MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R Patna, Mandya -571438.

WEB PROGRAMMING LABORATORY [10CSL78]

7th SEMESTER

Academic Year: 2016-17 (ODD Semester)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Maharaja Institute of Technology Mysore

Vision

" To be recognized as a premier technical and management institution promoting extensive education fostering research, innovation and entrepreneurial attitude"

Mission

1. To empower students with indispensable knowledge through dedicated teaching and collaborative learning.
2. To advance extensive research in science, engineering and management disciplines.
3. To facilitate entrepreneurial skills through effective institute-industry collaboration and interaction with alumni.
4. To instill the need to uphold ethics in every aspect.
5. To mould holistic individuals capable of contributing to the advancement of the society.

Department of Computer Science and Engineering

Vision

"To be a leading academic department offering computer science and engineering education, fulfilling industrial and societal needs effectively."

Mission

M1 : To enrich the technical knowledge of students in diversified areas of Computer Science and Engineering by adopting outcome based approaches**.**

M2 : To empower students to be competent professionals maintaining ethicality.

M3 : To facilitate the development of academia-industry collaboration.

M4 : To create awareness of entrepreneurship opportunities.

Program Educational Objectives Statements

PEO1 Be successful in solving engineering problems associated with computer science and engineering domains

PEO2 Work collaboratively on multidisciplinary projects and acquire high levels of professionalism backed by ethics

PEO3 Communicate effectively and exhibit leadership qualities, team spirit necessary for a successful career in either industry, research or entrepreneurship

PEO4 Continue to learn and advance their career through participation in the activities of professional bodies, obtaining professional certification, pursue of higher education

Program Specific Outcome (PSO)

PSO 1: Apply software engineering practices and strategies in diversified areas of computer science for solving problems using open source environment.

PSO 2: Develop suitable algorithms and codes for applications in areas of cognitive technology, computer networks with software engineering principles and practices.

INDEX PAGE

SL. NO.	Content		Pg. No.
1	Do’s and Don’ts in lab		
2	VTU Lab Syllabus		
3	CO‘s with mapping to PO’s and PSO’s		
4	**1 (a)**	Develop and demonstrate a XHTML file that includes Javascript script for the following problems: Input: A number n obtained using prompt Output: The first n Fibonacci numbers	01
	1 (b)	Input: A number n obtained using prompt Output: A table of numbers from 1 to n and their squares using **alert**	02
	2 (a)	Develop and demonstrate, using Javascript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.	04
	2 (b)	Modify the above program to get the current semester also (restricted to be a number from 1 to 8)	06
	3 (a)	Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.	07
	3 (b)	Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.	09
	4 (a)	Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.	10
	4 (b)	Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.	12
	5 (a)	Write a Perl program to display various Server Information like Server Name, Server Software, Server protocol, CGI Revision etc.	13
	5 (b)	Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.	13

	6 (a)	Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.	14
	6 (b)	Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.	14
	7	Write a Perl program to display a digital clock which displays the current time of the server.	15
	8	Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.	15
	9	Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.	16
	10	Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.	17
	11	Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.	17
	12	Build a Rails application to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.	20
5	Viva questions		24

General Lab Guidelines:

- Maintain laboratory etiquettes during the laboratory sessions.
- Do not wander around or distract other students or interfere with the conduction of the experiments of other students.
- Keep the laboratory clean, do not eat, drink or chew gum in the laboratory.

DO’S

- Sign the log book when you enter/leave the laboratory.
- Read the hand out/procedure before starting the experiment. If you do not understand the procedure, clarify with the concerned staff.
- Report any problem in system (if any) to the person in-charge.
- After the lab session, shut down the computers.
- All students in the laboratory should follow the directions given by staff/lab technical staff.

DON’TS

- Do not insert metal objects such as pins, needle or clips into the computer casing. They may cause fire.
- Do not open any irrelevant websites in labs.
- Do not use flash drive on laboratory computers without the consent of lab instructor.
- Do not upload, delete or alter any software/ system files on laboratory computers.
- Students are not allowed to work in laboratory alone or without presence of the teaching staff/ instructor.
- Do not change the system settings and keyboard keys.
- Do not damage any hardware.

Web Programming Laboratory

Subject Code: 10CSL78 **I.A. Marks : 25**
Hours/Week : 03 **Exam Hours: 03**
Total Hours : 42 **Exam Marks: 50**

1	Develop and demonstrate a XHTML file that includes Javascript script for the following problems: a) Input: A number n obtained using prompt Output: The first n Fibonacci numbers
	b) Input: A number n obtained using prompt Output: A table of numbers from 1 to n and their squares using **alert**
2	a) Develop and demonstrate, using Javascript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.
	b) Modify the above program to get the current semester also (restricted to be a number from 1 to 8)
3	a) Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.
	b) Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.
4	a) Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
	b) Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.
5	a) Write a Perl program to display various Server Information like Server Name, Server Software, Server protocol, CGI Revision etc.

	b) Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.
6	a) Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.
	b) Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
7	Write a Perl program to display a digital clock which displays the current time of the server.
8	Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.
9	Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.
10	Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
11	Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.
12	Build a Rails application to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.
Note: In the examination *each* student picks one question from the lot of *all* 12 questions.	

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

BELAWADI, SRIRANGAPATNA Taluk, MANDYA-571438

Department of Computer Science and Engineering

Date: 27/7/2016

Course Outcome

(2016-2017)

Subject: Web Programming Laboratory **Subject code: 10CSL78**

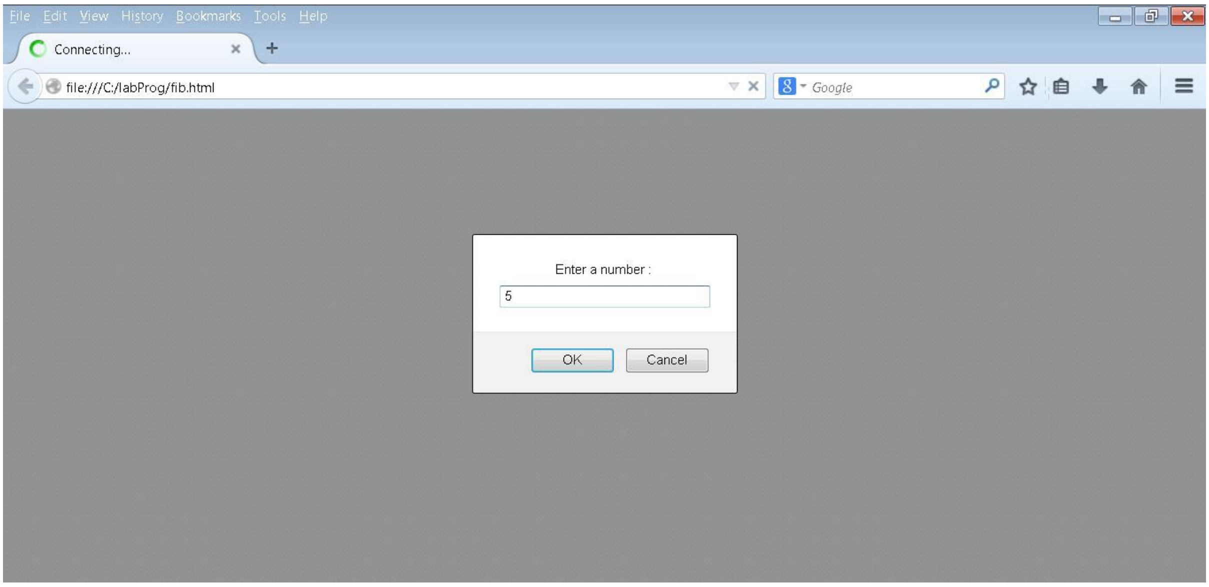
CO's	DESCRIPTION OF THE OUTCOMES
10CSL78.1	Understand the principles of web pages creation and procedural adaption in creating a web page and present them appropriately
10CSL78.2	Apply various web development techniques, scripting languages to serve the purpose of given application
10CSL78.3	Analyze the correctness of the scripts to ensure a qualitative web page and document the same
10CSL78.4	Design the interactive web page with database connectivity

CO No	PO No												PSO	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
1	2	-	-	-	-	-	-	-	-	-	-	2	2	2
2	3	-	-	-	3	-	-	-	-	-	-	-	2	2
3	-	3	-	-	-	-	-	-	-	-	-	-	2	2
4	-	-	3	2	-	-	-	-	-	-	-	-	2	2
CO Average	2.5	3	3	2	3	-	-	-	-	-	-	2	2	2

Program 01(a): Develop and demonstrate a XHTML file that includes Javascript script for the following problems:
Input: A number n obtained using prompt
Output: The first n Fibonacci numbers

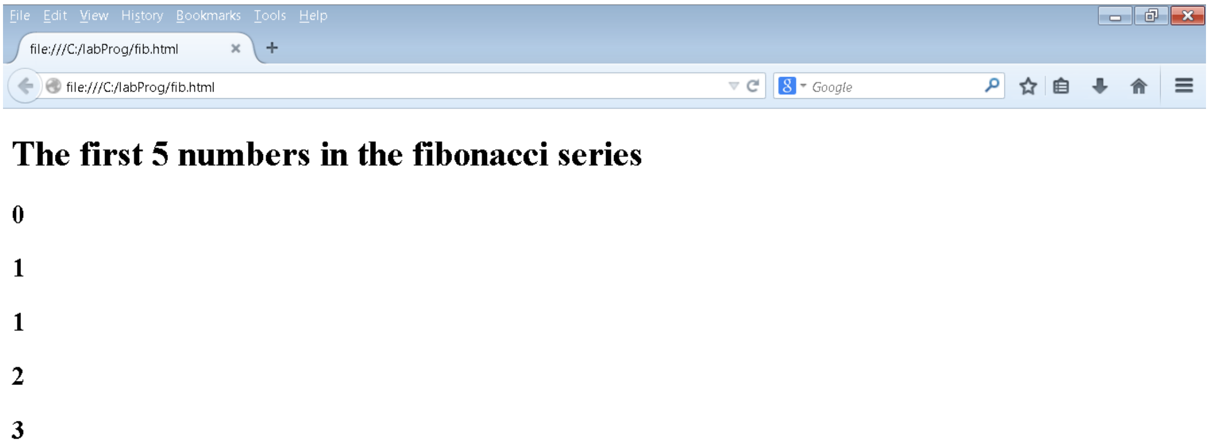
```
<?xml version="1.0" encoding="utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml/">
<head><title>Fibonacci Series</title>
</head>
<body bgcolor="yellow">
<h3 style="text-align:center;color:red"> Program to generate
first n fibonacci numbers </h3>
<script type="text/javascript">
var limit = prompt("Enter the number");
var f1=0;
var f2=1;
document.write("<h3>The limit entered is:
</h3>",limit,"<br/>");
document.write("<h3>The fibonacci series is: </h3> <br/>");
if(limit == 1)
{
document.write("",f1,"<br/>");
}
if(limit == 2)
{
document.write("",f1,"<br/>");
document.write("",f2,"<br/>");
}
if(limit > 2)
{
document.write("",f1,"<br/>");
document.write("",f2,"<br/>");
for(i=2;i<limit;i++)
{
f3 = f2+f1;
document.write("",f3,"<br/>");
f1=f2;
f2=f3;
}
}
</script>
</body>
</html>
```

Sample Output:


File Edit View History Bookmarks Tools Help
Connecting...
file:///C:/labProg/fib.html
Google
OK
Cancel

Enter a number :

5


File Edit View History Bookmarks Tools Help
file:///C:/labProg/fib.html
file:///C:/labProg/fib.html
Google

The first 5 numbers in the fibonacci series

0

1

1

2

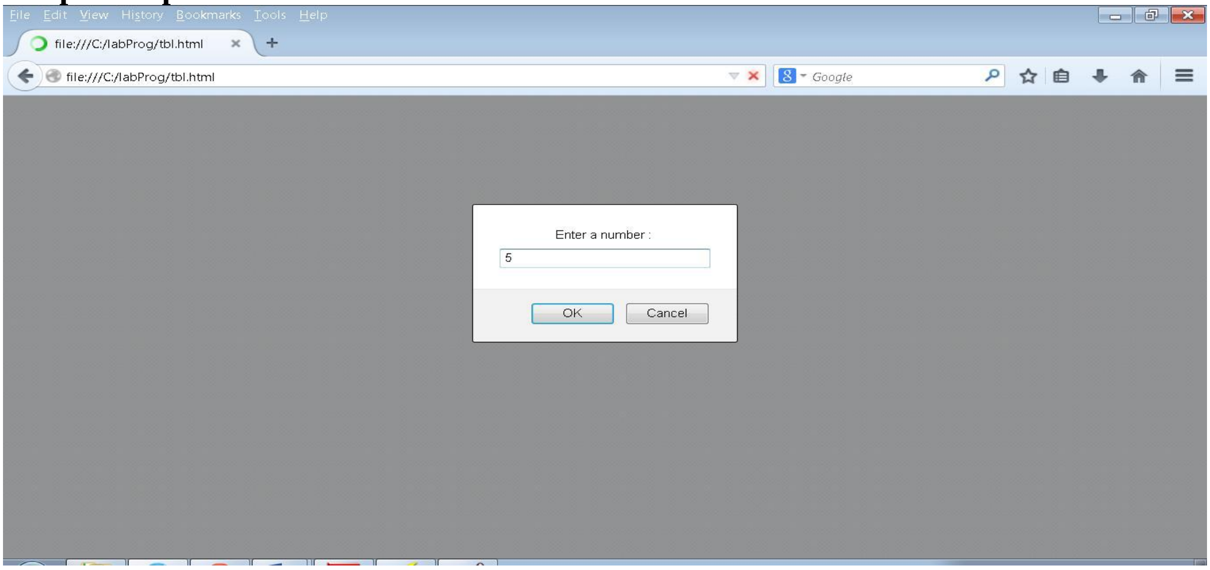
3

Program 1 (b) Input: A number n obtained using prompt
Output: A table of numbers from 1 to n and their squares using alert

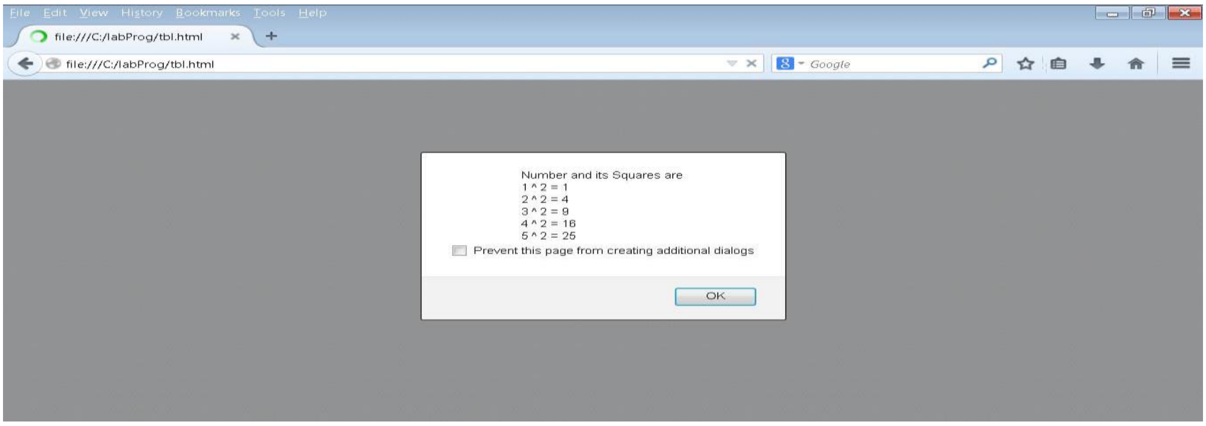
```
<?xml version ="1.0" encoding = "utf-8?>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title> square.html </title>
</head>
<body style="background-color:yellow">
<h3 style="text-align:center;color:red"> Program to generate a
table
of numbers from 1 to n and their squares using alert</h3>
<script type="text/javascript" >
```

```
var n= prompt("Enter the limit 'n' to generate the table of
numbers from 1 to n:","");
var msg = "";
var res = "0";
for(var x = 1; x <= n; x++)
{
res = x * x;
msg = msg + " "+ x + " * "+ x + " = " + res + "\n";
}
alert(msg);
</script>
</body>
</html>
```

Sample Output:


File Edit View History Bookmarks Tools Help
file:///C:/labProg/tbl.html
file:///C:/labProg/tbl.html
Google
OK
Cancel

```
Enter a number :
5
```


File Edit View History Bookmarks Tools Help
file:///C:/labProg/tbl.html
file:///C:/labProg/tbl.html
Google
OK

```
Number and its Squares are
1 ^ 2 = 1
2 ^ 2 = 4
3 ^ 2 = 9
4 ^ 2 = 16
5 ^ 2 = 25
```

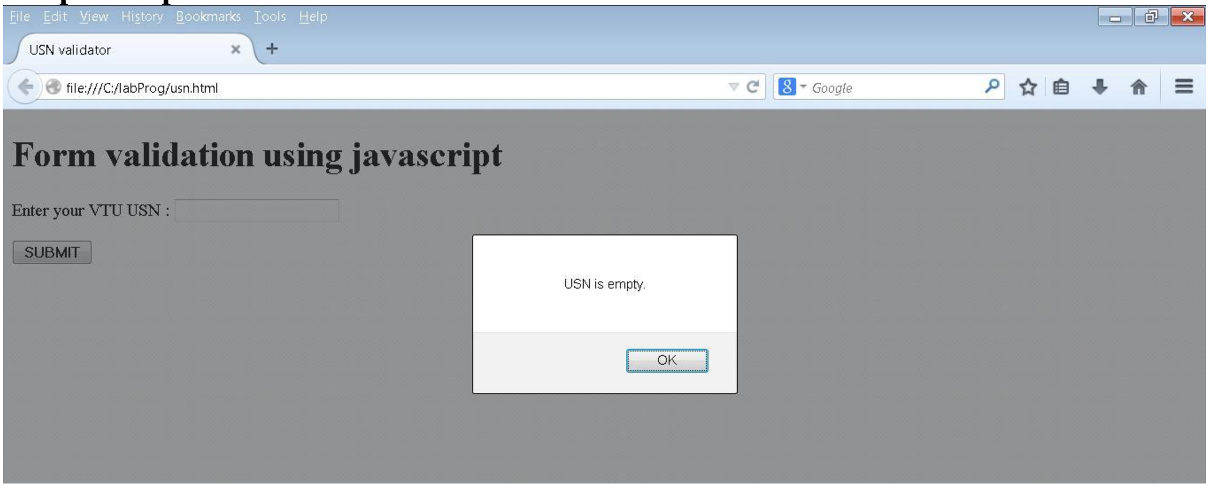
□ Prevent this page from creating additional dialogs

Program 2 (a) : Develop and demonstrate, using Javascript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.

```
<?xml version="1.0" encoding="utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml/">
<head><title>USN.html</title>
<script type="text/javascript">
function checkusn()
{
alert("CIT");
var str=document.getElementById("usn");
alert(str);
var result=str.value.search(/^[1-4]{1}[A-Z]{2}\d{2}[A-
Z]{2}\d{3}$/);
alert(result);
if(result != 0)
{
alert("Entered usn("+str.value+") is not in correct form. The
correct pattern is :ICG09CS100");
}
else
{
alert("Entered usn("+str.value+") is in the correct form.");
}
}
</script>

</head>
<body style="background-color:yellow">
<h3 style="text-align:center;color:red"> Program includes
XHTML document to collect the Student-Information</h3>
<h3 style="color:purple">Student Information </h3>
<form name="my form">
<p style="color:purple"> Enter your USN:</p>
<input type="text" id="usn" size=15/>
<br/>
<br/>
<input type="button" onclick="checkusn()" value="Submit"/>
<input type = "reset" value= "Reset" />
<br/>
</form>
</body>
</html>
```

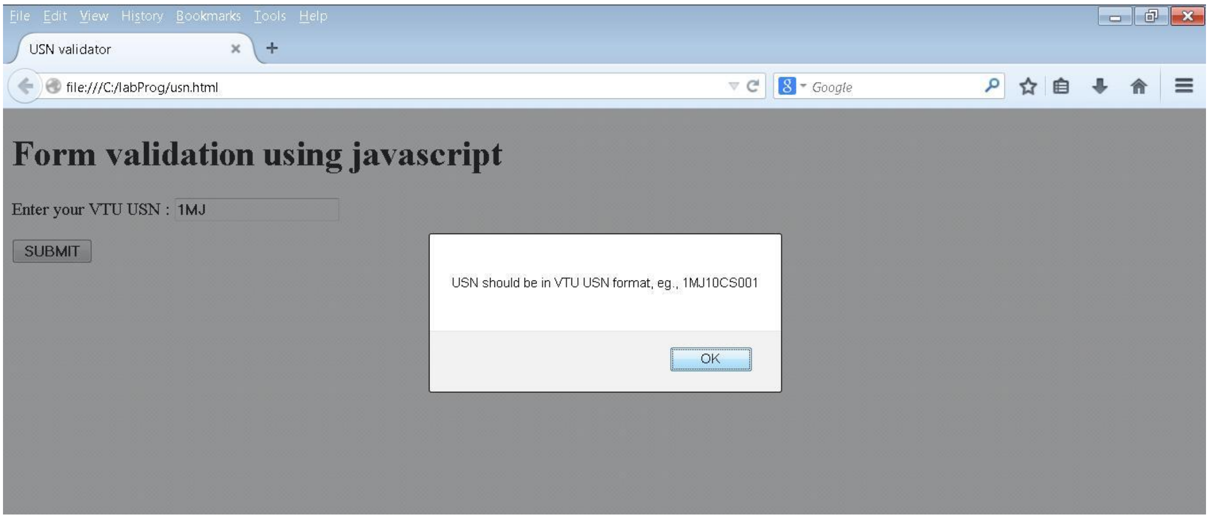
Sample Output:


File Edit View History Bookmarks Tools Help
USN validator
file:///C:/labProg/usn.html
Google
SUBMIT
OK

Form validation using javascript

Enter your VTU USN :

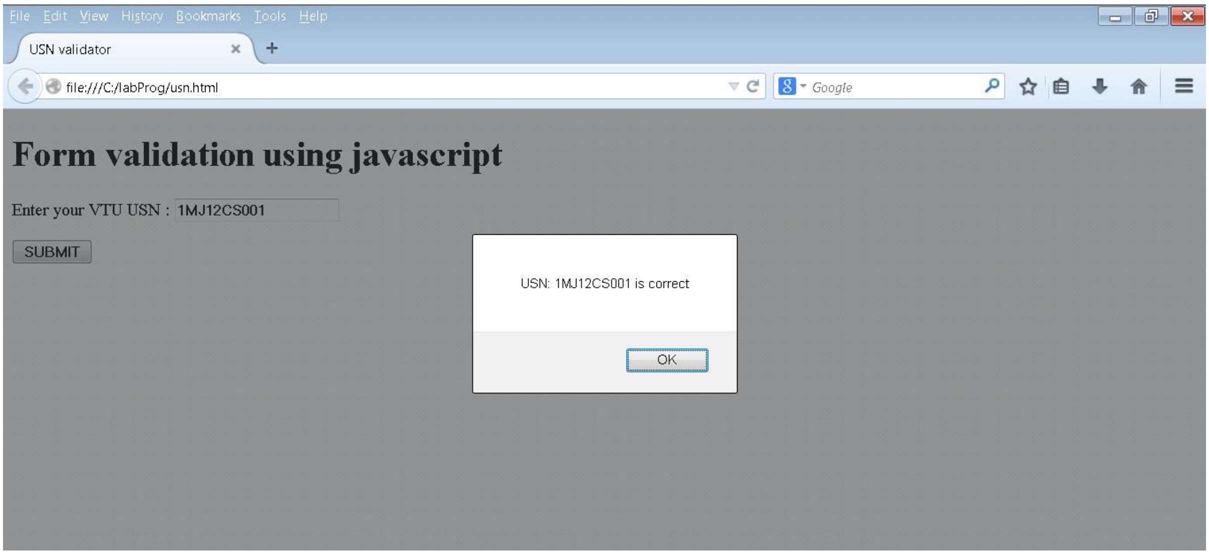
USN is empty.


File Edit View History Bookmarks Tools Help
USN validator
file:///C:/labProg/usn.html
Google
SUBMIT
OK

Form validation using javascript

Enter your VTU USN : 1MJ

USN should be in VTU USN format, eg., 1MJ10CS001


File Edit View History Bookmarks Tools Help
USN validator
file:///C:/labProg/usn.html
Google
SUBMIT
OK

Form validation using javascript

Enter your VTU USN : 1MJ12CS001

USN: 1MJ12CS001 is correct

Program 2 (b): Modify the above program to get the current semester also (restricted to be a number from 1 to 8).

```
<?xml version="1.0" encoding="utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml/">
<head><title>USN.html</title>
<script type="text/javascript">
function validate()
{
var str=document.getElementById("usn");
var str1 = document.getElementById("sem");
var result=str.value.search(/^[1-4]{1}[A-Z]{2}\d{2}[A-
Z]{2}\d{3}$/);
var semres = str1.value.search(/^[1-8]{1}$/);
if(result>=0 && semres>=0)
{
alert("Entered usn("+str.value+") is in correct form and
entered sem("+str1.value+") is in correct form.");
}
else if(result<0 && semres>=0)
{
alert("Entered usn("+str.value+") is not in correct form The
correct form is:1CG09CS100.");
}
else if(result>=0 && semres<0)
{
alert("Entered sem("+str1.value+") is not in correct form The
number is between 1 to 8.");
}
else
{
alert("Entered usn("+str.value+") and entered sem
("+str1.value+") is not in the correct form.");
}
}
</script>
</head>
<body style="background-color:yellow">
<h3 style="text-align:center;color:red"> Program includes
XHTML document to collect the Student-Information</h3>
<h3 style="color:purple">Student Information </h3>
<form name="my form">
<p style="color:purple"> Enter your USN:</p>
<input type="text" id="usn" size=15/>
<br/>
<br/>
<p style="color:purple"> Enter your SEM:</p>
<input type="text" id="sem" size=15/>
<br/>
```

```
<br/>
<input type="button" onclick="validate()" value="Validate"/>
<input type = "reset" value= "Reset" />
<br/>
</form>
</body>
</html>
```

Program 3 (a): Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.

```
<?xml version = "1.0" encoding = "utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>The Stacking order</title>
<style type="text/css">
.layer1Style
{
border: solid thick black;
padding: 1em;
width:300px;
background-color:green;
position:absolute;
top:100px;
left:200px;
z-index:0;
}
.layer2Style
{
border: solid thick red;
padding: 1em;
width:300px;
background-color:BLUE;
position:absolute;
top:120px;
left:220px;
z-index:0;
}
.layer3Style
{
border: solid thick green;
padding: 1em;
width:300px;
background-color:purple;
position:absolute;
top:140px;
left:240px;
z-index:0;
}
</style>
<script type="text/javascript">
var topLayer="layer3";
function mover(toTop)
{
var oldTop=document.getElementById(topLayer).style;
var newTop=document.getElementById(toTop).style;
oldTop.zIndex="0";
newTop.zIndex="10";
topLayer=document.getElementById(toTop).id;
alert(topLayer);
}
</script>
</head>
<body bgcolor = "red">
<h2 style="text-align:center;color:yellow"> Program includes
XHTML
document to show the Stacking of Paragraphs</h2>
<div style="z-index: 10;" class="layer1Style" id="layer1"
onmouseover="mover('layer1')">
The lives of most inhabitants of Industrialized Countries, has
well as some unindustrialized countries, have been changed
forever by the advent of WWW.
</div>
<div style="z-index: 2;" class="layer2Style" id="layer2"
onmouseover="mover('layer2')">
The www may seem like magic , untill you undrestand how it
works.The Web is accessed through a browser.
</div>
<div style="z-index: 0;" class="layer3Style" id="layer3"
onmouseover="mover('layer3')">
Windows XP provides many ways for you to communicate with
friends,
co-workers, and I with the rest of the world.
</div>
</body>
</html>
```

Program 3 (b): Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.

```
<?xml version = "1.0" encoding = "utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>The Stacking order 5b</title>
<style type="text/css">
.layer1Style{
border: solid thick black;
padding: 1em;
width:300px;
background-color:green;
position:absolute;
top:100px;
left:400px;
z-index:1;
}
.layer2Style{
border: solid thick blue;
padding: 1em;
width:300px;
background-color:red;
position:absolute;
top:120px;
left:420px;
z-index:2;
}
.layer3Style{
border: solid thick brown;
padding: 1em;
width:300px;
background-color:orange;
position:absolute;
top:140px;
left:440px;
z-index:3;
}
</style>
<script type="text/javascript">
var topLayer="layer3";
var origPos;
function mover(toTop,pos)
{
var newTop=document.getElementById(toTop).style;
newTop.zIndex="10";
topLayer=document.getElementById(toTop).id;
origPos=pos;
}
```

```
function moveBack()
{
document.getElementById(topLayer).style.zIndex=origPos;
}
</script>
</head>

<body style="background-color:yellow">
<h1 style="text-align:center;color:red"> The Stacking of
paragraphs when
moved from the top stacking position, it returns to its
original
position.</h1>
<div style="z-index: 1;" class="layer1Style" id="layer1"
onmouseover="mover('layer1','1')" onmouseout="moveBack()">
The lives of most inhabitants of Industrailzed Countries, has
well as some
unindustralized countries, have been changed forever by the
advent of WWW.
</div>
<div style="z-index: 2;" class="layer2Style" id="layer2"
onmouseover="mover('layer2','2')" onmouseout="moveBack()">
The www may seem like magic , untill you undrestand how it
works.The Web is
accessed through a browser.
</div>
<div style="z-index: 3;" class="layer3Style" id="layer3"
onmouseover="mover('layer3','3')" onmouseout="moveBack()">
Windows XP provides many ways for you to communicate with
friends, coworkers,
and Iwith the rest of the world.
</div>
</body>
</html>
```

Program 4 (a):Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

```
<?xml version ="1.0" encoding = "utf-8"?>
<!DOCTYPE student[
<!ELEMENT student_information (ad+)>
<!ELEMENT ad (usn,name,collegename,branch,year,email)>
<!ELEMENT usn (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT collegename (#PCDATA)>
<!ELEMENT branch (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT label (#PCDATA|email|year|branch|college|name|usn)*>
<!ELEMENT h3 (#PCDATA)>
<!ELEMENT h2 (#PCDATA)>
]>
<?xml-stylesheet type="text/css" href="stu.css"?>
<student_information>
<h3>Student-Information</h3>
<h2> student 1</h2>
<ad><label> usn:<usn> 4bd06499 </usn></label></ad>
<ad><label> Name:<name> AAA </name></label></ad>
<ad><label> College name:<College> CIT,Gubbi
</College></label></ad>
<ad><label> Branch:<branch> CSE </branch></label></ad>
<ad><label> Year of joining:<year> 2006 </year></label></ad>
<ad><label> Email-ID:<email> aaa@gmail.com
</email></label></ad>
<h2> student 2</h2>
<ad><label> usn:<usn> 4bd06490 </usn></label></ad>
<ad><label> Name:<name> BBB</name></label></ad>
<ad><label> College name:<College> CIT,Gubbi
</College></label></ad>
<ad><label> Branch:<branch> CSE </branch></label></ad>
<ad><label> Year of joining:<year> 2006 </year></label></ad>
<ad><label> Email-ID:<email> bbb@gmail.com
</email></label></ad>
<h2> student 3</h2>
<ad><label> usn:<usn> 4bd06491 </usn></label></ad>
<ad><label> Name:<name> CCC </name></label></ad>
<ad><label> College name:<College> CIT,Gubbi
</College></label></ad>
<ad><label> Branch:<branch> CSE </branch></label></ad>
<ad><label> Year of joining:<year> 2006 </year></label></ad>
<ad><label> Email-ID:<email> ccc@gmail.com
</email></label></ad>
</student_information>
```

Stu.css

```
ad{display:block; margin-top:15px; color:blue; font-size:13pt;
}
usn {color;red; font-size:12pt; margin-left:15px; }
name {color;red; font-size:12pt; margin-left:15px; }
college {color;red; font-size:12pt; margin-left:15px; }
branch {color;red; font-size:12pt; margin-left:15px; }
year {color;red; font-size:12pt; margin-left:15px; }
email {color;red; font-size:12pt; margin-left:15px; }
h3{color;red; font-size:16pt; }
h2 {display:block; color;red; font-size:16pt; }
```

Program 4 (b): Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.

```
<?xml version ="1.0" encoding="utf-8"?>
<!DOCTYPE student[
<!ELEMENT student_information (ad+)>
<!ELEMENT ad (usn,name,collegename,branch,year,email)>
<!ELEMENT usn (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT collegename (#PCDATA)>
<!ELEMENT branch (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT email (#PCDATA)>
]>
<?xml-stylesheet type = "text/xsl" href = "xslstudent.xsl" ?>
<student>
<usn> 4bd06cs099 </usn>
<name> Kumar </name>
<college> CIT </college>
<branch> CSE </branch>
<year> 2006 </year>
<email> abc@gmail.com</email>
</student>
```

Xslstudent.xsl

```
<?xml version ="1.0"?>
<xsl:stylesheet version ="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns="http://www.w3.org/1999/xhtml">
<xsl:template match = "/">
<h2> Student information </h2>
<span style = "font-style: italic;color:red;margin-
right::5pt;"> USN: </span>
<xsl:value-of select = "student/usn" /> <br />
<span style = "font-style: italic;color:red;margin-
right::5pt;">Name: </span>
<xsl:value-of select = "student/name" /> <br />
<span style = "font-style: italic;color:red;margin-
right::5pt;"> Branch: </span>
<xsl:value-of select = "student/branch" /> <br />
<span style = "font-style: italic;color:red;margin-
right::5pt;"> College Name: </span>
<xsl:value-of select = "student/college" /> <br />
<span style = "font-style: italic;color:red;margin-
right::5pt;"> Year of joining: </span>
<xsl:value-of select = "student/year" /> <br />
<span style = "font-style: italic;color:red;margin-
right::5pt;"> Email-ID: </span>
<xsl:value-of select = "student/email" /> <br />
</xsl:template>
```

```
</xsl:stylesheet>
```

Program 5 (a): Write a Perl program to display various Server Information like Server Name, Server Software, Server protocol, CGI Revision etc.

```
#!/usr/bin/perl
print "content-type:text/html \n\n",
"<html>",
"<h1>Welcome to CIT</h1>",
"<h2>Welocome to PWEB LAB</h2>",
"<body>",
"Server Name: $ENV{'SERVER_NAME'}<br>",
"Server Port: $ENV{'SERVER_PORT'}<br>",
"Server Software: $ENV{'SERVER_SOFTWARE'}<br>",
"Server Protocol: $ENV{'SERVER_PROTOCOL'}<br>",
"CGI VERSION: $ENV{'GATEWAY_INTERFACE'}",
"</html>";
```

Program 5 (b): Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.

```
#!/usr/bin/perl
use CGI':standard';
print
header(),
start_html(-title=>'UNIX COMMAND',-bgcolor=>'#00ffff'),
h1({-align=>'center'},'UNIX COMMAND EXECUTION'),hr(),
startform(-method=>'get',-action=>'./lab5b.cgi'),
'ENTER UNIX COMMAND:', textfield(-name=>'cmd'),br(),
br(),
submit(-value=>'EXECUTE'),
endform(),
hr(),
'$',$cmd=param("cmd"),
br(),
pre(`$cmd`),
end_html();
```

Program 6 (a): Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.

```
#!/usr/bin/perl
use CGI‟:standard‟;
print
header(),
start_html(-title=>‟GREETING‟,-bgcolor=>‟brown‟),
h1({-align=>‟center‟},‟USER GREETING‟),
hr,
font({-color=>‟lightblue‟,-weight=>‟bold‟,-size=>‟5‟},‟ENTER
THE USER NAME HERE‟),
br,
start_form(-action=>‟./6a.cgi‟),
textfield(-name=>‟name‟),
submit(-value=>‟SEND‟),
end_form(),
hr,
font({-color=>‟yellow‟,-size=>‟5‟},‟DEAR USER‟),
font({-color=>‟white‟,-size=>‟8‟},$username=param(„name‟),
br);
@msgs=(‟GOOD”,”BAD”,”HELLO”,”HELL”);
print
font({-color=>‟lightblue‟,-size=>‟8‟},$msgs[int rand scalar
@msgs]),
hr,
end_html;
```

Program 6 (b): Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.

```
#!/usr/bin/perl
use CGI‟:standard‟;
print
header(),
start_html(-bgcolor=>‟lightyellow‟),
h1({-align=>‟center‟},‟PAGE VISITOR INFO‟),
hr,
font({-color=>‟lightblue‟,-size=>‟4‟},‟This page is visited‟),
br,
font({-color=>‟orange‟,-size=>‟7‟},`grep „/cgi-bin/6b.cgi‟
accesslog | wc –l`),
br,
font({-color=>‟lightblue‟,-size=>‟4‟},‟no of times‟),
br,
hr,
end_html;
ln /var/log/httpd/access_log accesslog
```

Program 7: Write a Perl program to display a digital clock which displays the current time of the server.

```
#!/usr/bin/perl
use CGI ':standard';
($sec,$min,$hr)=localtime();
print
header(),
start_html(-title=>'digital clock',-bgcolor=>'#00ffff'),
h1({-align=>'center'},'DIGITAL CLOCK'),
hr(),
h3({-align=>'center'},"The Current Time is:"),
h1({-align=>'center'},b("$hr:$min:$sec")),
"<meta http-equiv='refresh' content = '1'>",
hr(),
end_html();
```

Program 8: Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.

```
#!/usr/bin/perl
use CGI':standard';
use DBI;
$dbh = DBI->connect(“DBI:mysql:veena”,”root”) or die “can not
connect”.DBI->errstr();
$sth = $dbh->prepare(„insert into userinfo values(?,?)‟) or
die “can not insert”.$dbh->errstr();
$sth1 = $dbh->prepare(„select * from userinfo‟) or die “cannot
select”.$dbh->errstr();
print header(),
start_html(-title=>'database access',-bgcolor=>'#00ffff'),
h1({-align=>'center'},'Database Access'),
hr(),
h2({-align=>'center'},'Database insert'),
„ENTER USER INFORMATION‟,
start_form(-action=>‟./8.cgi‟),
„NAME:‟,
textfield(-name=>‟name‟),
br,
„AGE:‟,
textfield(-name=>‟age‟),br,
submit(-value=>‟Insert‟),
reset(-value=>‟Reset‟),
end_form(),
hr();
$ename = param(„name‟);
$eage = param(„age‟);
```

```
if($ename eq “”)
{
print “do not enter null values”;
}
else
{
$sth->execute($ename, $eage) or die “cannot insert”.$sth-
>errstr();
print “successfully inserted”;
}
print
hr(),
h2({-align=>'center'},'Database display'),
pre(
„userinfo‟
br(),
„-‟x 50,br(),
„NAME AGE‟,br(),
„-‟x 50);
$st1->execute();
while(($ename,$eage)=$sth1->fetchrow())
{
print “<pre>$ename $eage \n</pre>”;
}
$sth1->finish();
$sth->finish();
$dbh->disconnect;
end_html();
```

Program 9: Write a PHP program to store current date-time in a COOKIE and display the ‘Last visited on’ date-time on the web page upon reopening of the same page.

```
<?php
setcookie("lastvisit",date("h:i-m/d/y"),time()+60*60);
?>
<html>
<body bgcolor=”lightyellow”>
<font size=20pt>
<p align=”center”><b><u>COOKIE INFORMATION</u></b></p></font>
<hr>
<?
if(isset($_COOKIE["lastvisit"]))
{
echo “<font size=10pt color = ligtblue><p align=center><i>DEAR
USER,YOU LAST VISITED THE SITE ON </i></p></font>”;
echo “<font size=10pt color =white><p
align=center>”.$_COOKIE["lastvisit"];
echo “</p></font>”;
}
```

```
else
echo "You have got some stale cookies";
?>
```

Program 10: Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.

```
<?php
session_start();
?>
<html>
<body bgcolor=”lightyellow”>
<hr>
<h1 align= “center”>SESSION INFORMATION</h1>
<hr>
<?
if(isset($_SESSION['views']))
$_SESSION['views'] = $_SESSION['views']+ 1;
else
$_SESSION['views'] = 1;
?>
<h2><i><b>The page is visited</h2>
<font size=20pt,color=red>
<?
echo $_SESSION['views']
?>
</font><br>
<i><b>Times</b></i>
</body>
</html>
```

Program 11: Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.

INSERT FORM

```
<?xml version=”1.0” encoding = “utf-8”>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<body bgcolor="lightyellow">
<form method ="post" action="13.php">
<table align="center" width="100%" border="1">
<tr>
<td colspan="2" align="center"><h1>USER INFORMATION</h1></td>
</tr>
```

```
<tr>
<td width="50%"><h3>USER INFO ENTRY</h3></td>
<tr>
<td>
<table border="1">
<tr>
<td>Name:</td>
<td><input type="text" name="name"/></td>
</tr>
<tr>
<td>ADDRESS1:</td>
<td><input type="text" name="add1"/></td>
</tr>
<tr>

<td>ADDRESS2:</td>
<td><input type="text" name="add2"/></td>
</tr>
<tr>
<td>EMAIL:</td>
<td><input type="text" name="email"/></td>
</tr>
<tr>
<td colspan="2" align="center">
<input type="submit" value="INSERT"/>&nbsp;
<input type="reset" value="RESET"/></td>
</tr>
</table>
</td>
</tr>
</table>
<a href="13a.html" align = center>CLICK HERE TO SEARCH</a>
</form>
</body>
</html>
```

SEARCH FORM

```
<html>
<head>
<title>Insert title here</title>
</head>

<html>
<body bgcolor = "lightyellow">
<h2 align ="center">DATABASE SEARCH</h2>
<form action = "13a.php "method="post">
<table>
<tr>
<td>
```

```
<p><b>Enter the user name here:</b></p><br/>
<input type="text" name="name1"/>&nbsp;
<input type="submit" value="SEARCH"/>
</td>
</tr>
</table>
<a href="13.html" align = center>HOME</a>
</body>
</html>
13.php
<?php
$mysql = mysql_connect("localhost","root","") or die("Cannot
connect");
?>
<html>
<body bgcolor = "lightyellow">
<h2 align ="center">USER ENTRY</h2>
<?php
$name = $_REQUEST["name"];
$add1 = $_REQUEST["add1"];
$add2 = $_REQUEST["add2"];
$email = $_REQUEST["email"];
if($name == NULL && $add1 == NULL && $add2 == NULL && $email
== NULL)
{
echo "Dont enter null values";
}
else
{
mysql_select_db("perlexample") or die("Cannot select the
database");
$result = mysql_query("insert into user_info
values('$name','$add1','$add2','$email')") or die("Cannot
insert");
echo "Successfully Inserted";
}
?>
<a href="13.html" align = center>HOME</a>
</table>
</body>
</html>

13a.php
<?php
$mysql = mysql_connect("localhost","root","") or die("Cannot
connect");
?>
<html>
<body bgcolor="lightyellow">
```

```
<h2 align = "center">SEARCH RESULTS</h2>
<?php
$name1 = $_REQUEST["name1"];
if($name1==NULL)
{
echo "Dont enter null values";
}
else
{
mysql_select_db("perlexample") or die("Cannot select the
database");

$result1 = mysql_query("select * from user_info where name =
'$name1'") or die("Cant select");
}
echo "<hr>";
echo "<table border= 1 align = center>
<tr>
<td>Name:</td>
<td>Address1:</td>
<td>Address2:</td>
<td>Email:</td>
</tr>";
while($array = mysql_fetch_row($result1))
{
echo "<tr>
<td>$array[0]</td>
<td>$array[1]</td>
<td>$array[2]</td>
<td>$array[3]</td>";
}
echo "</table>";
echo "<hr>";
?>
<a href="13.html" align = center>HOME</a>
</body>
</html>
```

Program 12: Build a Rails application to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.

```
Steps to create a ruby application
```

1. Create a new application, by issuing the command => rails Books
2. Create the model by using the command => ruby script/generate model book
3. Go to the directory db and make the following changes in the 001_create_books.rb

```
t.column :accessionnumber, :string
t.column :title, :string
t.column :author, :string
t.column :edition, :string
t.column :publisher, :string
```

4. Create the controller file by using the command => ruby script/generate controller main
5. Create the database by name books_development and the table by the name books or use the db migrate command
6. Create the front end forms for insert and search and the result forms and save it in the directory /app/views/main/
7. Write the controller code in file named /app/controllers/main_controller.rb.
8. Start the server by using the command ruby script/server

dbinsert.rhtml

```
<html>
<head><title>DBINSERT</title></head>
<body bgcolor="lightyellow">
<h1 align = "center">DBINSERT</h1>
<hr />
<form action = "resultinsert" method="post">
<table>
<tr>
<td> <b>Enter the Accession Number here:</b> </td>
<td> <input type="text" name="book[accessionnumber]"
size="30"/></td>
</tr>
<tr>
<td> <b>Enter the title here:</b> </td>
<td> <input type="text" name="book[title]" size="30"/></td>
</tr>
<tr>
<td> <b>Enter the Author:</b> </td>
<td> <input type="text" name="book[author]" size="30"/></td>
</tr>
<tr>
<td> <b>Enter the Edition:</b> </td>
<td> <input type="text" name="book[edition]" size =
"30"/></td>
</tr>
<tr>
<td> <b>Enter the Publisher here:</b> </td>
<td> <input type="text" name="book[publisher]"
size="30"/></td>
</tr>
<tr>
<td colspan = "2">
<input type="submit" value="INSERT"/>
```

```
<input type="reset" value="RESET"/>
</td>
</tr>
</table>
<hr>
</form>
</body>
</html>
```

resultinsert.rhtml

```
<html>
<head><title>DBSEARCH BASED ON TITLE</title></head>
<body bgcolor="lightyellow">
<h1 align = "center">SUCCESSFULLY INSERTED
</h1>
<hr />
</body>
</html>
```

dbsearch.rhtml

```
<html>
<head><title>DBSEARCH BASED ON TITLE</title></head>
<body bgcolor="lightyellow">
<h1 align = "center">DATABASE SEARCH BASED ON TITLE</h1>
<hr />
<form action = "searchresult" method="post"> <table>
<tr>
<td> <b>Enter the title here:</b> </td>
<td> <input type="text" name="title"/></td>
</tr>
<tr>
<td colspan = "2">
<input type="submit" value="SEARCH"/>
<input type="reset" value="RESET"/>
</td>
</tr>
</table>
<hr>
</form>
</body>
</html>
```

searchresult.rhtml

```
<html>
<head>
<title> result.rhtml </title>
</head>
<body bgcolor="lightyellow">
<h1 align = "center"> SEARCH RESULTS </h1>
```

```
<hr>
<table border = "border" align="center">
<tr>
<th> Accession Number </th>
<th> Title </th>
<th> Author </th>
<th> Edition </th>
<th> Publication </th>
</tr>
<% @names.each do |boo|
@name = boo. accessionnumber
@title = boo.title
@author = boo.author
@edition = boo.edition
@publication = boo.publisher %>
<tr>
<td><%= @name %></td>
<td><%= @title %></td>
<td><%= @author %></td>
<td><%= @edition %></td>
<td><%= @publication%></td>
</tr>
<% end %>
</table>
</body>
</html>
```

main_controller.rb

```
class MainController < ApplicationController
def resultinsert
@book = Book.new(params[:book])
if @book.save
flash[:notice] = 'Book details was successfully created.'
else
render :action => 'dbinsert'
end
end
def searchresult
@titlename = params[:title]
@names = book.find(:all, :conditions =>["title =
?",@titlename])
end
end
```

VIVA QUESTIONS

1. What is HTML?
2. What is a tag?
3. What is the simplest HTML page?
4. How do I create frames? What is a frameset?
5. How can I include comments in HTML?
6. What is a Hypertext link?
7. What is a DOCTYPE? Which one do I use?
8. Can I nest tables within tables?
9. How do you align a table to the right (or left)?
10. How can I use tables to structure forms?
11. What’s relationship between JavaScript and ECMAScript?
12. What are JavaScript types?
13. How do you convert numbers between different bases in JavaScript? –
14. What does isNaN function do? –
15. What is negative infinity? –
16. What boolean operators does JavaScript support?
17. What are Cascading Style Sheets?
18. What is class?
19. What are different selector forms
20. What is grouping?
21. What is ID selector?
22. What is contextual selector?
23. What does \ABCD (and \ABCDE) mean?
24. What are the advantages / disadvantages of various style methods?
25. What is property?
26. What is the CSS clear property?
27. What are the necessities of using HTML forms?
28. What are the sequences of steps for each HTTP request from a client to the server?
29. What is XML?
30. What are the advantages of XML?
31. What does "1"+2+4 evaluate to? –
32. How about 2+5+"8"?
33. What looping structures are there in JavaScript?
34. How do you create a new object in JavaScript?
35. How do you assign object properties?
36. What’s a way to append a value to an array?
37. What is this keyword?
38. What is an Empty HTML Tag?
39. How do I open a link into a new window?
40. How do I let people download a file from my page