`**National University of Computer and Emerging Sciences**



**Lab Manual ICT**

|  |  |  |
| --- | --- | --- |
| **Submitted By** | Muhammad Musawar Ali Shah | |
| **Session** | Fall 2024 | |
| **Reg. No** | 24P-0619 | |
| **Marks** | **Total: 5** | **Obtained :** |

# Declaration

I **Muhammad Musawar Ali Shah** under registration number **24P-0619** hereby declare that this Lab Assessment for ICT is entirely my own original work. I have not copied any part of this submission from any other source or individual. Should any similarity or plagiarism be found in this document, I acknowledge that I will be held responsible for academic dishonesty and will be subject to the disciplinary actions outlined in the academic rules of FAST University.

**Checked by: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**\_\_\_\_\_\_

# List of Experiments

1. To join the google classroom.
2. Windows Installation
3. Introduction to Google Docs
4. Introduction to Ubuntu linux
5. Introduction to Latex
6. Introduction to Microsoft PowerPoint
7. Introduction to Microsoft Excel
8. Basics of HTML
9. Local server.
10. Creating a web page using HTML
11. Basics of JavaScript
12. Buttons and forms.
13. Hyperlinks and page linking

# LAB ASSESSMENT 1: JOINING THE GOOGLE CLASSROOM

**Objective**

To guide the students to join the Google Clasroom using the class code provided by the instructor

**Apparatus/Materials**

1. Laptop or smartphone with an internet connection.
2. An active Google account
3. Google Classroom installed on your mobile or any updated web browser
4. Class code provided by the instructor

**Theory**

Google Classroom is an online platform for teachers and students to share materials, assignments, and announcements. Using a class code, students can securely join a virtual classroom to access resources and collaborate effectively.

**Procedure**

You can join the google classroom using in web browsers as well as through mobile application

***Part 1: Accessing through web browser***

* Open the web browser **Google chrome recommended.**
* Goto to URL: <https://classroom.google.com>
* Sign In with your google account
* Join the classroom
* Enter the class code provided by the instructor
* Click on enter the classroom

***Part 2: Through mobile application***

***Download the App:***

* Install the **Google Classroom** app from the Google Play Store (Android) or App Store (iOS).

***Sign In:***

* Open the app and sign in using your Google Account credentials.

***Join a Class:***

* Tap the **"+"** icon at the bottom-right corner.
* Select **"Join Class"** and input the class code.
* Tap **"Join"** to access the class.

**Lab Task**

* To join the Google classroom with the class code provided by the lab instructor

**Conclusion**

This lab familiarizes students with the process of accessing and joining a virtual classroom on Google Classroom. It ensures that students are prepared to participate in online learning activities seamlessly.

**Questions**

1. How does ICT improve access to education for remote learners?
2. What are the potential drawbacks of over-reliance on ICT in education?
3. Which ICT tool did you find most effective, and why?
4. Suggest improvements to the ICT tools used in this experiment.

# LAB ASSESSMENT 2: INSTALLING WINDOWS

**Objective**

To guide students through the process of installing the Windows operating system on a computer.

**Apparatus/Materials**

* A computer or laptop for installation.
* A bootable USB drive or DVD with the Windows installation files.
* Access to the internet (optional for updates).
* Windows Product Key (if required).

**Theory**

Windows is a widely used operating system (OS) developed by Microsoft. It provides a user-friendly interface for personal and professional use. Installing Windows involves preparing the hardware, booting from installation media, and configuring system settings. A proper installation ensures optimal system performance and usability.

**Procedure**

You can install Windows using a bootable USB or DVD as installation media.

***Part 1: Preparing the Installation Media***

1. Create a bootable USB using tools like **Rufus** or the **Windows Media Creation Tool** (if not already prepared).
2. Ensure the USB or DVD contains the correct Windows version required for the installation.

***Part 2: Installing Windows***

1. ***Insert the Installation Media:***
   * Plug the bootable USB into the computer or insert the DVD.
2. ***Access the Boot Menu:***
   * Restart the computer and press the appropriate key (e.g., **F2**, **F12**, **DEL**, or **ESC**) to open the boot menu.
   * Select the USB or DVD as the boot device.
3. ***Start Installation:***
   * Once the system boots, select your language, time, and keyboard preferences, and click **Next**.
   * Click **Install Now** to begin.
4. ***Enter Product Key:***
   * If prompted, enter the Windows Product Key or choose **I don’t have a product key** to proceed (you can activate Windows later).
5. ***Choose Installation Type:***
   * Select **Custom: Install Windows Only** for a clean installation.
6. ***Partition the Drive:***
   * Select the partition where Windows will be installed. Format it if necessary, or create a new partition.
   * Click **Next** to start the installation process.
7. ***Complete Installation:***
   * Wait while Windows installs files, expands them, and applies updates.
   * The system will restart several times during the process.
8. ***Set Up Windows:***
   * After installation, follow on-screen instructions to set up Windows (e.g., username, password, and region).
   * Connect to a network if needed, and customize privacy settings.

**Lab Task**

* Perform a clean installation of Windows on a computer or laptop using a bootable USB or DVD.

**Conclusion**

This lab introduces students to the process of installing the Windows operating system, preparing them to handle hardware setup and software configuration independently.

**Questions**

1. Why is it essential to format or partition the drive during a clean installation?
2. What are the benefits of a clean Windows installation compared to an upgrade?
3. What steps should you take to ensure data safety before installing Windows?
4. How can the installation process be made faster or more efficient?

# LAB ASSESSMENT 3: INTRODUCTION TO GOOGLE DOCS

**Objective**

To introduce students to Google Docs, enabling them to create, edit, and collaborate on documents online.

**Apparatus/Materials**

1. A computer, laptop, or smartphone with internet access.
2. An active Google account.
3. Access to a web browser or the Google Docs app installed on your device.

**Theory**

Google Docs is a cloud-based word processing tool that allows users to create, edit, and share documents in real time. It facilitates collaboration by enabling multiple users to work simultaneously on a document, with changes saved automatically. Its features make it an essential tool for modern online education and professional work.

**Procedure**

***Part 1: Accessing Google Docs on a Web Browser***

1. Open a Web Browser: Launch Google Chrome or any other browser.
2. Sign In:
   * Go to <https://docs.google.com>.
   * Sign in with your Google account credentials.
3. Create a New Document:
   * Click the **"+"** icon labeled **Blank** to start a new document.
4. Explore the Toolbar:
   * Familiarize yourself with formatting tools such as bold, italic, font styles, and alignment.
5. Share the Document**:**
   * Click the **Share** button in the top-right corner.
   * Enter an email address to invite collaborators and set their permissions (viewer, commenter, editor).

***Part 2: Using Google Docs on a Mobile Device***

1. Install the Google Docs App:
   * Download the app from the Google Play Store (Android) or App Store (iOS).
2. Sign In:
   * Open the app and log in with your Google account credentials.
3. Create and Edit Documents:
   * Tap the **"+"** icon to start a new document.
   * Explore editing tools like text formatting, inserting images, and sharing options.

**Lab Task**

* Create a Google Doc, format the text with various styles, and share it with your instructor or classmates for collaborati

**Conclusion**

This lab introduces students to Google Docs, emphasizing its features for creating, editing, and collaborating on documents online. It ensures students can use this tool effectively for academic and professional purposes.

**Questions**

1. How does Google Docs support collaboration in real time?
2. What are the advantages of using Google Docs over traditional word processors?
3. How can permissions (viewer, commenter, editor) affect collaboration?
4. Suggest other cloud-based tools similar to Google Docs and their benefits.

# LAB ASSESSMENT 4: INTRODUCTION TO UBUNTU

**Objective:**

To introduce students to Ubuntu Linux, enabling them to navigate the user interface, access the terminal, and perform basic commands.

**Apparatus/Materials**

* A computer or laptop with Ubuntu Linux installed (or access to a live USB/DVD).
* Basic knowledge of computer operation.
* Internet connection (optional for updates).

**Theory**

Ubuntu is a free and open-source Linux-based operating system known for its simplicity, security, and stability. It provides a graphical interface for general use and a command-line interface for advanced functionality. Understanding Ubuntu allows users to explore Linux features and execute basic tasks, making it ideal for education, development, and everyday use.

**Procedure**

***Part 1: Exploring the Ubuntu User Interface***

1. ***Log In to Ubuntu:***
   * Boot into the Ubuntu operating system and log in with your username and password.
2. ***Explore the Desktop Environment:***
   * Familiarize yourself with the Activities Overview, taskbar, and workspace switcher.
3. ***Access System Settings:***
   * Open the Settings menu to explore system customization options like display, network, and appearance.
4. ***Launch Applications:***
   * Use the application launcher to open apps like Firefox, LibreOffice, and Files.

***Part 2: Using the Terminal***

1. ***Open the Terminal:***
   * Click the Terminal icon from the launcher or press Ctrl+Alt+T.
2. ***Execute Basic Commands:***
   * Try the following commands:
     + ls – List files and directories.
     + pwd – Display the current directory.
     + cd – Change directory.
     + mkdir <folder\_name> – Create a new directory.
     + sudo apt update – Update package lists.
3. ***Create a File:***
   * Use touch <file\_name> to create a new file in the current directory.
4. ***Display File Content:***
   * Use cat <file\_name> to display the contents of a file.

***Part 3: Customizing Ubuntu***

1. ***Change Wallpaper:***
   * Right-click on the desktop, select Change Background, and select a new wallpaper.
2. ***Install Applications:***
   * Use the Ubuntu Software Center or terminal (sudo apt install <app\_name>) to install new applications.

**Lab Task**

* Perform basic file and directory operations in Ubuntu Linux using the terminal and explore the graphical interface.

**Conclusion**

This lab introduces students to the basics of Ubuntu Linux, covering its graphical interface and essential terminal commands. It provides a foundation for understanding Linux-based systems for educational and professional purposes.

**Questions**

1. What is the purpose of the pwd command in the terminal?
2. How can you create a new folder in Ubuntu using the terminal? Provide the command.
3. Describe the steps to change the desktop wallpaper in Ubuntu.
4. Explain how to install an application using the terminal. Include an example.
5. What is the function of the sudo apt update command? Why is it important?

# LAB ASSESSMENT 5: INTRODUCTION TO LATEX

**Objective:**

To introduce students to LaTeX, a typesetting system used for creating professional-quality documents. This lab will enable students to understand the basic structure of a LaTeX document, compile it, and explore common commands for text formatting and creating mathematical expressions.

**Apparatus/Materials**

A computer with a LaTeX editor installed (e.g., TeXworks, Overleaf, or Texmaker).

Basic knowledge of text editing.

Internet connection (optional for cloud-based editors or installing packages).

**Theory**

LaTeX is a high-quality typesetting system commonly used for technical and scientific documentation. Unlike traditional word processors, LaTeX focuses on the content and structure of documents, leaving formatting to predefined styles. It excels in handling complex layouts, mathematical expressions, and references, making it a preferred tool for researchers and academicians.

**Procedure**

***Part 1: Setting Up LaTeX***

1. ***Install LaTeX:***
   * Install a LaTeX distribution like TeX Live (Linux), MikTeX (Windows), or MacTeX (Mac).
   * Download and set up an editor such as TeXworks or Overleaf (online).
2. ***Create a New Document:***
   * Open the editor and start a new .tex file.
   * Begin the document with the following template:

\documentclass{article}

\begin{document}

Hello, LaTeX!

\end{document}

***3.Compile the Document:***

* + Compile the code using pdf latex or the compile button in the editor to generate a PDF file.

***Part 2: Writing Text and Formatting***

***Add Sections and Subsections:***

* Use the following commands:

**latex**

\section{Introduction}

\subsection{Background}

***Format Text:***

* Bold text: \textbf{Bold Text}
* Italics: \textit{Italic Text}
* Underline: \underline{Underlined Text}

**Lab Task**

Create a basic document in LaTeX with the following features:

* Title, author, and date.
* At least one section and one subsection.
* A mathematical expression.
* A table or image.
* A bibliography entry

**Conclusion**

This lab introduces students to LaTeX, covering its basic syntax and commands for creating structured, professional documents. Students gain skills in formatting text, writing mathematical expressions, and managing references, building a foundation for academic and professional documentation.

**Questions:**

* How does Google Slides facilitate real-time collaboration?
* What are the benefits of using Google Slides over traditional presentation software?
* How do different permissions (viewer, commenter, editor) impact collaborative presentations?

# LAB ASSESSMENT 6: INTRODUCTION TO GOOGLE SLIDES

**Objective**

To introduce students to Google Slides, enabling them to create, edit, and present slideshows using its features for effective presentations.

**Apparatus/Materials**

* A computer or laptop with a web browser (e.g., Google Chrome).
* Google Account for accessing Google Slides.
* Internet connection.

**Theory**

Google Slides is a cloud-based presentation software that is part of the Google Workspace suite. It allows users to create, share, and collaborate on presentations in real time. Its intuitive interface and wide range of customization tools make it an ideal choice for educational, business, and personal use. Google Slides can be accessed directly through a web browser, eliminating the need for additional software installations.

**Procedure**

***Part 1: Accessing the Google Slides***

* Open the web browser and login to the google slides through your account.
* Familiarize yourself with the Google Slides homepage, including the template gallery and recent files section.

***Part 2: Creating and Customizing a Presentation***

***Start a New Presentation:***

* Click on the “Blank” presentation or choose a template from the gallery.

***Add and Format Slides:***

* Use the “+” button or *Ctrl + M* to add a new slide.
* Right-click to duplicate, delete, or rearrange slides in the slide panel.

***Customize Slide Design:***

* Use the Themes panel to select or change slide themes.
* Customize the background by selecting *Background > Change Background*.

**Part 3: Working with Content**

***Add Text and Format:***

* Click on text boxes to type content.
* Use the toolbar to format text (font, size, color, alignment).

***Insert Media:***

* Insert images (*Insert > Image > Upload from computer / Search the web*).
* Embed videos (*Insert > Video*).

***Add Shapes and Charts:***

* Use the *Insert > Shape* or *Insert > Chart* menu to add visual elements.

**Part 4: Collaboration and Presentation**

1. ***Share and Collaborate:***
   * Click *Share* to add collaborators by email or share a link.
   * Set permissions (viewer, commenter, editor).
2. ***Present the Slides:***
   * Click the *Present* button to start the slideshow.
   * Use the speaker notes section for presentation guidance.

**Lab Task**

* Create a five-slide presentation on any topic.
* Include at least one image, one chart, and one video.
* Format the text, apply a theme, and customize the background.
* Share the presentation link with the instructor.

**Conclusion**

This lab introduces students to Google Slides, focusing on its core functionalities and collaborative features. Mastering Google Slides empowers students to create professional presentations for academic and workplace scenarios.

**Questions**

* How does Google Slides facilitate real-time collaboration?
* What are the benefits of using Google Slides over traditional presentation software
* How do different permissions (viewer, commenter, editor ) impact collaborative presentations?
* Suggest other cloud-based tools similar to Google Slides and their advantages.

LAB ASSESSMENT 7: INTRODUCTION TO SPREADSHEETS

**Objective**

To familiarize students with spreadsheets, enabling them to organize, analyze, and visualize data effectively using spreadsheet tools such as Microsoft Excel, Google Sheets, or LibreOffice Calc.

**Apparatus/Materials**

* A computer, laptop, or smartphone with internet access.
* A spreadsheet application (e.g., Microsoft Excel, Google Sheets, LibreOffice Calc).
* An active account for online spreadsheet tools (if applicable).

**Theory**

Spreadsheets are versatile tools used for data organization, analysis, and visualization. They support various features such as formulas, functions, charts, and conditional formatting to facilitate efficient data handling. Collaborative spreadsheet platforms like Google Sheets allow real-time teamwork, enabling multiple users to edit and view the same sheet simultaneously. These tools are widely utilized in academic, professional, and personal contexts to manage data systematically and derive insights effectively.

**Procedure**

**Part 1: Using Google Sheets on a Web Browser**

1. Open a web browser such as Google Chrome.
2. Go to Google Sheets and sign in with your Google account.
3. Click on the "+" icon labeled *Blank* to create a new spreadsheet.
4. Familiarize yourself with the basic components such as rows, columns, cells, and tabs.
5. Perform the following tasks:
   * Enter sample data into cells.
   * Use basic formulas (e.g., SUM, AVERAGE).
   * Apply cell formatting such as bold text, background colors, and borders.
   * Insert a chart (e.g., bar chart or pie chart) based on the entered data.
6. Click the *Share* button in the top-right corner.
7. Invite collaborators by entering their email addresses and set permissions (viewer, commenter, editor).

**Part 2: Using Microsoft Excel on a Desktop**

1. Open Microsoft Excel on your computer.
2. Start a new workbook.
3. Perform the following tasks:
   * Create a simple table with data.
   * Apply formulas to calculate totals, averages, or percentages.
   * Use conditional formatting to highlight specific cells based on values.
   * Insert a chart and adjust its appearance.
4. Save the file locally or on a cloud platform (e.g., OneDrive).

**Part 3: Using Spreadsheets on a Mobile Device**

1. Download and install a spreadsheet app like Google Sheets or Microsoft Excel from the app store.
2. Open the app and log in with your account credentials.
3. Create a new spreadsheet and enter data.
4. Experiment with basic editing tools such as formulas, formatting, and sorting.
5. Save and share the file for collaboration.

**Lab Task**

Create a spreadsheet document with the following:

* A data table with at least five rows and three columns.
* Use at least two formulas (e.g., SUM, AVERAGE).
* Apply conditional formatting to highlight specific data.
* Insert a chart to visualize the data.
* Share the spreadsheet with your instructor or classmates for collaboration

**Conclusion**

This lab introduces students to the fundamental features of spreadsheet tools, highlighting their capabilities for data organization, analysis, and visualization. Students will gain hands-on experience with spreadsheet applications, preparing them for academic and professional tasks that require efficient data handling.

**Questions**

1. What are the advantages of using spreadsheets for data management?
2. How does real-time collaboration in Google Sheets improve productivity?
3. Compare and contrast Google Sheets and Microsoft Excel in terms of features and accessibility.
4. Explain the importance of formulas and functions in spreadsheets.
5. Suggest other tools or software for data analysis and their unique benefits.

# LAB ASSESSMENT 8: BASICS OF HTML

**Objective**

To familiarize students with the foundational concepts of HTML, enabling them to create, structure, and format web pages effectively.

**Apparatus/Materials**

* A computer, laptop, or tablet with internet access.
* A text editor such as Notepad++ or Visual Studio Code.
* A web browser (e.g., Google Chrome, Mozilla Firefox).

**Theory**

HTML (HyperText Markup Language) is the standard language used to structure content on the web. It uses a system of tags and attributes to define elements such as headings, paragraphs, links, images, and more. Understanding HTML is essential for web development as it forms the backbone of web pages, working seamlessly with CSS and JavaScript to create dynamic and visually appealing websites.

**Procedure**

**Part 1: Setting Up the Development Environment**

1. Open a text editor (e.g., Notepad++ or Visual Studio Code).
2. Create a new file and save it with the extension .html.
3. Open a web browser for previewing your code.

**Part 2: Writing Basic HTML Code**

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title>My First Web Page</title>  </head>  <body>  <h1>Musawar Ali Shah</h1>  <p>This is a paragraph of text on my first web page.</p>  <ul>  <li>Item 1</li>  <li>Item 2</li>  <li>Item 3</li>  </ul>  <img src="image.jpg" alt="A sample image">  <a href="https://example.com">Visit Example</a>  </body>  </html> |

Create an HTML webpage with the following elements:

1. A title and a heading describing the purpose of the page.
2. A paragraph introducing the topic of your choice.
3. An unordered list of three items.
4. A table containing at least two rows and two columns.
5. A form with input fields for name and email, and a submit button.

Save the file and open it in a web browser to verify functionality.

**Lab Task**

Create a spreadsheet document with the following:

* A data table with at least five rows and three columns.
* Use at least two formulas (e.g., SUM, AVERAGE).
* Apply conditional formatting to highlight specific data.
* Insert a chart to visualize the data.
* Share the spreadsheet with your instructor or classmates for collaboration

**Conclusion**

This lab introduces students to the foundational concepts of HTML, emphasizing its role in web development. Students will gain hands-on experience in structuring and formatting content using various HTML tags, laying the groundwork for further exploration of web technologies.

**Questions**

1. What is the purpose of the <head> and <body> sections in an HTML document?
2. How does the <img> tag enhance a webpage’s content?
3. What is the significance of the alt attribute in the <img> tag?
4. Describe the role of forms in collecting user data on a webpage.
5. List other technologies commonly used alongside HTML in web development.

# LAB ASSESSMENT 9: LOCALHOST SERVER

**Objective**

To guide students through the process of setting up a localhost server using Apache2 on a Linux system and verifying its functionality using the ping command.

**Apparatus/Materials**

* A computer or laptop running a Linux operating system.
* A terminal application.
* Internet access to download and install packages.

**Theory**

A localhost server is a server that runs locally on your computer, providing an environment to test and develop websites or web applications without requiring an external server. Apache2 is one of the most widely used HTTP servers for this purpose. By configuring a local server, users can host web pages accessible only on their network or locally on the machine.

**Procedure**

**Step 1: Update System Packages**

* Open a terminal.
* Run the following command to update the package index:
* **sudo apt update**

**Step 2: Install Apache2**

* Install Apache2 using the package manager:
* **sudo apt install apache2**

**Step 3: Start and Enable Apache2**

1. Start the Apache2 service:

* **sudo systemctl start apache2**

1. Enable Apache2 to start on boot:

* **sudo systemctl enable apache2**

**Step 4: Verify Apache2 is Working**

1. Check if Apache2 is active by running:

**sudo systemctl status apache2**

**Step 5: Using the ping Command for Verification**

1. Open the terminal.
2. Run the ping command to test connectivity:

**ping localhost**

1. Observe the responses to confirm connectivity.

**Step 6: Use Your IP Address to Host the Server**

1. Open the terminal.
2. Run the following command:

**ping <your-ip-address>**

1. To test the server, open a web browser and type:

**http://<your-ip-address>**

1. Share this link with devices connected to the same network for access.

**Lab Task**

* Configure a localhost server on your Linux system following the steps above.
* Verify its functionality by:
* Checking Apache2 status.
* Testing connectivity with ping.
* Accessing the server using your IP address in a web browser.

Document each step with screenshots and describe the output for each command.

**Conclusion**

This lab introduces students to setting up and verifying a local server using Apache2. By completing the lab, students gain practical knowledge of configuring and managing a localhost server for development and testing purposes.

**Questions**

* What is the purpose of using a localhost server during web development?
* How does Apache2 support the development and testing of web applications?
* What is the significance of enabling Apache2 to start on boot?
* How can the ping command help verify server functionality?
* Describe scenarios where a localhost server might be used instead of an external server.

# LAB ASSESSMENT 10: CREATING A WEB PAGE USING HTML

**Objective**

To guide students in creating a structured and well-formatted webpage using HTML, demonstrating essential web development concepts such as headings, lists, and tables.

**Apparatus/Materials**

* A computer or laptop with a text editor (e.g., Notepad++, Visual Studio Code).
* A web browser for previewing the HTML file.

**Theory**

HTML (HyperText Markup Language) is the backbone of web development, providing the structure and layout for web pages. This lab focuses on using semantic HTML tags to create a webpage with hierarchical headings, ordered lists, and tables. Understanding these elements helps in building clear and user-friendly websites.

**Procedure**

***Step 1: Setting Up the HTML Document***

1. Open a text editor and create a new file.
2. Save the file with the extension .html (e.g., portfolio.html).
3. Write the basic structure of the HTML document:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Portfolio</title>  </head>  <body>  <!-- Content goes here -->  </body>  </html> |

***Step 2: Adding Headings and Personal Details***

1. Use headings (<h1>, <h2>, <h3>) to display your name, section, and roll number

|  |
| --- |
| <h1>John Doe</h1>  <h2>BCS-1E Section</h2>  <h3>24P-1234</h3> |

**Step 3: Creating an Ordered List**

1. Use the <ol> tag to create an ordered list of subjects.
2. Each subject is listed as a <li> (list item):

|  |
| --- |
| <ol type="1">  <li>Calculus</li>  <li>ICT</li>  <li>Programming Fundamentals</li>  </ol> |

**Step 4: Building a Table for Marks**

1. Use the <table> tag to create a table for displaying subjects and their respective marks.
2. Add rows with <tr>, headers with <th>, and data with <td>

|  |
| --- |
| <table border="1">  <tr>  <th>Subject</th>  <th>Obtained Marks</th>  <th>Max Marks</th>  </tr>  <tr>  <td>Calculus</td>  <td>79</td>  <td>100</td>  </tr>  <tr>  <td>ICT</td>  <td>90</td>  <td>100</td>  </tr>  <tr>  <td>English Lab</td>  <td>95</td>  <td>100</td>  </tr>  </table> |

**Step 5: Saving and Testing**

1. Save the file.
2. Open it in a web browser to verify the output.

**Lab Task**

* Create an HTML webpage following the procedure above.
* Customize the content with your name, section, roll number, and subject marks.
* Save the file and preview it in a web browser.

**Conclusion**

This lab assessment introduces students to the basics of creating a webpage using HTML. By following the steps, students learn to structure content with headings, lists, and tables, gaining essential skills for web development.

**Questions**

* What is the purpose of the <!DOCTYPE html> declaration in an HTML document?
* How does the <table> tag help in organizing data on a webpage?
* What is the difference between <th> and <td> in an HTML table?
* How does the type attribute in the <ol> tag affect the appearance of the list?
* Why is it important to use semantic HTML tags?

# LAB ASSESSMENT 11: BASICS OF JAVASCRIPT

**Objective**

To familiarize students with the fundamental concepts of JavaScript, including alerts, variables, functions, and array manipulation, for building interactive and dynamic web applications.

**Apparatus/Materials**

* A computer or laptop with a text editor (e.g., Visual Studio Code, Notepad++).
* A web browser to execute and test the JavaScript code

**Theory**

JavaScript is a versatile programming language used to add interactivity to web pages. This lab demonstrates essential JavaScript concepts, such as variables, arithmetic operations, functions, arrays, and loops. These techniques are the foundation for creating dynamic and responsive web applications.

**Procedure**

**Step 1: Setting Up the HTML Document**

Open a text editor and create a new file.

Save the file with a .html extension (e.g., javascript\_basics.html).

Write the basic structure of an HTML document:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Basics of JavaScript</title>  </head>  <body>  <script>  // JavaScript code goes here  </script>  </body>  </html> |

**Step 2: Displaying Alerts**

1. Use the alert method to display messages:

**alert("Hello world");**

**Step 3: Working with Variables and Arithmetic Operations**

1. Declare and initialize a variable using var:

|  |
| --- |
| var x = 10;  alert(x);  alert(x **+** 10); |

**Step 4: Creating and Using Functions**

1. Define a function and use local variables:

|  |
| --- |
| function displayResult() {  let y = 10;  alert(y + 20);  }  displayResult(); |

**Step 5: Using Arrays and Loops**

1. Create an array and iterate over its elements using a for loop:

|  |
| --- |
| let array = [1, 2, 3, 4, 5];  for (let i = 0; i < array.length; i++) {  alert(array[i]);  i += 3;  } |

**Step 6: Saving and Testing**

1. Save the file and open it in a web browser.
2. Verify the output of each script step as alerts appear.

**Lab Task**

* Write a JavaScript program that:
* Displays an introductory alert message.
* Performs arithmetic operations on variables and shows the results.
* Defines and calls a function that manipulates and displays data.
* Iterates over an array using a loop and displays its elements.
* Test the code in a web browser and document your observations.

**Conclusion**

This lab introduces students to the basics of JavaScript, emphasizing the use of alerts, variables, functions, and arrays. By completing the lab, students gain foundational skills to create interactive web applications.

**Questions**

* What is the purpose of the alert method in JavaScript?
* Explain the difference between var and let for declaring variables.
* How does the for loop in JavaScript facilitate array iteration?
* What is the scope of a variable declared inside a function using let?
* Suggest a practical application for using arrays and loops in a web application.

# LAB ASSESSMENT 12: BUTTONS AND FORMS

**Objective**

To explore JavaScript's integration with HTML forms and buttons, focusing on event handling, user input validation, and dynamic feedback mechanisms.

**Apparatus/Materials**

* A computer or laptop with a text editor (e.g., Visual Studio Code, Notepad++).
* A web browser for testing the HTML and JavaScript code.

**Theory**

HTML forms provide the structure for user input, while JavaScript handles interactions and validations. By linking form elements to JavaScript functions, developers can create responsive and interactive web applications. Key concepts include DOM manipulation, event handling, and conditional logic.

**Procedure**

**Step 1: Setting Up the HTML Document**

1. Open a text editor and create a new file.
2. Save the file with an .html extension (e.g., buttons\_forms.html).
3. Write the basic structure of an HTML document:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Buttons and Forms</title>  </head>  <body>  <center>  <form>  <p>Username</p>  <input type="text" id="username">  <p>Password</p>  <input type="password" id="password">  <p>Submit</p>  <button type="button" onclick="validateForm()">Submit</button>  </form>  </center>  <script>  function validateForm() {  let user = document.getElementById("username").value;  let pass = document.getElementById("password").value;  if (user === "fast" && pass === "123") {  alert("Login Successful");  } else {  alert("Invalid Username or Password");  }  }  </script>  </body>  </html> |

**Step 2: Testing the Form and Script**

1. Open the file in a web browser.
2. Enter a username and password, then click Submit.
3. Verify the feedback messages based on input validation.

**Lab Task**

Write a JavaScript program that:

* Accepts user input for username and password via a form.
* Validates the input against predefined credentials.
* Displays an alert with appropriate feedback based on validation results.
* Ensures password masking through the use of the password input type

**Conclusion**

This lab demonstrates the integration of JavaScript with HTML forms and buttons, highlighting event-driven programming and user interaction. By completing this lab, students gain practical experience in creating dynamic web applications.

**Questions**

* What method is used to retrieve the value of an input field in JavaScript?
* How does the onclick attribute link a button to a JavaScript function?
* Why is it important to mask passwords in form inputs?
* Explain the use of if-else statements in form validation.
* Suggest a real-world application of form validation in web development.

# Lab Assessment 13: Hyperlink and page linking

**Objective**

To explore the fundamental concepts of creating and using hyperlinks and page linking in HTML, enabling navigation between web pages and enhancing website functionality.

**Apparatus/Materials**

* A computer or laptop with a text editor (e.g., Visual Studio Code, Notepad++).
* A web browser for viewing and testing the HTML code.

**Theory**

Hyperlinks are one of the most essential features of HTML, allowing users to navigate between pages or sections within a web page. Using the <a> tag, developers can create links to internal pages, external websites, email addresses, or specific locations within the same document. Page linking is crucial for creating a well-structured, user-friendly website.

**Procedure**

**Step 1: Setting Up the HTML Document**

1. Open a text editor and create a new file.

2. Save the file with a .html extension (e.g., hyperlinks\_basics.html).

3. Write the basic structure of an HTML document:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Hyperlinks and Page Linking</title>  </head>  <body>  <!-- Hyperlink code will go here -->  </body>  </html> |

**Step 2: Creating Basic Hyperlinks:**

1. Use the <a> tag to link to an external website:

<a href="https://www.example.com" target="\_blank">Visit Example Website</a>

2. Add a link to another local HTML file (e.g., page2.html)

<a href="page2.html">Go to Page 2</a>

**Step 3: Linking to Specific Sections Within the Same Page**

1. Define an anchor point with an id attribute:

<h2 id="section1">Section 1</h2>

<p>This is the content of Section 1.</p>

2. Create a hyperlink to navigate to the anchor point:

<a href="#section1">Jump to Section 1</a>

**Step 4: Linking to an Email Address**

1.Use the mailto: protocol to create an email link:

<a href="mailto:example@example.com">Send an Email</a>

**Step 6: Saving and Testing**

1. Save the HTML file and open it in a web browser.

2. Click on the hyperlinks to verify navigation functionality.

**Lab Task**

* Create an HTML document that includes:
* A hyperlink to an external website.
* A link to another local HTML file.
* An anchor link to a section within the same page.
* An email link using the mailto: protocol.
* Styled hyperlinks using inline CSS.
* Test the document in a web browser and document the results.

**Conclusion**

This lab introduces students to the fundamental concepts of hyperlinks and page linking in HTML. By completing the lab, students gain the skills needed to create navigable and user-friendly web pages.

**Questions**

1. What is the purpose of the target="\_blank" attribute in a hyperlink?

2. Explain the difference between an external link and an internal link.

3. How do anchor links enhance user navigation on a single page?

4. What is the role of the mailto: protocol in hyperlink creation?

5. Suggest a practical application for anchor links in a large web document.