

Web Designing Lab

LABORATORY MANUAL

B.Tech Semester -IV

Subject Code: (KIT-451)



Session: 2020-21, Even Semester

Name:	
Roll. No.:	
Group/Branch:	

**JSS MAHAVIDYAPEETHA
DEPARTMENT OF INFORMATION TECHNOLOGY
JSS ACADEMY OF TECHNICAL EDUCATION
C-20/1, SECTOR-62, NOIDA**

Table of Contents

1. Vision and Mission of the Institute
2. Vision and Mission of the Department
3. Programme Educational Objectives (PEOs)
4. Programme Outcomes (POs)
5. Programme Specific Outcomes (PSOs)
6. University Syllabus
7. Course Outcomes (COs)
8. CO- PO and CO-PSO mapping
9. Course Overview
10. List of Experiments
11. DOs and DON'Ts
12. General Safety Precautions
13. Guidelines for students for report preparation
14. Lab assessment criteria
15. Details of Conducted Experiments
16. Lab Experiments

Vision and Mission of the Institute

Vision:

“JSS Academy of Technical Education Noida aims to become an Institution of excellence in imparting quality Outcome Based Education that empowers the young generation with Knowledge, Skills, Research, Aptitude and Ethical values to solve Contemporary Challenging Problems”

Mission:

M1: Develop a platform for achieving globally acceptable level of intellectual acumen and technological competence.

M2: Create an inspiring ambience that raises the motivation level for conducting quality research.

M3: Provide an environment for acquiring ethical values and positive attitude.

Vision and Mission of the Department

Vision:

“To become a Centre of Excellence in teaching and research in Information Technology for producing skilled professionals having a zeal to serve society”

Mission:

M1: To create an environment where students can be equipped with strong fundamental concepts, programming and problem solving skills.

M2: To provide an exposure to emerging technologies by providing hands on experience for generating competent professionals.

M3: To promote Research and Development in the frontier areas of Information Technology and encourage students for pursuing higher education

M4: To inculcate in students ethics, professional values, team work and leadership skills.

Programme Educational Objectives (PEOs)

- PEO1:** To provide students with a sound knowledge of mathematical, scientific and engineering fundamentals required to solve real world problems.
- PEO2:** To develop research oriented analytical ability among students and to prepare them for making technical contribution to the society.
- PEO3:** To develop in students the ability to apply state-of-the-art tools and techniques for designing software products to meet the needs of Industry with due consideration for environment friendly and sustainable development.
- PEO4:** To prepare students with effective communication skills, professional ethics and managerial skills.
- PEO5:** To prepare students with the ability to upgrade their skills and knowledge for life-long learning.

Programme Outcomes (POs)

- PO1: Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- PO2: Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- PO3: Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- PO4: Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5: Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- PO6: The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- PO7: Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- PO8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- PO9: Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- PO10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- PO11: Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO12: Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO1: Analyze, identify and clearly define a problem for solving user needs by selecting, creating and evaluating a computer based system through an effective project plan.

PSO2: Design, implement and evaluate processes, components and/or programs using modern techniques, skills and tools of core Information Technologies to effectively integrate secure IT-based solutions into the user environment.

PSO3: Develop impactful IT solutions by using research based knowledge and research methods in the fields of integration, interface issues, security & assurance and implementation.

University Syllabus

1. To create a simple html file to demonstrate the use of different tags.
2. To create an html file to link to different html page which contains images, tables, and also link within a page.
3. To create an html page with different types of frames such as floating frame, navigation frame & mixed frame.
4. To create a registration form as mentioned below. Procedure: Create an html page named as “registration.html” a) set background colors b) use table for alignment c) provide font colors & size
5. To create an html file by applying the different styles using inline, external & internal style sheets.
6. To write a Javascript program to define a user defined function for sorting the values in an array.
7. To create an html page to explain the use of various predefined functions in a string and math object in java script.
8. To create an html page to explain the use of various predefined functions in a array & Date object in Javascript.
9. To create an html page to demonstrate exception handling in JavaScript
10. To display the calendar using Javascript code by getting the year from the user.
11. To create a html registration form and to validate the form using Javascript code.
12. To create a html file. To open new window from the current window using Javascript.
13. To create an html page to change the background color for every click of a button using Javascript.
14. To create an html page with 2 combo box populated with month & year, to display the calendar for the selected month & year from combo box using Javascript.

15. To create a html page to display a new image & text when the mouse comes over the existing content in the page.

Course Outcomes (COs)

Upon successful completion of the course, the students will be able to

- Design static web pages using HTML.
- Create HTML web pages for demonstration of web based objects.
- Demonstrate dynamic implementation of web pages using HTML, CSS, XML, and JavaScript.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C214.1	1		3		3			1	2	1	3	1
C214.2	1		3		3			1	2	1	3	1
C214.3	1		3		3			1	2	1	3	1

CO-PSO Mapping

	PSO1	PSO2	PSO3
C214.1	2	2	
C214.2	2	2	
C214.3	2	2	

Course Overview

In order to make websites look and function a certain way, web developers utilize different languages. The three core languages that make up the World Wide Web are HTML5, CSS, and JavaScript. In the IT world, the internet is an essential platform, whether it's for developing or for consumer use. When developing a website, typically three main languages come into play. These languages are JavaScript, CSS, and HTML. HTML is the backbone of most webpages. Essentially, it is used to create the structure of how a specific website would look like, from the headings, to the paragraphs, the body, links, and even images.

This course is intended to teach the basics involved in publishing content on the World Wide Web. This includes the 'language of the Web' – HTML, the fundamentals of how the Internet and the Web function, a basic understanding of graphic production with a specific stress on creating graphics for the Web, and a general grounding introduction to more advanced topics such as programming and scripting. This will also expose students to the basic tools and applications used in Web publishing. The objective of this lab is to develop an ability to design and implement static and dynamic website. The courses contains web basics: Design web pages through coding using HTML and DHTML, Browser side scripting using JavaScript with a focus on, event handling and validation ,server side scripting: Php syntax, variables, loops and constructs. Java graphics, Browser side scripting: Introduction to programming world of xml technologies.

List of Experiments mapped with Cos

<i>Sl No.</i>	<i>List of Experiments</i>	<i>Course Outcome</i>
1	To create a simple html file to demonstrate the use of different tags.	C214.1
2	To create an html file to link to different html page which contains images, tables, and also link within a page.	C214.1
3	To create an html page with different types of frames such as floating frame, navigation frame & mixed frame.	C214.1
4	To create a registration form as mentioned below. Procedure: Create an html page named as “registration.html” a) set background colors b) use table for alignment c) provide font colors & size	C214.1
5	To create an html file by applying the different styles using inline, external & internal style sheets.	C214.3
6	To write a Javascript program to define a user defined function for sorting the values in an array.	C214.3
7	To create an html page to explain the use of various predefined functions in a string and math object in java script.	C214.2
8	To create an html page to explain the use of various predefined functions in a array & Date object in Javascript.	C214.2
9	To create an html page to demonstrate exception handling in JavaScript	C214.2
10	To display the calendar using javascript code by getting the year from the user.	C214.3

Web Designing Lab (KIT-451)

11	To create a html registration form and to validate the form using javascript code.	C214.3
12	To create a html file. To open new window from the current window using javascript.	C214.3
13	To create an html page to change the background color for every click of a button using Javascript.	C214.3
14	To create an html page with 2 combo box populated with month & year, to display the calendar for the selected month & year from combo box using javascript.	C214.3
15	To create a html page to display a new image & text when the mouse comes over the existing content in the page.	C214.3

DOs and DON'Ts

DOs

1. Login-on with your username and password.
2. Log off the Computer every time when you leave the Lab.
3. Arrange your chair properly when you are leaving the lab.
4. Put your bags in the designated area.
5. Ask permission to print.

DON'Ts

1. Do not share your username and password.
2. Do not remove or disconnect cables or hardware parts.
3. Do not personalize the computer setting.
4. Do not run programs that continue to execute after you log off.
5. Do not download or install any programs, games or music on computer in Lab.
6. Personal Internet use chat room for Instant Messaging (IM) and Sites is strictly prohibited.
7. No Internet gaming activities allowed.
8. Tea, Coffee, Water & Eatables are not allowed in the Computer Lab.

General Safety Precautions

Precautions (In case of Injury or Electric Shock)

1. To break the victim with live electric source, use an insulator such as fire wood or plastic to break the contact. Do not touch the victim with bare hands to avoid the risk of electrifying yourself.
2. Unplug the risk of faulty equipment. If main circuit breaker is accessible, turn the circuit off.
3. If the victim is unconscious, start resuscitation immediately, use your hands to press the chest in and out to continue breathing function. Use mouth-to-mouth resuscitation if necessary.
4. Immediately call medical emergency and security. Remember! Time is critical; be best.

Ambulance : **9810611477(Fortis Ambulance)**
120-2400222(Fortis Ambulance)

Security : **260 (Gate No.1)**
230 (Gate No.2)

Precautions (In case of Fire)

1. Turn the equipment off. If power switch is not immediately accessible, take plug off.
2. If fire continues, try to curb the fire if possible by using the fire extinguisher or by covering it with a heavy cloth if possible isolate the burning equipment from the other surrounding equipment.
3. Sound the fire alarm by activating the nearest alarm switch located in the hallway.
4. Call security and emergency department immediately:

Emergency : **219 (Reception)**
298(Health Center)

Security : **260 (Gate No.1)**
230 (Gate No.2)

Guidelines to students for report preparation

All students are required to maintain a record of the experiments conducted by them. Guidelines for its preparation are as follows:-

- 1) All files must contain a title page followed by an index page. *The files will not be signed by the faculty without an entry in the index page.*
- 2) Student's Name, Roll number and date of conduction of experiment must be written on all pages.
- 3) For each experiment, the record must contain the following
 - (i) Aim/Objective of the experiment
 - (ii) Pre-experiment questions.
 - (iii) Algorithm
 - (v) Results/ output

Note:

1. Students must bring their lab record along with them whenever they come for the lab.
2. Students must ensure that their lab record is regularly evaluated.

Lab Assessment Criteria

An estimated 10 lab classes are conducted in a semester for each lab course. These lab classes are assessed continuously. Each lab experiment is evaluated based on 5 assessment criteria as shown in following table. Assessed performance in each experiment is used to compute CO attainment as well as internal marks in the lab course.

Grading Criteria	Exemplary (4)	Competent (3)	Needs Improvement (2)	Poor (1)
AC1: Pre-Lab written work (for last lab class, this may be assessed through viva)	Complete procedure with underlined concept is properly written	Underlined concept is written but procedure is incomplete	Not able to write concept and procedure	Underlined concept is not clearly understood
AC2: Program Writing/ Modeling	Assigned problem is properly analyzed, correct solution designed, appropriate language constructs/ tools are applied, Program/solution written is readable	Assigned problem is properly analyzed, correct solution designed, appropriate language constructs/ tools are applied	Assigned problem is properly analyzed & correct solution designed	Assigned problem is properly analyzed
AC3: Identification & Removal of errors/ bugs	Able to identify errors/ bugs and remove them	Able to identify errors/ bugs and remove them with little bit of guidance	Is dependent totally on someone for identification of errors/ bugs and their removal	Unable to understand the reason for errors/ bugs even after they are explicitly pointed out
AC4: Execution & Demonstration	All variants of input /output are tested, Solution is well demonstrated and implemented concept is clearly explained	All variants of input /output are not tested, However, solution is well demonstrated and implemented concept is clearly explained	Only few variants of input /output are tested, Solution is well demonstrated but implemented concept is not clearly explained	Solution is not well demonstrated and implemented concept is not clearly explained
AC5: Lab Record Assessment	All assigned problems are well recorded with objective, design constructs and solution along with	More than 70 % of the assigned problems are well recorded with objective, design	Less than 70 % of the assigned problems are well recorded with objective, design contracts and solution	Less than 40 % of the assigned problems are well

Web Designing Lab (KIT-451)

	Performance analysis using all variants of input and output	contracts and solution along with Performance analysis is done with all variants of input and output	along with Performance analysis is done with all variants of input and output	recorded with objective, design contracts and solution along with Performance analysis is done with all variants of input and output
--	---	--	---	--

LAB EXPERIMENTS

LAB EXPERIMENT 1

OBJECTIVE:

Write HTML Script to create a simple html file to demonstrate the use of different tags.

BRIEF DESCRIPTION:

HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language.
- HTML describes the structure of Web pages using markup.
- HTML elements are the building blocks of HTML pages.
- HTML elements are represented by tags.
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on.
- Browsers do not display the HTML tags, but use them to render the content of the page.

Basic HTML Tags

- The <!DOCTYPE html> declaration defines this document to be HTML5
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the document
- The <title> element specifies a title for the document
- The <body> element contains the visible page content
- The <h1> element defines a large heading
- The <p> element defines a par

PRE-EXPERIMENT QUESTIONS:

1. What is Internet Browser?
2. What are the basic browser we used to execute HTML?
3. Difference between text & graphics?

Explanation:

Open an Editor and use the following tags to create a resume.

Begin Tag	End Tag	Summary
<p>	</p>	Separates two blocks of text by denoting a paragraph break. To justify a paragraph, use ALIGN="left, right, justify, or indent" inside the tag.
		Changes text between tags to a bold font.

Web Designing Lab (KIT-451)

<code></code>	<code></code>	Logically strengthens the text between the tags.
<code><i></code>	<code></i></code>	Changes text between tags to an italic font.
<code><u></code>	<code></u></code>	Underlines text between the tags.
<code></code>	<code></code>	Defines an unordered list consisting of one or more <code></code> elements.
<code></code>	<code></code>	Defines an ordered list in which each of one or more <code></code> elements are automatically numbered.
<code></code>	<code></code>	Defines a list item.
<code><hr></code>	<code></hr></code>	Produces a divider between sections of text
<code><table></code>	<code></table></code>	Defines a series of rows and columns to format the placement of text and image on the page.
<code><tr></code>	<code></tr></code>	Defines a row of a table.
<code><td></code>	<code></td></code>	Defines a data cell. The data cell contains the actual text or image that is to be displayed in a table cell.
<code><center></code>	<code></center></code>	Centers the text vertically between the left and right margins.
<code></code>		Specifies an image file that is to be displayed. The '...' must be replaced with the path and filename of the image. Additional parameters include: ALIGN="alignment option for image" ALT="description of image" WIDTH="width of image" HEIGHT="height of image" BORDER="value to specify width of border" 0 indicates no border"
<code><a></code>	<code></code>	Defines text as a hypertext link. The element must have either the HREF or NAME attribute defined inside it. <code>Text to display</code> <code>Text to display</code>
<code><BQ></code>	<code></BQ></code>	Defines a separated multi-line set of text to be rendered as quoted text.
<code><blink></code>	<code></blink></code>	Causes the text between the tags to blink. This tag is rarely considered professional by web developers and therefore should be used very sparingly.

Sample Output:

Students will be able to design simple resume.

POST EXPERIMENT QUESTIONS:

1. Explain the table properties in HTML, how it is used?
2. Write the Frames properties .How frame is divided in to three windows?
3. Explain CSS & its Types.

LAB EXPERIMENT 2

OBJECTIVE:

To create an html file to link to different html page which contains images, tables, and also link within a page.

BRIEF DESCRIPTION:

Images are defined with the tag. The tag is empty, it contains attributes only, and does not have a closing tag. The src attribute specifies the URL (web address) of the image. Table is defined with the <table> tag. Each table row is defined with the <tr> tag. A table header is defined with the <th> tag. By default, table headings are bold and centered. A table data/cell is defined with the <td> tag. HTML links are hyperlinks. You can click on a link and jump to another document. When you move the mouse over a link, the mouse arrow will turn into a little hand.

PRE-EXPERIMENT QUESTIONS:

1. Explain the concept of Navigation?
2. Explain Hypertext and clickable Image?
3. What is rowspan and colspan?

Explanation:

HTML Image Tags

<i>Tags</i>	<i>Description</i>
<u></u>	Defines an image
<u><map></u>	Defines an image-map
<u><area></u>	Defines a clickable area inside an image-map
<u><picture></u>	Defines a container for multiple image resources

PSEUDO-CODE:

Img tag: ``

Attributes of Img tag : Src, Alt, Style, Height, Width.

Example:

```

```

```

```

Map Tag:

```

```

```
<map name="planetmap">
```

```
  <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">
```

```
  <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm">
```

```
  <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm">
```

```
</map>
```

Picture Tag:

```
<picture>
```

```
  <source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
```

```
  <source media="(min-width: 465px)" srcset="img_white_flower.jpg">
```

```
  
```

```
</picture>
```

Table syntax

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Link syntax

```
<a href="url">link text</a>
```

Example:Visit our HTML tutorial

Sampled Output: Students will be able to design a html page with clickable image, table and hypertext which will navigate the user to other page.

POST EXPERIMENT QUESTIONS:

1. How a particular area of map became clickable?
2. What is hypertext?
3. What are the types of navigation?

LAB EXPERIMENT 3

OBJECTIVE:

To create an html page with different types of frames such as floating frame, navigation frame & mixed frame.

BRIEF DESCRIPTION:

Frames : **<frame> tag** define the particular area within an **HTML** file. A **<frame> tag** is used with **<frameset>** and it divides a webpage into multiple sections or **frames**, each **frame** can contain different web pages.

Floating frame: The **floating frame** is used to create an inline framed region or window that acts similarly to any other embedded object in so far as text can be flowed around it.

Navigation frame: A **navigation frame** contain **frame a** and **frame b**, which on click **frame a** and **frame b** hyperlink open the respective **html** page.

Mixed frame: **Mixed Frameset** is used to display vertical and horizontal **frameset** in a same web page.

PRE-EXPERIMENT QUESTIONS:

1. What is frame?
2. Explain the different types of frames.

PSEUDO-CODE:

```
<frameset cols="25%,50%,25%">
    <frame src="frame_a.htm">
    <frame src="frame_b.htm">
    <frame src="frame_c.htm">
</frameset>
```

SAMPLED OUTPUT:

Different pages are displayed on different frames.

POST-EXPERIMENT QUESTIONS:

1. What are the usage of frames?
2. What is the latest version of HTML?
3. Frame tag is not compatible with which version of HTML?

LAB EXPERIMENT 4

OBJECTIVE:

To create a registration form as mentioned below.

Procedure: Create an html page named as “registration.html”

- a) Set background colors
- b) Use table for alignment
- c) Provide font colors & size.

BRIEF DESCRIPTION:

The <form> tag is used to create an HTML form for user input. The <form> element can contain one or more of the following form elements: <input>, <textarea>, <button>, <select>, <option>, <optgroup>, <fieldset>, <label>, <output>.

PRE-EXPERIMENT QUESTIONS:

1. List the components of Form.
2. What is order and unordered list?

PSEUDO-CODE:

```
<form action="#">  
  First name: <input type="text" placeholder = "First Name" value="">  
  <br><br>  
  Last name: <input type="text" placeholder = "Last Name" value="">  
  <br><br> <input type="submit" value="Submit">  
</form>
```

SAMPLED OUTPUT:

Student registration form as the output.

POST-EXPERIMENT QUESTIONS:

1. Explain check box, radio button, ordered list and unordered list.
2. What are the target value?

LAB EXPERIMENT 5

OBJECTIVE:

To create an html file by applying the different styles using inline, external & internal style sheets.

BRIEF DESCRIPTION:

Cascading Style Sheets (**CSS**) is a simply designed language intended to simplify the process of making web pages presentable. CSS allows to apply styles to web pages.

Inline CSS: Inline CSS contains the CSS property in the body section attached with element. This kind of style is specified within an HTML tag using style attribute.

Internal CSS: The CSS rule set is written within the HTML file in the head section.

External CSS: External CSS contains separate CSS file which contains only style property with the help of tag attributes (For example class, id, heading) CSS property written in a separate file with .css extension and should be linked to the HTML document using **link** tag.

PRE-EXPERIMENT QUESTIONS:

1. What is CSS?
2. What are the types of CSS?

PSEUDO-CODE:

Inline CSS:

```
<head> <title>Inline CSS</title> </head>
<body>
<p style = "color:#009900; font-size:50px;font-style:italic; text-align:center;">
    GeeksForGeeks </p>
</body>
```

Internal CSS:

```
<head>
    <title>Internal CSS</title>
    <style>
        .main {text-align:center; }
        .GFG {color:#009900;
            font-size:50px;
            font-weight:bold; }
        .geeks { font-style:bold;
            font-size:20px;}
    </style>
</head>
<body>
    <div class = "main">
        <div class ="GFG">GeeksForGeeks</div>
        <div class ="geeks">A computer science portal for geeks. </div>
    </div>
</body>
```

External CSS:

```
body { background-color:powderblue; }
.main{ text-align:center; }
```

Web Designing Lab (KIT-451)

```
.GFG {color:#009900; font-size:50px; font-weight:bold;}  
#geeks { font-style:bold; font-size:20px; }
```

SAMPLED OUTPUT:

Well formatted and designed web page.

POST-EXPERIMENT QUESTIONS:

1. What is selector?
2. What is the use of selector for different section of web page?

LAB EXPERIMENT 6

OBJECTIVE:

To write a Javascript program to define a user defined function for sorting the values in an array.

BRIEF DESCRIPTION:

JavaScript was once upon a time used only in client side but node js have made possible to run javascript on server side. JavaScript is everywhere – on Desktop/Server/Mobile. JavaScript has stormed the web technology and node js is used for web apps. However JavaScript is quirky/dynamic/scripting/ functional oriented language and it has its own characteristics. It is not scalable, it is good for some 3000 line of code but for a bigger app, it becomes difficult to manage, read and debug.

PRE-EXPERIMENT QUESTIONS:

1. What is JavaScript?
2. What is client-server Architecture?

PSEUDO-CODE:

function displaySort()

```
{ int[] numbers = {4, 11, 13, 12, 17, 35, 15, 7, 19, 3, 45};  
  for (int i = 0; i < numbers.length - 1; i++)  
  {for(int j = 0; j < numbers.length - 1; j++)  
    { if(numbers[i] < numbers[j + 1])  
      { tempVar = numbers [j + 1];  
        numbers [j + 1]= numbers [i];  
        numbers [i] = tempVar;}  
    }  
  }  
}
```

SAMPLED OUTPUT:

Sorted array.

POST-EXPERIMENT QUESTIONS:

1. Enumerate the differences between Java and JavaScript?
2. What are JavaScript Data Types?

LAB EXPERIMENT 7

OBJECTIVE:

To create an html page to explain the use of various predefined functions in a string and math object in java script.

BRIEF DESCRIPTION:

Object, in JavaScript, is most important data-type and the building blocks for modern JavaScript.

Strings in Java are Objects that are backed internally by a char array. Since arrays are immutable as well as Strings are immutable. Whenever a change to a String is made, an entirely new String is created.

The **Math object** is used to perform mathematical operations on numbers.

PRE-EXPERIMENT QUESTIONS:

1. What are the uses of string and Math object in Java Script?
2. List the method of Math objects.

PSEUDO-CODE:

String Object:

```
var val = new String(string);
```

Math Object:

```
<script>
    document.getElementById("GFG").innerHTML =
        "Math.LN10: " + Math.LN10 + "<br>" +
        "Math.LOG2E: " + Math.LOG2E + "<br>" +
        "Math.Log10E: " + Math.LOG10E + "<br>" +
        "Math.SQRT2: " + Math.SQRT2 + "<br>" +
        "Math.SQRT1_2: " + Math.SQRT1_2 + "<br>" +
        "Math.LN2: " + Math.LN2 + "<br>" +
        "Math.E: " + Math.E + "<br>" +
        "Math.PI: " + Math.PI;
</script>
```

SAMPLED OUTPUT:

S1=Java, S2 Script

Concatenated string = JavaScript

POST-EXPERIMENT QUESTIONS:

1. List the use of String and Math Objects.
2. What are the various way to use String and Math objects.

LAB EXPERIMENT 8

OBJECTIVE:

To create an html page to explain the use of various predefined functions in an array & Date object in JavaScript.

BRIEF DESCRIPTION:

Objects, in JavaScript, is its most important data-type and forms the building blocks for modern JavaScript.

The **Array** object lets you store multiple values in a single variable. It stores a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

The Date object is an inbuilt datatype of JavaScript language. It is used to work with dates and times. The Date object is created by using new keyword.

PRE-EXPERIMENT QUESTIONS:

1. What are the uses of array and date objects in JavaScript?
2. List some methods of Date object.

PSEUDO-CODE:

```
<ol>
  <li> Array Object </li>
    <ul>
      <li> toString(): Converts an array to a string of array values. </li>
      <li> join(): Joins all array elements into a string. </li>
    </ul>
  <li> Date Object </li>
    <ul>
      <li> getDate(): Get the day as a number (1-31). </li>
      <li> getHours(): Get the hour (0-23). </li>
    </ul>
</ol>
```

SAMPLED OUTPUT:

1. Array Object

- `toString()`: Converts an array to a string of array values.
- `join()`: Joins all array elements into a string.

2. Date Object

- `getDate()`: Get the **day** as a number (1-31).
- `getHours()`: Get the **hour** (0-23).

POST-EXPERIMENT QUESTIONS:

1. List and explain at least ten methods of each object.
2. What are the various ways to declare an object?

LAB EXPERIMENT 9

OBJECTIVE:

_To create an html page to demonstrate exception handling in JavaScript

BRIEF DESCRIPTION:

Exception handling is the process of responding to the occurrence, during computation, of exceptions – anomalous or exceptional conditions requiring special processing – often disrupting the normal flow of program execution. It is provided by specialized programming language constructs, computer hardware mechanisms which includes interrupts or operating system IPC facilities like signals.

PRE-EXPERIMENT QUESTIONS:

1. What is exceptional handling?
2. What are the various keywords used in exceptional handling?

PSEUDO-CODE:

```
<p id="error"></p>

    <script>
        try {
            addlert("Welcome!");
        }
        catch(err) {
            document.getElementById("error").innerHTML = err.message;
        }
    </script>
```

SAMPLED OUTPUT:

Addlert is not defined

POST-EXPERIMENT QUESTIONS

1. Where do you use throw keyword?
2. Use every keyword associated with exceptional handling.

LAB EXPERIMENT 10

OBJECTIVE:

_To display the calendar using JavaScript code by getting the year from the user.

BRIEF DESCRIPTION:

Functions are one of the fundamental building blocks in JavaScript. A function is a JavaScript procedure—a set of statements that performs a task or calculates a value. To use a function, you must define it somewhere in the scope from which you wish to call it.

A function definition (also called a function declaration, or function statement) consists of the function keyword, followed by:

- The name of the function.
- A list of parameters to the function, enclosed in parentheses and separated by commas.
- The JavaScript statements that define the function, enclosed in curly brackets

PRE-EXPERIMENT QUESTIONS:

1. What are the types of functions that can be used in JavaScript?
2. Suggest some applications that you can build using functions.

PSEUDO-CODE:

Step 1 – Input the year and pass it to a function that checks if it is leap or not.

Step 2 – According to the given year, retrieve the information day on first of January.

Step 3 – Create a table for each month as reference from first day and date.

Step 4 – Show this table via document object on the page

SAMPLE OUTPUT

Year 2020

Calendar of year 2020 must be displayed on webpage.

POST-EXPERIMENT QUESTIONS

1. Which document method is used to show data on a webpage?
2. In how many ways can you use JavaScript in your HTML document?

LAB EXPERIMENT 11

OBJECTIVE:

To create a html registration form and to validate the form using JavaScript code

BRIEF DESCRIPTION:

Forms are used in webpages for the user to enter their required details that are further send it to the server for processing. A form is also known as web form or HTML form. Examples of form use are prevalent in e-commerce websites, online banking, online surveys to name a few.

The data entered into a form needs to be in the right format and certain fields need to be filled in order to effectively use the submitted form. Username, password, contact information are some details that are mandatory in forms and thus need to be provided by the user.

PRE-EXPERIMENT QUESTIONS:

1. How do you insert forms in a webpage?
2. Why do we need validation in form?

PSEUDO-CODE:

```
<script>
    function validate()
    {
        var name = document.forms["Form"]["name_field"];
        if (name.value == "")
        {
            window.alert("Please enter your name.");
            name.focus();
            return false;
        }
    }
}
```

```
        return true;
    }
</script>
```

SAMPLED OUTPUT:

An embedded page on this page says

Name must be filled out

POST-EXPERIMENT QUESTIONS

1. How do forms submit the values entered into the fields?
2. How can we introduce custom validation message?

LAB EXPERIMENT 12

OBJECTIVE:

To create a html file to open new window from the current window using JavaScript.

BRIEF DESCRIPTION:

The window object represents a window in browser. An object of window is created automatically by the browser. Window is the object of browser; it is not the object of JavaScript. The JavaScript objects are string, array, date etc. The window object is supported by all browsers. It represents the browser's window. All global JavaScript objects, functions, and variables automatically become members of the window object. Global variables are properties of the window object. Global functions are methods of the window object.

PRE-EXPERIMENT QUESTIONS:

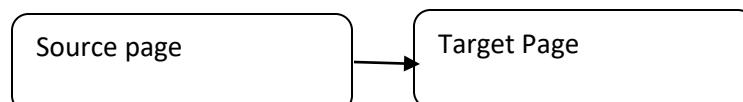
1. What do you know about the window object? Whose object is it?
2. List some methods of windows object?

PSEUDO-CODE:

```
<button onclick="newWindow()">Launch</button>

<script>
    function newWindow() {
        var myWindow = window.open("_blank");
    }
</script>
```

SAMPLE OUTPUT



POST-EXPERIMENT QUESTIONS:

1. Write a note about JS Browser BOM?
2. What is the difference between DOM and BOM?

LAB EXPERIMENT 13

OBJECTIVE:

To create an html page to change the background color for every click of a button using JavaScript.

BRIEF DESCRIPTION

The backgroundColor property in HTML DOM is used to set or return the background-color of an element.

Syntax:

object.style.backgroundColor

It returns the background color of an element.

PRE-EXPERIMENT QUESTIONS:

1. Write a short note on DOM.
2. How do we set an initial value for a background color?

ALGORITHM:

```
<script>

    var el = document.getElementById("g1");
    function changeColor(color) {
        document.body.style.background = color;
    }

    function _Run() {
        r = Math.floor(Math.random() * 256);
        var rf = Number(r).toString(16);
        if (rf.length < 2) {
            rf = "0" + rf;
        }
        g = Math.floor(Math.random() * 256);
        var gf = Number(g).toString(16);
```

```
        if (gf.length < 2) {  
            gf = "0" + gf;  
        }  
        b = Math.floor(Math.random() * 256);  
        var bf = Number(b).toString(16);  
        if (bf.length < 2) {  
            bf = "0" + bf;  
        }  
        var c = '#' + rf + gf + bf;  
        changeColor(c);  
        el.innerHTML = "Background Color changed";  
    }  
</script>
```

SAMPLE OUTPUT

On click background color will be changed.

POST-EXPERIMENT QUESTIONS:

1. What is the purpose of using floor and random functions in the above pseudocode?
2. Write a short note on concatenation of strings in JS.
3. How are colors represented in HTML?

LAB EXPERIMENT 14

OBJECTIVE:

To create an html page with 2 combo boxes populated with month & year, to display the calendar for the selected month & year from combo box using JavaScript.

BRIEF DESCRIPTION:

A drop-down menu is a menu that offers a list of options. The title of the menu, or the currently-selected item in the list, is always displayed. When the visible item is clicked, other items from the list "drop down" into view, and the user can choose from those options.

The HTML <input> element is used to create interactive controls for web-based forms in order to accept data from the user; a wide variety of types of input data and control widgets are available, depending on the device and user agent. The <input> element is one of the most powerful and complex in all of HTML due to the sheer number of combinations of input types and attributes.

PRE-EXPERIMENT QUESTIONS:

1. What do you know about the <input> tag in HTML?
2. How do you deselect a default option in drop down menu?

PSEUDO-CODE:

- Step 1 – Input the month and year and pass it to a function that checks if it is leap or not.
- Step 2 – According to the given combination, retrieve the information day on first of that month.
- Step 3 – Create a table for this combination with dates and days like in a calendar.
- Step 4 – Show this table on HTML webpage.

SAMPLE OUTPUT

Calendar of particular month must be displayed.

POST-EXPERIMENT QUESTIONS:

1. How do you send information from JavaScript function to be rendered on a webpage?
2. What CSS elements can be applied to the table or calendar to make it more appealing?

LAB EXPERIMENT 15

OBJECTIVE:

To Write a Program for merging two unsorted-students-name-list in sorted order

BRIEF DESCRIPTION:

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML. CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

CSS is one of the core languages of the open Web and is standardized across Web browsers according to the W3C specification. Developed in levels, CSS1 is now obsolete, CSS2.1 is a recommendation, and CSS3, now split into smaller modules, is progressing on the standardization track.

PRE-EXPERIMENT QUESTIONS:

1. How do you insert images in a HTML document?
2. Explain the types of CSS?

ALGORITHM:

```
<style>
```

```
.hiddenimg {  
    display: none;  
}
```

```
.hiddentxt {  
    font-weight: bold;  
    color: #F00;  
}
```

```
.hiddentxt:hover ~ .hiddenimg {  
    display: block;  
}
```

```
</style>
```

```
<body>
```

Web Designing Lab (KIT-451)

<p>

Hover to see a image.

</p>

</body>

SAMPLE OUTPUT

Input:

<u>abc</u>	<u>abf</u>	<u>abp</u>	<u>abd</u>
------------	------------	------------	------------

Output

<u>abc</u>	<u>abd</u>	<u>abf</u>	<u>abp</u>
------------	------------	------------	------------

POST-EXPERIMENT QUESTIONS:

1. How do you change properties on hovering of mouse?
2. Can you implement the above problem in JavaScript? If yes, how? If no, why?

This lab manual has been prepared by

Mr. Sukhendra Singh
(sukhendrasingh@jssaten.ac.in)

Mr Manoj Kumar
(manojkumar@jssaten.ac.in)

Please spare some time to provide your valuable feedback