XML/cookie/". The address bar shows "localhost/XML/cookie/". The page content includes a login form with the following elements:

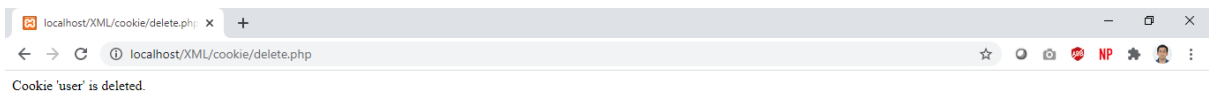
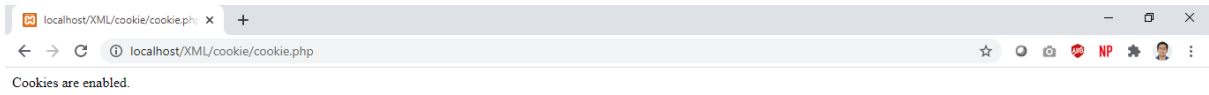
- A label "Username:" followed by a text input field.
- A label "Email:" followed by a text input field.
- A checkbox labeled "Remember me".
- A "Login" button.
- A link labeled "Go to Login Page".



A screenshot of a web browser window with two tabs: "(109) WhatsApp" and "localhost/XML/cookie/page.php". The address bar shows "localhost/XML/cookie/page.php". The page content displays a message:

Cookies Not Set





Question:

Write a program to demonstrate the concept of data storage and parsing in XML

Develop a thesaurus tool by creating a schema for thesaurus. When a word is entered the synonyms or antonyms must be displayed based on the user request.

Procedure:

<thesauras> is the root node

<word> enclosed all the words in the "thesauras"

<synonym> encloses all the query words synonyms

xsl:for-each to loop through similar tags

xsl:value-of to print the value enclosed within the "selected" tag

xsl:text specifies text data and xsl:number specifies numerical data

sum adds the values enclosed within the selected tag

format is used to specify the format in which a numerical value is to be displayed

xsl:choose is used for decision making

xsl:when is equivalent to if statement

xsl:otherwise is equivalent to else statement

Code:

thesaurus.xsl

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html"/>
<xsl:template match="/">
<html>
<head>
<title> thesaurus.xsl</title>
</head>
<body>
<form method="post" action="">
<table>
<tr>
<td>Enter word:</td>
<td><input type="text" id="search"/></td>
```

```

<td><input type="submit" id="submit" value="Submit"/></td>
</tr>
<xsl:for-each select="thesaurus/word">
<xsl:if test="@content='Smart'">
<tr>
<td>
<xsl:value-of select="synonyms"/>
</td>
</tr>
</xsl:if>
</xsl:for-each>
</table>
</form>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

thesaurus.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="thesaurus.xsl"?>
<thesaurus>

<word content="Beautiful">

<synonyms>attractive, pretty, charming, pleasing, alluring</synonyms> </word>

<word content="Dumb">

<synonyms>stupid, dumbo</synonyms>

</word>

</thesaurus>

```

thesaurus.dtd

```

</xsl:template>

</xsl:stylesheet>

<?xml version='1.0' encoding='UTF-8'?>

<!ELEMENT thesaurus (word)*>

<!ELEMENT word (synonyms)*>

<!ATTLIST word

```

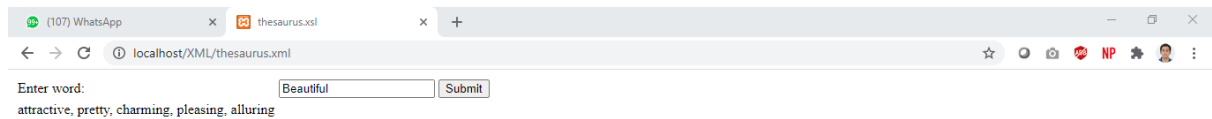


```
content CDATA #IMPLIED  
  
>  
  
<!ELEMENT synonyms (#PCDATA)>
```

Output:



A screenshot of a web browser window. The address bar shows 'localhost/XML/thesaurus.xml'. The page contains a form with the label 'Enter word:' followed by an empty text input field and a 'Submit' button.



A screenshot of the same web browser window after a submission. The text input field now contains the word 'Beautiful'. Below the input field, the synonyms 'attractive, pretty, charming, pleasing, alluring' are displayed. The 'Submit' button remains visible.

Question:

XSLT – Create a student mark maintenance system using XML. Create a webpage to display all the students consolidated mark statement with pass (green color) or fail (red color) using XSLT.

Procedure:

<sr> is the root node

<student> tag enclosed the details for one student

<n> contains the student name

<m> contains the mark for the subject

<res> contains the result

xsl:for-each to loop through similar tags

xsl:value-of to print the value enclosed within the "selected" tag

xsl:text specifies text data and xsl:number specifies numerical data

sum adds the values enclosed within the selected tag

format is used to specify the format in which a numerical value is to be displayed

xsl:choose is used for decision making

xsl:when is equivalent to if statement

xsl:otherwise is equivalent to else statement

Code:

sr.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="sr.xsl"?>
<sr>
<student>
<n>Ishaan Ohri</n>
<m>100</m>
<res>PASS-Distinction</res>
</student>
<student>
<n>Shreya Basu</n>
<m>70</m>
```

```

<res>PASS-Very Good</res>
</student>
<student>
<n>Siddhant Sharda</n>
<m>01</m>
<res>FAIL-Better luck next time</res>
</student>
<student>
<n>Gurprasad Singh</n>
<m>88</m>
<res>PASS-Very good ,keep it up</res>
</student>
<student>
<n>Shivam Anand</n>
<m>33</m>
<res>FAIL-Better luck next time</res>
</student>
<student>
<n>Sameer Rupani</n>
<m>80</m>
<res>PASS-Very good ,keep it up</res>
</student>
<student>
<n>Rupin Singh</n>
<m>78</m>
<res>PASS-Very Good</res>
</student>
<student>
<n>Riya</n>
<m>50</m>
<res>PASS- Can do better</res>
</student>
</sr>

```

sr.xsl

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<h1 align="center"> STUDENTS' RESULT RECORD</h1>
<table border="4" align="center" width="500px" height="400px">
<tr>
<th>Name</th>
<th>Marks</th>
<th>Result</th>
</tr>
<xsl:for-each select="sr/student">

```

```

<xsl:if test = "m > 40">
<tr bgcolor = "green" align="center">
<td><xsl:value-of select="n"/></td>
<td><xsl:value-of select="m"/></td>
<td><xsl:value-of select="res"/></td>
</tr>
</xsl:if>
<xsl:if test = "m <= 40">
<tr bgcolor = "red" align="center">
<td><xsl:value-of select="n"/></td>
<td><xsl:value-of select="m"/></td>
<td><xsl:value-of select="res"/></td>
</tr>
</xsl:if>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

Output:

| Name | Marks | Result |
|-----------------|-------|----------------------------|
| Ishaan Ohri | 100 | PASS-Distinction |
| Shreya Basu | 70 | PASS-Very Good |
| Siddhant Sharda | 01 | FAIL-Better luck next time |
| Gurprasad Singh | 88 | PASS-Very good ,keep it up |
| Shivam Anand | 33 | FAIL-Better luck next time |
| Sameer Rupani | 80 | PASS-Very good ,keep it up |
| Rupin Singh | 78 | PASS-Very Good |
| Riya | 50 | PASS- Can do better |