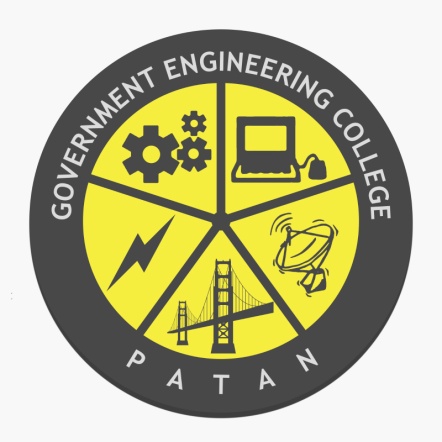
**A Laboratory Manual for**

Web Programming

**(3160713)**

**B.E. Semester6**

**(Computer Science and Engineering)**

****



**Directorate of Technical Education,**

**Gandhinagar, Gujarat**

**Government Engineering college, Patan**

**Certificate**

This is to certify that Mr./Ms. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ Enrollment No. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Of B.E. Semester \_\_\_\_\_ Computer Science and Engineering of this Institute (GTU Code: 022) has satisfactorily completed the Practical work for the subject **Web Programming (3160713)** for the academic year 2023-24.

Place: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

**Name and Sign of Faculty member**

**Head of the Department**

**Preface**

Main motto of any laboratory/practical/field work is for enhancing required skills as well as creating ability amongst students to solve real time problem by developing relevant competencies in psychomotor domain. By keeping in view, GTU has designed competency focused outcome-based curriculum for engineering degree programs where sufficient weightage is given to practical work. It shows importance of enhancement of skills amongst the students and it pays attention to utilize every second of time allotted for practical amongst students, instructors and faculty members to achieve relevant outcomes by performing the experiments rather than having merely study type experiments. It is must for effective implementation of competency focused outcome-based curriculum that every practical is keenly designed to serve as a tool to develop and enhance relevant competency required by the various industry among every student. These psychomotor skills are very difficult to develop through traditional chalk and board content delivery method in the classroom. Accordingly, this lab manual is designed to focus on the industry defined relevant outcomes, rather than old practice of conducting practical to prove concept and theory.

By using this lab manual students can go through the relevant theory and procedure in advance before the actual performance which creates an interest and students can have basic idea prior to performance. This in turn enhances pre-determined outcomes amongst students. Each experiment in this manual begins with competency, industry relevant skills, course outcomes as well as practical outcomes (objectives). The students will also achieve safety and necessary precautions to be taken while performing practical.

This manual also provides guidelines to faculty members to facilitate student centric lab activities through each experiment by arranging and managing necessary resources in order that the students follow the procedures with required safety and necessary precautions to achieve the outcomes. It also gives an idea that how students will be assessed by providing rubrics.

In the era of digitization, the demand of Internet based applications is increasing day by day. Web Development is one of the required skills for IT Engineer. This focuses on front-end and back-end design. After learning Web Development students can advance their career in the field of web development.

Utmost care has been taken while preparing this lab manual however always there is chances of improvement. Therefore, we welcome constructive suggestions for improvement and removal of errors if any.

**Practical – Course Outcome matrix**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes (COs):**  CO-1: Use the various HTML tags with appropriate styles to display the various types of contents effectively.  CO-2: Develop the dynamic web pages using HTML, CSS and JavaScript applying web design principles to make pages effective.  CO-3: Develop the server side PHP scripts using various features for creating customized web services.  CO-4: Write the server side scripts for designing web based services with database connectivity.  CO-5: Develop a web application using advanced web programming features including AJAX and JQuery using concepts of Web API. | | | | | | |
| **Sr. No.** | **Objective(s) of Experiment** | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1. | Create your resume using HTML (Suggested sections of resume are Personal Information, Educational Information, Professional Skills, Experience, Achievements, Hobbies), Experiment with text, colors, link and lists. | √ |  |  |  |  |
| 2. | Create your class time table using table tag, experiment with rowspan, colspan, cellspacing and cellpadding attributes. | √ |  |  |  |  |
| 3. | Design static web pages for your college containing a description of the courses, departments, faculties, library etc. Provide links for navigation among pages. | √ |  |  |  |  |
| 4. | Create User Registration Form in HTML (Suggested to use fields like Name, Date of Birth, Gender, Email Id, Mobile No.,Address, State , Education , Image Upload etc) using textbox, textarea, checkbox, radio button, select box, button, file upload etc. | √ |  |  |  |  |
| 5. | Create two web pages, one contains audios and other page contains videos (using HTML5 audio and video tags). Also provide link for navigation between pages. | √ |  |  |  |  |
| 6. | Create a web page using frame. Divide the page into two parts with Navigation links on left hand side of page (width=20%) and content page on right hand side of page (width = 80%). On clicking the navigation Links corresponding content must be shown on the right-hand side. | √ |  |  |  |  |
| 7. | Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS). | √ |  |  |  |  |
| 8. | Use Inline CSS to format your resume that you created in practical no 01. | √ |  |  |  |  |
| 9. | Use External, Internal, Inline CSS to format College Web site that you created in Practical No.03 | √ |  |  |  |  |
| 10. | Develop a java script to display today’s date. |  | √ |  |  |  |
| 11. | Develop simple calculator for addition, subtraction, multiplication and division operation using java script. |  | √ |  |  |  |
| 12. | Write a java script code to combine and display the information in textbox when the button is clicked use registration page that you created in Practical No.4. |  | √ |  |  |  |
| 13. | Use JavaScript to Implement validation in Practical No.4. |  | √ |  |  |  |
| 14. | Write a PHP program to check if number is prime or not. |  |  | √ |  |  |
| 15. | Use Registration Form from practical number 4 to store user registration details in MySql database. On submission next page displays all registration data in in html table using php. Also provide feature to update and delete the registration data. |  |  |  | √ |  |
| 16. | Write a PHP script for user authentication using PHP-MYSQL. Use session for storing username. |  |  |  | √ |  |
| 17. | Using AJAX Create visual search feature to search using name for practical number 15 which list name, mobile number and email id of matching users. |  |  |  |  | √ |
| 18. | Create a REST API using php. |  |  |  |  | √ |
| 19. | Create an Image slider using jQuery. |  |  |  |  | √ |
| 20 | Cookie Example |  |  |  |  | √ |

**Industry Relevant Skills**

The following industry relevant competency are expected to be developed in the student by undertaking the practical work of this laboratory.

1. HTML/CSS Skills: HTML is used extensively by web developers to build web pages. CSS is used to implement different fonts, colors and layouts in the design of a website.
2. Javascript Skills: Java Script is used for creating interactive web pages to improve the user experience.
3. PHP/MySQLSkills:PHP/MySQL is used extensively by web developers to create fully functional dynamic web applications.
4. REST API, AJAX, JQuery Skills: RESTAPI,AJAX,JQuery are advanced asynchronous web communication mechanisms and used by web developers for building highly interactive webpages.

**Guidelines forFaculty members**

1. Teacher should provide the guideline with demonstration of practical to the students with all features.
2. Teacher shall explain basic concepts/theory related to the experiment to

the students before starting of each practical

1. Involve all the students in performance of each experiment.
2. Teacher is expected to share the skills and competencies to be developed in the students and ensure that the respective skills and competencies are developed in the students after the completion of the experimentation.
3. Teachers should give opportunity to students for hands-on experience after the demonstration.
4. Teacher may provide additional knowledge and skills to the students even though not covered in the manual but are expected from the students by concerned industry.
5. Give practical assignment and assess the performance of students based on task assigned to check whether it is as per the instructions or not.
6. Teacher is expected to refer complete curriculum of the course and follow the guidelines for implementation.

**Instructions for Students**

1. Students have to write answers / solutions of QUIZ in separate file page. The quiz of corresponding practical must be attached just behind each practical.
2. Students are expected to carefully listen to all the theory classes delivered by the faculty members and understand the COs, content of the course, teaching and examination scheme, skill set to be developed etc.
3. Students shall organize the work in the group and make record of all observations.
4. Students shall develop maintenance skill as expected by industries.
5. Student shall attempt to develop related hand-on skills and build confidence.
6. Student shall develop the habits of evolving more ideas, innovations, skills etc. apart from those included in scope of manual.
7. Student shall refer technical magazines and data books.
8. Student should develop a habit of submitting the experimentation work as per the schedule and s/he should be well preparedfor the same.

**Sample Rubrics for Practical Evaluation**

|  |  |  |  |
| --- | --- | --- | --- |
| Understanding of Problem  (3 marks) | Implementation of Problem  (4 marks) | Presentation and report writing  (3 marks) | Total  (10 marks) |
|  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Understanding of Problem | Excellent understanding of problem and relevance with the theory clearly understood. | **03** | Moderate level understanding of problem and relevance with the theory clearly understood. | **02** | Problem not understood and can't establish the relation with the theory. | **01** |
| Implementation of Problem | Efficient implementation with proper naming convention and understanding | **04** | Moderate level of implementation. Poor naming convention. | **03** | Partial implementation with poor understanding. | **01 to 02** |
| Presentation and report writing | Unique documentation (not copied from other sources) of given problem with proper formatting and language. | **03** | Ordinary documentation of given problem with proper formatting and language | **02** | Weak documentation of given problem without proper formatting and language | **01 to 02** |

**Index**

**(Progressive Assessment Sheet)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Sr. No. | Objective(s) of Experiment | Page No. | Date of performance | Date of submission | Assessment  Marks | Sign. of  Teacher with date | Remarks |
| 1. | Create your resume using HTML (Suggested sections of resume are Personal Information, Educational Information, Professional Skills, Experience, Achievements, Hobbies), Experiment with text, colors, link and lists. |  |  |  |  |  |  |
| 2. | Create your class time table using table tag, experiment with rowspan, colspan, cellspacing and cellpadding attributes. |  |  |  |  |  |  |
| 3. | Design static web pages for your college containing a description of the courses, departments, faculties, library etc. Provide links for navigation among pages. |  |  |  |  |  |  |
| 4. | Create User Registration Form in HTML (Suggested to use fields like Name, Date of Birth, Gender, Email Id, Mobile No.,Address, State , Education , Image Upload etc.) using textbox, textarea, checkbox, radio button, select box, button, file upload etc. |  |  |  |  |  |  |
| 5. | Create two web pages, one contains audios and other page contains videos (using HTML5 audio and video tags). Also provide link for navigation between pages. |  |  |  |  |  |  |
| 6. | Create a web page using frame. Divide the page into two parts with Navigation links on left hand side of page (width=20%) and content page on right hand side of page (width = 80%). On clicking the navigation Links corresponding content must be shown on the right-hand side. |  |  |  |  |  |  |
| 7. | Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS). |  |  |  |  |  |  |
| 8. | Use Inline CSS to format your resume that you created in practical no 01. |  |  |  |  |  |  |
| 9. | Use External, Internal, Inline CSS to format College Web site that you created in Practical No.03 |  |  |  |  |  |  |
| 10. | Develop a java script to display today’s date. |  |  |  |  |  |  |
| 11. | Develop simple calculator for addition, subtraction, multiplication and division operation using java script. |  |  |  |  |  |  |
| 12. | Write a java script code to combine and display the information in textbox when the button is clicked use registration page that you created in Practical No.4. |  |  |  |  |  |  |
| 13. | Use JavaScript to Implement validation in Practical No.4. |  |  |  |  |  |  |
| 14. | Write a PHP program to check if number is prime or not. |  |  |  |  |  |  |
| 15. | Use Registration Form from practical number 4 to store user registration details in MySql database. On submission next page displays all registration data in in html table using php. Also provide feature to update and delete the registration data. |  |  |  |  |  |  |
| 16. | Write a PHP script for user authentication using PHP-MYSQL. Use session for storing username. |  |  |  |  |  |  |
| 17. | Using AJAX Create visual search feature to search using name for practical number 15 which list name, mobile number and email id of matching users. |  |  |  |  |  |  |
| 18. | Create a REST API using php. |  |  |  |  |  |  |
| 19. | Create an Image slider using jQuery. |  |  |  |  |  |  |
| 20 | Cookie Example |  |  |  |  |  |  |
| Total | | | | |  |  |  |

**Experiment No: 1**

**Create your resume using HTML (Suggested sections of resume are Personal Information, Educational Information, Professional Skills, Experience, Achievements, Hobbies), Experiment with text, colors, link and lists.**

**Date:**

**Competency and Practical Skills:**

**Relevant CO:** 1

**Objectives:**

1. To understand HTML Page Structure.
2. To understand how to use HTML tag attributes.

**Theory:**

**HTML:**

* HTML stands for Hypertext Markup Language
* It is used to display the document in the web browser
* Hypertext is simply a piece of text that works as a link
* Markup Language is a way of writing layout information within documents

**HTML Document Structure**

* HTML Document consists of three main parts
  + DOCTYPE declaration
  + <head> section
  + <body> section

<!DOCTYPE html>

<html>

<head>

<title>Page Title</title>

</head>

<body>

<h1> Hello World </h1>

</body>

</html>

* DOCTYPE specifies the document type. the document type is specified by the Document Type Definition (DTD).
* <head> section is used to specify title of the page using <title> tag. It is also used for adding external css and javascript files to html document.

**How to save and check output**

* Editors like notepad, notepad++, sublime text, visual studio code can be used to write html code
* Save html document file with .html extension
* To check output open html document with browser like google chrome, Microsoft edge, Firefox etc.

**HTML Formatting Tags**

* <b>text </b>- for making the text bold.
* <strong> text</strong>- for making the text Important text
* <i>text </i>- for making the text Italic text
* <em>text </em>- to make the Emphasized text
* <mark>text</mark>- to make the text Marked text
* <small>text </small>- to make the text Smaller text
* <del>text</del> - to make the text Deleted text
* <ins>text </ins>- to make the text Inserted text
* <sub>text <sub>- to make the text Subscript text
* <sup>text</sup>- to make the text Superscript text
* <h1> to <h6> tags – for making Headings
* Font Color (<font color=” red”>Hello</font>) – to change font color
* Font Size (<font size=”10px”>Hello</font>) – to change font size

**HTML List Tag**

* HTML List allow web developers to group a set of related items in lists
* **Unordered HTML List**
  + Starts with <ul> tag list item starts with <li> tag
  + Lists items will be marked with bullets
  + Example

|  |
| --- |
| <ul>  <li>C</li>  <li>C++</li>  <li>Java</li>  </ul> |

* **Ordered HTML List**
  + Starts with <ol> tag. Each list item starts with the <li> tag.
  + Lists items will be marked with numbers by default

|  |
| --- |
| <ol>  <li>Apple</li>  <li>Mango</li>  <li>Banana</li>  </ol> |

* **HTML Description Lists**
  + A description list is a list of terms with a description of each term.

|  |
| --- |
| <dl>  <dt>CE</dt>  <dd>- Computer Engineering</dd>  <dt>IT</dt>  <dd>- Information Technology</dd>  </dl> |

* + <dl> tag defines the description list,<dt> tag defines the term (name) <dd> tag describes each term

**Implementation:**

Create your resume using HTML (Suggested sections of resume are Personal Information, Educational Information, Professional Skills, Experience, Achievements, Hobbies), Experiment with text, colors, link and list.

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

1. Explain HTML Document Structure.
2. List and explain any five HTML formatting tags.
3. Explain ordered and unordered list with example.

# Suggested Reference:

# <https://www.w3schools.com/html/html_basic.asp>

# <https://www.w3schools.com/html/html_lists.asp>

# <https://www.w3schools.com/html/html_formatting.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 2**

**Create your class time table using table tag, experiment with rowspan, colspan, cellspacing and cellpadding attributes.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 1

**Objectives:**

1. To study HTML table tag
2. To study how to organize data in tabular format

**Theory:**

**HTML Table Tag**

* HTML tables allow web developers to arrange data into rows and columns.
* The **<table>** tag defines an HTML table.
* table row is defined with a **<tr>** tag.
* table header is defined with a **<th>** tag.
* text in **<th>** elements are bold and centered.
* Each table data/cell is defined with a **<td>.**
* By default, the text in **<td>** elements are regular and left-aligned.
* **colspan** attribute is used to make a cell span more than one column.
* **rowspan**attriute is used to make a call span more than one row.
* **cellpadding** represents the distance between cell borders and the content within a cell.
* The **cellspacing** attribute defines space between table cells.
* **Example**
  + Below code is for arranging car details in tabular format.
  + You may stude table tag and output as below.

|  |  |
| --- | --- |
| Code | Output |
| <table border="1">  <tr>  <th>Name</th>  <th>Color</th>  <th>Price</th>  </tr>  <tr>  <td>Swift VXI</td>  <td>Red</td>  <td>800000</td>  </tr>  <tr>  <td>Vagon R</td>  <td>White</td>  <td>600000</td>  </tr>  </table> |  |

**Implementation:**

Create your class time table using table tag, experiment with rowspan, colspan, cellspacing and cellpadding attributes.

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

1. Explain the use of rowspan and colspan attributes in table tag.
2. Differentiate between <td> and <th>.

# Suggested Reference:

# <https://www.w3schools.com/html/html_tables.asp>

# References used by the students: (Sufficient space to be provided)

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 3**

**Design static web pages for your college containing a description of the courses, departments, faculties, library etc. Provide links for navigation among pages.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 1

**Objectives:**

1. To study HTML Link.

**Theory:**

**HTML Links:**

* Links allow users to click their way from page to page.
* User 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.
* **Syntax**
  + **<a href="url"> link text </a>**
* **Example**

<a href=http://www.gtu.ac.in / target=”\_blank”>Visit GTU</a>

* **links will appear as follows in all browsers:**
  + An unvisited link is underlined and blue
  + A visited link is underlined and purple
  + An active link is underlined and red
* **HTML Link Taget Attribute**
  + By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.
  + The **target attribute specifies where to open the linked document**.
  + **The target attribute can have one of the following values:**
    - **\_self** - Default. Opens the document in the same window/tab as it was clicked
    - **\_blank** - Opens the document in a new window or tab
    - **\_parent** - Opens the document in the parent frame
    - **\_top** - Opens the document in the full body of the window

**Implementation:**

Design static web pages for your college containing a description of the courses, departments, faculties, library etc. Provide links for navigation among pages.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

1. Explain HTML Link target attribute.
2. How to use image as a link ?

# Suggested Reference:

# <https://www.w3schools.com/html/html_links.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 4**

**Create User Registration Form in HTML (Suggested to use fields like Name, Date of Birth, Gender, Email Id, Mobile No,Address, State,Education, Image Upload etc.) using textbox, textarea, checkbox, radio button, select box, button, file upload etc.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 1**

**Objectives:**

1. To study HTML Form Tag.
2. To Study Various Input Types like textbox, password, radio button, checkbox etc.

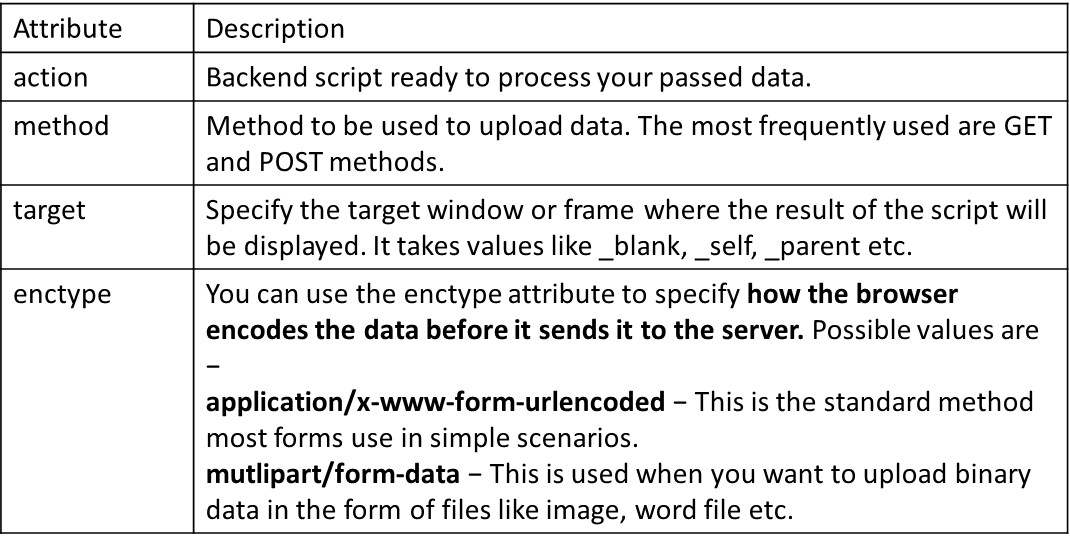
**Theory:**

HTML Forms are required, when .

* For example, for registration you may collect information like user name,email, contact number, address etc.
* A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc.
* The back-end application will perform required processing on the passed data based on defined business logic inside the application.
* There are various form elements available like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.
* The HTML **<form>** tag is used to create an HTML form
* **Syntax:**

|  |
| --- |
| <form action = "Script URL" method = "GET|POST">  form elements like input, textarea etc.  </form> |

* **Important form attributes are as given below**



* **HTML Form Controls**

|  |  |  |
| --- | --- | --- |
| **Control Name** | **Used for** | **Sample Code** |
| **Text input control**  Single line text input control | Textbox is used for accepting text from user, like firstname,lastname etc. | <input type = "text" name = "first\_name" /> |
| **Text input control**  Password input control | Password input control is used to accept password from user. | <input type = "password" name = "password" /> |
| **Text input control**  Multiline input control. | Teaxtarea is used to accept multiline text input , like comments. | <textarea rows = "5" cols = "50" name = "description">  Enter description here...  </textarea> |
| **Checkbox Control** | Checkboxes are used when more than one option is required to be selected. | <input type = "checkbox" name = "maths" value = “maths"> Maths  <input type = "checkbox" name = "physics" value = “physics">name = "password" /> |
| **Radio Button Controls** | Radio buttons are used when out of many options, just one option is required to be selected. | <input type = "radio" name = "subject" value = "maths"/> Maths  <input type = "radio" name = "subject" value = "physics"/> Physics |
| **Drop Down box Control** | provides option to list down various options in the form of drop down list, from where a user can select one or more options. | <select name = "dropdown">  <option value = "Maths" selected>Maths</option>  <option value = "Physics">Physics</option>  </select> |
| **File Upload** | It allows site users to upload a file to website.it is also known as file select box. | <input type = "file" name = "fileupload" accept = "image/\*" /> |
| **Button Control**  Submit | This creates a button that automatically submits a form. | <input type = "submit" name = "submit" value = "Submit" /> |
| **Button Control**  Reset | This creates a button that automatically resets form controls to their initial values. | <input type = "reset" name = "reset" value = "Reset" /> |
| **Button Control**  Button | This creates a button that is used to trigger a client-side script when the user clicks that button. | <input type = "button" name = "ok" value = "OK" /> |
| **Button Control**  Image | This creates a clickable button but we can use an image as background of the button. | <input type = "image" name = "imagebutton" src = "/html/images/logo.png" /> |
| **Hidden Control** | Hidden form controls are used to hide data inside the page which later on can be pushed to the server. This control hides inside the code and does not appear on the actual page. | <input type = "hidden" name = "pagename" value = "10" /> |

**Implementation:**

Create User Registration Form in HTML (Suggested to use fields like Name, Date of Birth, Gender, Email Id, Mobile No., Address, State,Education, Image Upload etc.) using textbox, textarea, checkbox, radio button, select box, button, file upload etc.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

1. Explain form tag with its attributes.
2. Differentiate between text input and password input controls.
3. Explain various types of buttons available in HTML.

# Suggested Reference:

# <https://www.w3schools.com/html/html_forms.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 5**

**Create two web pages, one contains audios and other page contains videos (using HTML5 audio and video tags). Also provide link for navigation between pages.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 1

**Objectives:**

1. To study how to add audio and video content in html page.

**Theory:**

**HTML Video**

The HTML <video> element is used to show a video on a web page.

**Example:**

<video width="320" height="240" **controls**>

<source src="movie.mp4" type="video/mp4">

<source src="movie.ogg" type="video/ogg">

Your browser does not support the video tag.

</video>

The controls attribute adds video controls, like play, pause, and volume.

The <source> element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.

The text between the <video> and </video> tags will only be displayed in browsers that do not support the <video> element.

**HTML Audio**

The HTML <audio> element is used to play an audio file on a web page.

<audio controls>

<source src="horse.ogg" type="audio/ogg">

<source src="horse.mp3" type="audio/mpeg">

Your browser does not support the audio element.

</audio>

The controlsattribute adds audio controls, like play, pause, and volume.

The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.

**Implementation:**

Create two web pages, one contains audios and other page contains videos (using HTML5 audio and video tags). Also provide link for navigation between pages.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

1. Explain audio and video tags.

# Suggested Reference:

# <https://www.w3schools.com/html/html_media.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 6**

**Create a web page using frame. Divide the page into two parts with Navigation links on left hand side of page (width=20%) and content page on right hand side of page (width = 80%). On clicking the navigation Links corresponding content must be shown on the right-hand side.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 1

**Objectives:**

1. To study frame and frameset to divide page multiple sections.
2. To understand about use of target attribute to open web page in target frame.

**Theory:**

The <frame> tag was used in HTML 4 to define one particular window (frame) within a <frameset>

HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document.

A collection of frames in the browser window is known as a frameset.

The window is divided into frames in a similar way the tables are organized: into rows and columns.

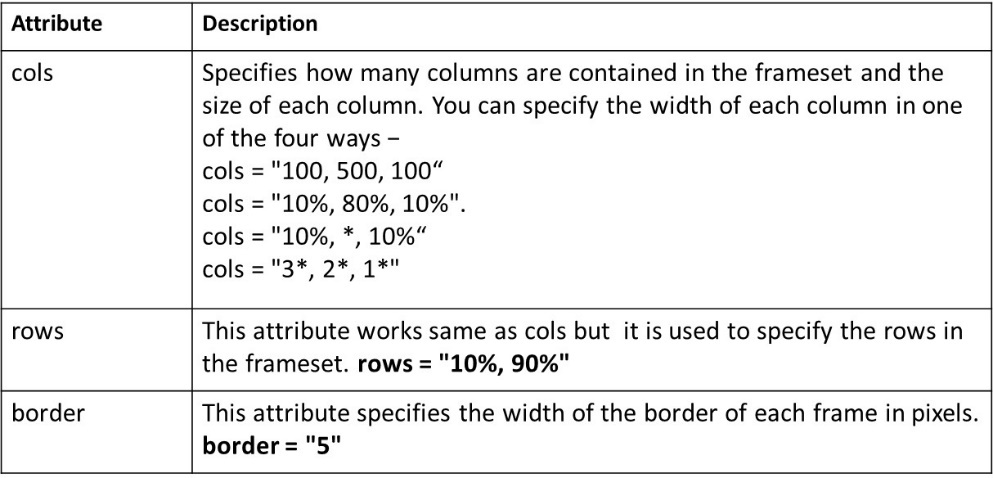
**Creating Frames**

To use frames on a page we use **<frameset> tag** instead of <body> tag.

The <frameset> tag defines, how to divide the window into frames. The rows attribute of <frameset> tag defines horizontal frames and cols attribute defines vertical frames.

Each frame is indicated by <frame> tag and it defines which HTML document shall open into the frame.

**The <frameset> Tag Attributes**

****

**Example: example to create three horizontal frames**

<frameset rows = "10%,80%,10%">

<frame name = "top" src = "./rows\_demo\_pages/top\_frame.htm" />

<frame name = "main" src = "./rows\_demo\_pages/main\_frame.htm" />

<frame name = "bottom" src = "./rows\_demo\_pages/bottom\_frame.htm" />

<noframes>

<body>Your browser does not support frames. </body>

</noframes>

</frameset>

In below implementation use target attribute in navigation link to open page in specific frame.

**Implementation:**

Create a web page using frame. Divide the page into two parts with Navigation links on left hand side of page (width=20%) and content page on right hand side of page (width = 80%). On clicking the navigation Links corresponding content must be shown on the right-hand side.

|  |
| --- |
|  |

**Output:**

**Conclusion:** (Sufficient space to be provided)

|  |
| --- |
|  |

**Quiz:**

1. Explain about rows and cols attribute of frame tag.
2. Which tag embed an inline frame in a web page?
3. Which attribute in frame tag is used to specifies the web page to load into that frame?

# Suggested Reference:

# <https://www.w3schools.com/tags/tag_frameset.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 7**

**Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS).**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 1

**Objectives:**

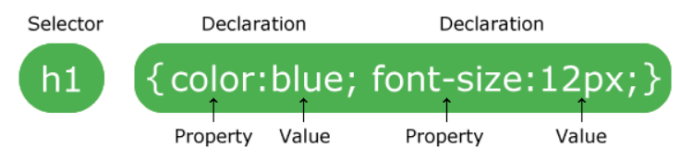
1. To understand how CSS works.

**Theory:**

**Introduction To CSS**

* CSS stands for Cascading Style Sheets
* CSS describes how HTML elements are to be displayed.
* CSS saves a lot of work. It can control the layout of multiple web pages all at once
* External stylesheets are stored in CSS files
* HTML was NEVER intended to contain tags for formatting a web page! HTML was created to describe the content of a web page.
* When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.
* To solve this problem, the World Wide Web Consortium (W3C) created CSS.
* CSS removed the style formatting from the HTML page!

**CSS Syntax**

****

* A CSS rule-set consists of a selector and a declaration block:
* The selector points to the HTML element you want to style.
* The declaration block contains one or more declarations separated by semicolons.
* Each declaration includes a CSS property name and a value, separated by a colon.
* declaration blocks are surrounded by curly braces.

**Example:**In this example all <p> elements will be center-aligned, with a red text color

|  |  |
| --- | --- |
| **Code** | **Output** |
| <!DOCTYPE html>  <html>  <head>  **<style>**  **p {**  **color: red;**  **text-align: center;**  **}**  **</style>**  </head>  <body>  **<p>Hello World!</p>**  **<p>These paragraphs are styled with CSS.</p>**  </body>  </html> |  |

* p is a selector in CSS (it points to the HTML element you want to style: <p>).
* color is a property, and red is the property value
* text-align is a property, and center is the property value

**CSS Selectors**

* **CSS Element Selector**
  + The element selector selects HTML elements based on the element name.
  + Example:

|  |
| --- |
| p {  text-align: center;  color: red;  } |

* The CSS id Selector
  + The id selector uses the id attribute of an HTML element to select a specific element.
  + The id of an element is unique within a page, so the id selector is used to select one unique element!
  + To select an element with a specific id, write a hash (#) character, followed by the id of the element.
  + Example

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  **<style>**  **#para1 {**  **text-align: center;**  **color: red;**  **}**  **</style>**  </head>  <body>  **<p id="para1">Hello World!</p>**  **<p>This paragraph is not affected by the style.</p>**  </body>  </html> |

* CSS Class Selector
  + The class selector selects HTML elements with a specific class attribute.
  + To select elements with a specific class, write a period (.) character, followed by the class name.
  + Example
    - In this example all HTML elements with class="center" will be red and center-aligned:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  **<style>**  **.center {**  **text-align: center;**  **color: red;**  **}**  **</style>**  </head>  <body>  **<h1 class="center">Red and center-aligned heading</h1>**  **<p class="center">Red and center-aligned paragraph.</p>**  </body>  </html> |

* **CSS Universal Selector**
  + The universal selector (\*) selects all HTML elements on the page.
  + Example

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  **<style>**  **\* {**  **text-align: center;**  **color: blue;**  **}**  **</style>**  </head>  <body>  <h1>Hello world!</h1>  <p>Every element on the page will be affected by the style.</p>  <p id="para1">Me too!</p>  <p>And me!</p>  </body>  </html> |

* **CSS Grouping Selector**
  + The grouping selector selects all the HTML elements with the same style definitions.
  + To group selectors, separate each selector with a comma.
  + Example:

|  |
| --- |
| h1, h2, p {  text-align: center;  color: red;  } |

* **The CSS Pseudo Class Selector**
  + Some selectors can be considered different because of the way the element they belong to works.
  + For example, the anchor that creates a link between documents can have pseudo classes attached to it simply because it is not known at the time of writing the markup what the state will be.
  + It could be visited, not visited, or in the process of being selected.
  + CSS pseudo-classes are used to add special effects to some selectors. You do not need to use JavaScript or any other script to use those effects.
  + selector: pseudo-class {property: value}
  + CSS classes can also be used with pseudo-classes
  + selector.class: pseudo-class {property: value}

|  |
| --- |
| a : link { color: red}  a : active { color: yellow}  a : visited { color: green}  a : hover { font-weight: bold}  a : link : hover {font-weight:bold} |

* + **Example**
* **Types Of CSS**
  + External CSS
  + Internal CSS
  + Inline CSS
* **Internal CSS** 
  + An internal style sheet may be used if one single HTML page has a unique style.
  + The internal style is defined inside the <style> element, inside the head section.
  + Example:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  **<style>**  **body {**  **background-color: linen;**  **}**  **h1 {**  **color: maroon;**  **margin-left: 40px;**  **}**  **</style>**  </head>  <body>  <h1>This is a heading</h1>  <p>This is a paragraph.</p>  </body>  </html> |

**-CSS Background Color**

* The background-color property specifies the background color of an element.
* With CSS, a color is most often specified by:
  + a valid color name - like "red"
  + a HEX value - like "#ff0000"
  + an RGB value - like "rgb (255,0,0)"

**Example:**

|  |
| --- |
| body {  **background-color: lightblue;**  } |

**-CSS Text Color**

* text color can be set using color property

**Example:**

|  |
| --- |
| **<h1 style="color:Tomato;">Hello World</h1>** |

**Implementation:**

Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS).

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

1. Explain the syntax of the CSS.
2. What is internal CSS?
3. Explain CSS class and Id selector.

# Suggested Reference:

# <https://www.w3schools.com/css/css_syntax.asp>

# <https://www.geeksforgeeks.org/types-of-css-cascading-style-sheet/>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 8**

**Use Inline CSS to format your resume that you created in practical no 01.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 1**

**Objectives:**

1. To understand the use of Inline CSS.

**Theory:**

**Internal CSS**

* An inline style may be used to apply a unique style for a single element.
* To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

**Example:**

|  |
| --- |
| <!DOCTYPE html>  <html>  <body>  <h1 **style="color:blue;text-align:center;"**>This is a heading</h1>  <p **style="color:red;"**>This is a paragraph.</p>  </body>  </html> |

**Implementation:**

Use Inline CSS to format your resume that you created in practical no 01.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# Explain Internal CSS VS Inline CSS.

# CSS stands for \_\_\_\_\_\_.

# Which HTML tag is used to define an internal style sheet?

# Suggested Reference:

# <https://www.geeksforgeeks.org/types-of-css-cascading-style-sheet/>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 9**

**Use External, Internal, Inline CSS to format College Web site that you created in Practical No.03**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 1**

**Objectives:**

1. To understand use of External CSS

**Theory:**

**External CSS**

* An external file is a good idea when you have a number of pages, or even a complete site, which you need to control in terms of presentation.
* it saves lots of effort as at one time you would have needed to alter each page individually.
* With an external style sheet, you can change the look of an entire website by changing just one file!
* Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.
* External CSS file must be saved with a .css extension.
* **Example**

|  |
| --- |
| **HTML Code : index.html** |
| <!DOCTYPE html>  <html>  **<head>**  **<link rel="stylesheet" type="text/css" href="mystyle.css">**  **</head>**  <body>  <h1>This is a heading</h1>  <p>This is a paragraph.</p>  </body>  </html> |
| **CSS Code : mystyle.css** |
| ***body {***  ***background-color: lightblue;***  ***}***  ***h1 {***  ***color: navy;***  ***margin-left: 20px;***  ***}*** |

**Implementation:**

Use External, Internal, Inline CSS to format Information Technology Department Web Pages that you created in Practical No. 03.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# Explain External CSS.

# Compare Internal, Inline and External CSS.

# Which property is used to change the background color?

# Which property is used to change the text color of the element?

# Suggested Reference:

# <https://www.geeksforgeeks.org/types-of-css-cascading-style-sheet/>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 10**

**Develop a java script to display today’s date.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 2

**Objectives:**

1. To understand how to write simple java script

**Theory:**

**Javascript**

* Javascript is a client side scripting language.
* HTML and CSS for static rendering of a page
* Scripting languages allows content to change dynamically
* Possible to interact with the user beyond what is possible with HTML
* Scripts are programs and can execute on the client side (the one with the browser) or server.
* Running a script on the client saves processing time on the server
* **Types Of Javascript**
  + **Internal Javascript**
    - JavaScript code is placed in the head and body section of an HTML page.
    - **Example**

|  |
| --- |
| <html>  <head>  <title>Internal JavaScript</title>  **<script type="text/javascript">**  **document.write("Hello World.!!!");**  **</script>**  </head>  <body>  </body>  </html> |

* + **External JavaScript**
    - If you want to use the same script on several pages it could be good idea to place the code in separate file, rather than writing it on each.
    - JavaScript code are stored in separate external file using the .js extension (Ex: external.js).
    - Example:

|  |
| --- |
| HTML File : index.html |
| <html>  <head>  <title>External JavaScript</title>  <script type="text/javascript" **src="external.js"**></script>  </head>  <body>  </body>  </html> |
| External JavaScript file : external.js |
| document.write("This is External Javascript Example.!!!"); |

**Implementation:**

Develop a java script to display today’s date.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is javascript?

# Explain internal and external javascript.

# Suggested Reference:

# <https://www.w3schools.com/JSREF/jsref_obj_date.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 11**

**Develop simple calculator for addition, subtraction, multiplication and division operation using java script.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 2**

**Objectives:**

1. To understand the use of mathematical operators in javascript.
2. To understand the use of document object model.
3. To understand javascript event handling.

**Theory:**

**Javascript Syntax**

**How to create and use variables?**

|  |
| --- |
| **varx,y,z;**  **x=5;**  **y=5**  **z=x+y;**  document.write(“total is : ”+z) |

**The HTML DOM (Document Object Model)**

* When a web page is loaded, the browser creates a Document Object Model of the page.
* The HTML DOM model is constructed as a tree of Objects:
* Using DOM Javascript can
  + change all the HTML elements in the page
  + change all the HTML attributes in the page
  + change all the CSS styles in the page
  + remove existing HTML elements and attributes
  + add new HTML elements and attributes
  + react to all existing HTML events in the page
  + create new HTML events in the page

|  |
| --- |
| Figure 1 Document Object Model |

**DOM Examples**

**Example 1: following example changes the content of <p> element**

|  |
| --- |
| <html> <body>  <p id="demo"></p>  <script> document.getElementById("demo").innerHTML = "Hello World!"; </script>  </body> </html> |

Here getElementById is a method, while innerHTML is a property.

**Example 2: Validate Numeric Input**

|  |
| --- |
| <!DOCTYPE html>  <html>  <body>  <h2>Number Validation</h2>  <p>Enter a number between 1 and 10:</p>  <input id="numb">  <button type="button" onclick="myFunction()">Submit</button>  <p id="demo"></p>  <script>  function myFunction() {  // Get the value of the input field with id="numb"  let x = document.getElementById("numb").value;  // If x is Not a Number or less than one or greater than 10  let text;  if (isNaN(x) || x < 1 || x > 10) {  text = "Input not valid";  } else {  text = "Input OK";  }  document.getElementById("demo").innerHTML = text;  }  </script>  </body>  </html> |

**Implementation:**

Develop simple calculator for addition, subtraction, multiplication and division operation using java script.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# Explain Document Object Model.

# Suggested Reference:

# <https://www.w3schools.com/js/js_htmldom.asp>

# <https://www.w3schools.com/js/js_validation.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 12**

**Write a java script code to combine and display the information in textbox when the button is clicked use registration page that you created in Practical No.4.**

**Date:**

**Competency and PracticalSkills:**

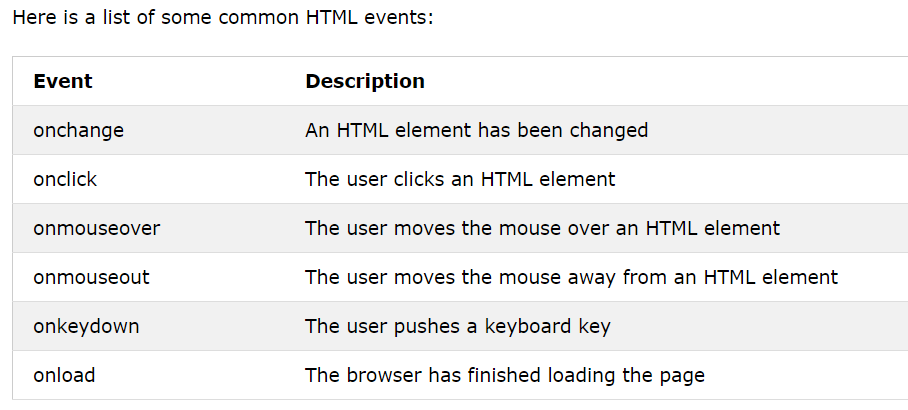
**Relevant CO:** 2

**Objectives:**

1. To understand the use of DOM for getting values from Form Controls.
2. To understand event handling with javascript

**Theory:**

**What is an Event?**

* JavaScript's interaction with HTML is handled through events that occur when the user or the browser manipulates a page.
* When the page loads, it is called an event. When the user clicks a button, that click too is an event. Other examples include events like pressing any key, closing a window, resizing a window, etc.
* Developers can use these events to execute JavaScript coded responses, which cause buttons to close windows, messages to be displayed to users, data to be validated, and virtually any other type of response imaginable.
* Events are a part of the Document Object Model (DOM) Level 3 and every HTML element contains a set of events which can trigger JavaScript Code.
* 

Example: the following javascript example demonstrate how to fetch value from textbox and display using alert()

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title>Java Script Demo</title>  <script>  function showData()  {  var uname,email;  uname = document.forms["myform"]["username"].value;  email = document.forms["myform"]["email"].value;  alert("you entered name:"+uname+" email:"+email);  }  </script>  </head>  <body>  <form name="myform">  UserName : <input type="text" name="username"/><br/>  Password : <input type="email" name="email"/><br/>  <input type="button" value="display" onclick="showData()" />  </form>  </body>  </html> |

**Implementation:**

Write a java script code to combine and display the information in textbox when the button is clicked use registration page that you created in Practical No.4.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# Explain event handling with javascript.

# Suggested Reference:

# <https://www.w3schools.com/js/js_validation.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 13**

**Use JavaScript to Implement validation in Practical No.4.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 2

**Objectives:**

1. To understand validation using javascript.

**Theory:**

Javascript can be used for HTML form validation

Following example demonstrate form validation using javascript

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <script>  function validateForm() {  let x = document.forms["myForm"]["fname"].value;  if (x == "") {  alert("Name must be filled out");  return false;  }  }  </script>  </head>  <body>  <h2>JavaScript Validation</h2>  <form name="myForm" action="/action\_page.php" onsubmit="return validateForm()" method="post">  Name: <input type="text" name="fname">  <input type="submit" value="Submit">  </form>  </body>  </html> |

**Implementation:**

Use JavaScript to Implement validation in Practical No.4.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# Explain javascript form validation.

# Suggested Reference:

# <https://www.w3schools.com/js/js_validation.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 14**

**Write a PHP program to check if number is prime or not.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 3**

**Objectives:**

1. To understand how to write simple php program
2. To understand how to use php conditional and Loops Statement

**Theory:**

**PHP**

* PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.
* PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.
* **Syntax**

|  |
| --- |
| <?php // PHP code goes here ?> |

* **Example:**demonstrate printing Hello World

|  |
| --- |
| <!DOCTYPE html>  <html>  <body>  <?php  echo "Hello World";  ?>  </body>  </html> |

* Creating (Declaring) PHP Variables

|  |
| --- |
| <?php $txt = "Hello world!"; $x = 5; $y = 10.5; ?> |

* PHP Conditional Statements

|  |  |
| --- | --- |
| Statement | Syntax |
| PHP - The if Statement | if (*condition*) { *code to be executed if condition is true*; } |
| PHP - The if...else Statement | if (*condition*) {   *code to be executed if condition is true;* } else {   *code to be executed if condition is false;* } |
| PHP - The if...elseif...else Statement | if (*condition*) {   *code to be executed if this condition is true;* } elseif (*condition*) {  *code to be executed if first condition is false and this condition is true;* } else {   *code to be executed if all conditions are false;* } |

* PHP Loop Statements

|  |  |
| --- | --- |
| Statement | Syntax |
| The PHP while Loop | while (*condition is true*) { *code to be executed*; } |
| The PHP do...while Loop | do { *code to be executed;* } while (*condition is true*); |
| The PHP for Loop | for (*init counter; test counter; increment counter*) {   *code to be executed for each iteration;* } |
| The PHP foreach Loop | foreach ($*array*as$*value*) {   *code to be executed;* } |

**Implementation:**

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is PHP? Explain PHP Syntax.

# Explain foreach Loop in PHP.

# Suggested Reference:

# <https://www.w3schools.com/php/php_looping.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 15**

**Use Registration Form from practical number 4 to store user registration details in MySql database. On submission next page displays all registration data in in html table using php. Also provide feature to update and delete the registration data.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 4

**Objectives:**

1. To understand how to use MySql database
2. To understand how to perform CRUD operations.

**Theory:**

Accessing MySQL from PHP Note that documentation is available online here: http://www.php.net/manual/en/ref.mysql.php

Basically, there are four things you want to be able to do in MySQL from within PHP:

1. connect to the mysql database

2. execute mysql queries

3. check the status of your mysql commands

4. disconnect from the mysql database

Queries can be any kind of MySQL query, including SELECT, UPDATE, INSERT, etc. Using SELECT queries, you can execute MySQL/PHP functions to put the data read from the MySQL database into PHP variables. Then you can use the PHP variables in your PHP script to do whatever analysis, display, etc. that you want.

1. Connect to the MySQL database

Here is an example of connecting to the MySQL database from within PHP:

|  |
| --- |
| $conn=mysql\_connect($mysql\_host,$mysql\_user,$mysql\_password) or die('Could not connect: ’.mysql\_error());  echo ’Connected successfully’;  mysql\_select\_db( $mysql\_db ) or die( ’Could not select database’ ); |

You will need to replace the variables $mysql\_host, $mysql\_user, $mysql\_password and $mysql\_db with strings containing the values for connecting to your database. $mysql\_host is "localhost"

Notice that there are two functions invoked:

- Logs into mysql: mysql\_connect()

- Selects the database to use: mysql\_select\_db()

Also notice that you put your un-encrypted password in the script that connects to the database. So be careful where you put that script! Make sure it is in a directory where there is a default index.html (or index.php) file so that nobody can get to the script from a web browser.

2. Execute MySQL queries

Here is an example of executing a SELECT query from within PHP:

|  |
| --- |
| // set up and execute the MySQL query  $query = ’SELECT \* FROM my\_table’;  $result = mysql\_query( $query ) or die( ’Query failed: ’. mysql\_error() );  // print the results as an HTML table  echo " \n";  while ( $row = mysql\_fetch\_array( $result, MYSQL\_ASSOC ))  {  echo "\t \n";  foreach ( $row as $item )  {  echo "\t\t $item\n";  }  echo "\t\n";  }  echo "\n";  // free result  mysql\_free\_result( $result ); |

There are three functions used here:

- To execute the query and store the result in a local variable: mysql\_query()

- Parse the data read returned from the query as an array: mysql\_fetch\_array()

- Free the memory used by the query result: mysql\_free\_result()

NOTE that if the result returned is a scalar and not an array, then only mysql\_query() needs to be called and does not need to be followed by a call to mysql\_fetch\_array().

Finally, note the use of mysql\_error() in the query function.

3. Check the status of your MySQL commands

If errors occur, the functions return errors. These errors can be read as strings using the function mysql\_error(). Note the usage in this statement:

|  |
| --- |
| $conn=mysql\_connect($mysql\_host,$mysql\_user,$mysql\_password) or die('Could not  connect: ’.mysql\_error());  echo ’Connected successfully’; |

4. Disconnect from the MySQL database

To disconnect from MySQL, there is one function needed:

|  |
| --- |
| mysql\_close($conn); |

**Implementation:**

Use Registration Form from practical number 5 to store user registration details in MySql database. On submission next page displays all registration data in in html table using php. Also provide feature to update and delete the registration data.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is MySql?

# Write a sample code to demonstrate phpmysql connectivity.

# Suggested Reference:

* <http://www.php.net/manual/en/ref.mysql.php>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 16**

**Write a PHP script for user authentication using PHP-MYSQL. Use session for storing username.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 4

**Objectives:**

1. To understand session in PHP

**Theory:**

**What is a PHP Session?**

When you work with an application, you open it, do some changes, and then you close it. This is much like a Session. The computer knows who you are. It knows when you start the application and when you end. But on the internet there is one problem: the web server does not know who you are or what you do, because the HTTP address doesn't maintain state.

Session variables solve this problem by storing user information to be used across multiple pages (e.g. username, favorite color, etc.). By default, session variables last until the user closes the browser.

So; Session variables hold information about one single user, and are available to all pages in one application.

**Start a PHP Session**

A session is started with the session\_start() function.

Session variables are set with the PHP global variable: $\_SESSION.

Now, let's create a new page called "demo\_session1.php". In this page, we start a new PHP session and set some session variables:

|  |
| --- |
| <?php // Start the session session\_start(); ?> <!DOCTYPE html> <html> <body>  <?php // Set session variables $\_SESSION["favcolor"] = "green"; $\_SESSION["favanimal"] = "cat"; echo "Session variables are set."; ?>  </body> </html> |

**Get PHP Session Variable Values**

Next, we create another page called "demo\_session2.php". From this page, we will access the session information we set on the first page ("demo\_session1.php").

Notice that session variables are not passed individually to each new page, instead they are retrieved from the session we open at the beginning of each page (session\_start()).

Also notice that all session variable values are stored in the global $\_SESSION variable:

|  |
| --- |
| <?php session\_start(); ?> <!DOCTYPE html> <html> <body>  <?php // Echo session variables that were set on previous page echo "Favorite color is " . $\_SESSION["favcolor"] . ".<br>"; echo "Favorite animal is " . $\_SESSION["favanimal"] . "."; ?>  </body> </html> |

**Modify a PHP Session Variable**

To change a session variable, just overwrite it:

|  |
| --- |
| <?php session\_start(); ?> <!DOCTYPE html> <html> <body>  <?php // to change a session variable, just overwrite it $\_SESSION["favcolor"] = "yellow"; print\_r($\_SESSION); ?>  </body> </html> |

**Destroy a PHP Session**

To remove all global session variables and destroy the session, use session\_unset() and session\_destroy():

|  |
| --- |
| <?php session\_start(); ?> <!DOCTYPE html> <html> <body>  <?php // remove all session variables session\_unset();  // destroy the session session\_destroy(); ?>  </body> </html> |

**Implementation:**

Write a PHP script for user authentication using PHP-MYSQL. Use session for storing username.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is PHP Session?

# How to destroy PHP Session?

# Suggested Reference:

# <https://www.w3schools.com/php/php_sessions.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 17**

**Using AJAX Create visual search feature to search using name for practical number 15 which list name, mobile number and email id of matching users.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 5

**Objectives:**

1. To understand how Ajax works.

**Theory:**

What is AJAX?

AJAX = Asynchronous JavaScript and XML.

AJAX is not a programming language.

AJAX just uses a combination of:

- A browser built-in XMLHttpRequest object (to request data from a web server)

- JavaScript and HTML DOM (to display or use the data)

AJAX allows web pages to be updated asynchronously by exchanging data with a web server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.



Ref: https://www.w3schools.com/js/js\_ajax\_intro.asp

Steps:

1. An event occurs in a web page (the page is loaded; a button is clicked)

2. An XMLHttpRequest object is created by JavaScript

3. The XMLHttpRequest object sends a request to a web server

4. The server processes the request

5. The server sends a response back to the web page

6. The response is read by JavaScript

7. Proper action (like page update) is performed by JavaScript

The keystone of AJAX is the XMLHttpRequest object.

1. Create an XMLHttpRequest object

|  |
| --- |
| *variable*= new XMLHttpRequest(); |

1. Define a callback function

|  |
| --- |
| *xhttp.onload = function() {   // What to do when the response is ready }* |

1. Open the XMLHttpRequest object

|  |
| --- |
| *xhttp.open("GET", "ajax\_info.txt");* |

1. Send a Request to a server

|  |
| --- |
| *xhttp.send();* |

## XMLHttpRequest Object Methods

|  |  |
| --- | --- |
| **Method** | **Description** |
| new XMLHttpRequest() | Creates a new XMLHttpRequest object |
| abort() | Cancels the current request |
| getAllResponseHeaders() | Returns header information |
| getResponseHeader() | Returns specific header information |
| open(method, url, async, user, psw) | Specifies the request  method: the request type GET or POST url: the file location async: true (asynchronous) or false (synchronous) user: optional user name psw: optional password |
| send() | Sends the request to the server Used for GET requests |
| send(string) | Sends the request to the server. Used for POST requests |
| setRequestHeader() | Adds a label/value pair to the header to be sent |

## XMLHttpRequest Object Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| Onload | Defines a function to be called when the request is recieved (loaded) |
| onreadystatechange | Defines a function to be called when the readyState property changes |
| readyState | Holds the status of the XMLHttpRequest. 0: request not initialized 1: server connection established 2: request received 3: processing request 4: request finished and response is ready |
| responseText | Returns the response data as a string |
| responseXML | Returns the response data as XML data |
| Status | Returns the status-number of a request 200: "OK" 403: "Forbidden" 404: "Not Found" For a complete list go to the [Http Messages Reference](https://www.w3schools.com/tags/ref_httpmessages.asp) |
| statusText | Returns the status-text (e.g. "OK" or "Not Found") |

**Call back function**

With the XMLHttpRequest object you can define a callback function to be executed when the request receives an answer. The function is defined in the onload property of the XMLHttpRequest object:

|  |
| --- |
| *xhttp.onload = function() {   document.getElementById("demo").innerHTML = this.responseText; } xhttp.open("GET", "ajax\_info.txt"); xhttp.send();* |

## The onreadystatechange Property

The readyState property holds the status of the XMLHttpRequest. The onreadystatechange property defines a callback function to be executed when the readyState changes. The status property and the statusText properties hold the status of the XMLHttpRequest object.

|  |  |
| --- | --- |
| readyState | Holds the status of the XMLHttpRequest. 0: request not initialized 1: server connection established 2: request received 3: processing request 4: request finished and response is ready |
| Status | 200: "OK" 403: "Forbidden" 404: "Page not found" For a complete list go to the [Http Messages Reference](https://www.w3schools.com/tags/ref_httpmessages.asp) |

**Implementation:**

Using AJAX Create visual search feature to search using name for practical number 16 which list name, mobile number and email id of matching users.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is Ajax?

# Explain XMLHttpRequest.

# Suggested Reference:

# <https://www.w3schools.com/xml/ajax_intro.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 18**

**Create a REST API using php.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 5**

**Objectives:**

1. To understand how REST API works.

**Theory:**

## What is REST?

REST stands for Representational State Transfer, REST is an architectural style which defines a set of constraints for developing and consuming web services through standard protocol (HTTP). REST API is a simple, easy to implement and stateless web service. There is another web service available which is SOAP which stands for Simple Object Access Protocol which is created by Microsoft.

REST API is widely used in web and mobile applications as compared to SOAP. REST can provide output data in multiple formats such as JavaScript Object Notation (JSON), Extensible Markup Language (XML), Command Separated Value (CSV) and many others while SOAP described output in Web Services Description Language (WSDL).

## How Does REST API Work

REST requests are related to CRUD operations (Create, Read, Update, Delete) in database, REST uses GET, POST, PUT and DELETE requests. Let me compare them with CRUD.

* **GET** is used to retrieve information which is similar to **Read**
* **POST** is used to create new record which is similar to **Create**
* **PUT** is used to update record which is similar to **Update**
* **DELETE** is used to delete record which is similar to **Delete**

## How to Create and Consume Simple REST API in PHP

JSON format is the most common output format of REST API, we will use the JSON format to consume our simple REST API. We will develop an online transaction payment REST API for our example. I will try to keep it as simple as possible so i will use **GET** request to retrieve information.

1. Create REST API in PHP
2. Consume REST API in PHP

## 1. Create REST API in PHP

To create a REST API, follow these steps:

1. Create a Database and Table with Dummy Data
2. Create a Database Connection
3. Create a REST API File

### A. Create a Database and Table with Dummy Data

To create database run the following query.

|  |
| --- |
| *CREATE DATABASE allphptricks;* |

To create a table run the following query. **Note:** I have already attached the SQL file of this table with dummy data, just download the complete zip file of this tutorial.

|  |
| --- |
| *CREATE TABLE IF NOT EXISTS `transactions` (*  *`id` int(20) NOT NULL AUTO\_INCREMENT,*  *`order\_id` int(50) NOT NULL,*  *`amount` decimal(9,2) NOT NULL,*  *`response\_code` int(10) NOT NULL,*  *`response\_desc` varchar(50) NOT NULL,*  *PRIMARY KEY (`id`),*  *UNIQUE KEY `order\_id` (`order\_id`)* *) ENGINE=InnoDB DEFAULT CHARSET=latin1 ;* |

### B. Create a Database Connection

Just create a **db.php** file and paste the following database connection in it. Make sure that you update these credentials with your database credentials.

|  |
| --- |
| *// Enter your Host, username, password, database below.*  *$con = mysqli\_connect("localhost","root","","allphptricks");*  *if (mysqli\_connect\_errno()){*  *echo "Failed to connect to MySQL: " .mysqli\_connect\_error();*  *die();*  *}* |

### C. Create a REST API File

Create a **api.php** file and paste the following script in it.

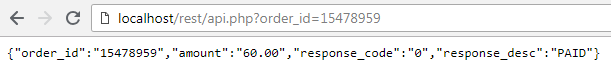
|  |
| --- |
| *<?php*  *header("Content-Type:application/json");*  *if (isset($\_GET['order\_id']) && $\_GET['order\_id']!="") {*  *include('db.php');*  *$order\_id = $\_GET['order\_id'];*  *$result = mysqli\_query(*  *$con,*  *"SELECT \* FROM `transactions` WHERE order\_id=$order\_id");*  *if(mysqli\_num\_rows($result)>0){*  *$row = mysqli\_fetch\_array($result);*  *$amount = $row['amount'];*  *$response\_code = $row['response\_code'];*  *$response\_desc = $row['response\_desc'];*  *response($order\_id, $amount, $response\_code,$response\_desc);*  *mysqli\_close($con);*  *}else{*  *response(NULL, NULL, 200,"No Record Found");*  *}*  *}else{*  *response(NULL, NULL, 400,"Invalid Request");*  *}*  *function response($order\_id,$amount,$response\_code,$response\_desc){*  *$response['order\_id'] = $order\_id;*  *$response['amount'] = $amount;*  *$response['response\_code'] = $response\_code;*  *$response['response\_desc'] = $response\_desc;*    *$json\_response = json\_encode($response);*  *echo $json\_response;*  *}*  *?>* |

The above script will accept the GET request and return output in the JSON format.

I have created all these files in folder name **rest,** now you can get the transaction information by browsing the following URL.

|  |
| --- |
| [*http://localhost/rest/api.php?order\_id=15478959*](http://localhost/rest/api.php?order_id=15478959) |

You will get the following output.

[](https://www.allphptricks.com/wp-content/uploads/2018/07/rest1.png)

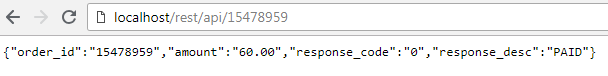
Above URL is not user friendly, therefore we will rewrite URL through the .htaccess file, copy paste the following rule in .htaccess file.

|  |
| --- |
| *RewriteEngine On # Turn on the rewriting engine*  *RewriteRule ^api/([0-9a-zA-Z\_-]\*)$ api.php?order\_id=$1 [NC,L]* |

Now you can get the transaction information by browsing the following URL.

|  |
| --- |
| [*http://localhost/rest/api/15478959*](http://localhost/rest/api/15478959) |

You will get the following output.

[](https://www.allphptricks.com/wp-content/uploads/2018/07/rest2.png)

## 2. Consume REST API in PHP

To consume a REST API, follow these steps:

1. Create an Index File with HTML Form
2. Fetch Records through CURL

### 1. Create an Index File with HTML Form

|  |
| --- |
| *<form action="" method="POST">*  *<label>Enter Order ID:</label><br />*  *<input type="text" name="order\_id" placeholder="Enter Order ID" required/>*  *<br /><br />*  *<button type="submit" name="submit">Submit</button>*  *</form>* |

### 2. Fetch Records through CURL

|  |
| --- |
| *<?php*  *if (isset($\_POST['order\_id']) && $\_POST['order\_id']!="") {*  *$order\_id = $\_POST['order\_id'];*  *$url = "http://localhost/rest/api/".$order\_id;*    *$client = curl\_init($url);*  *curl\_setopt($client,CURLOPT\_RETURNTRANSFER,true);*  *$response = curl\_exec($client);*    *$result = json\_decode($response);*    *echo "<table>";*  *echo "<tr><td>Order ID:</td><td>$result->order\_id</td></tr>";*  *echo "<tr><td>Amount:</td><td>$result->amount</td></tr>";*  *echo "<tr><td>Response Code:</td><td>$result->response\_code</td></tr>";*  *echo "<tr><td>Response Desc:</td><td>$result->response\_desc</td></tr>";*  *echo "</table>";*  *}*  *?>* |

You can do anything with these output data, you can insert or update it into your own database if you are using REST API of any other service provider. Usually in case of online transaction, the service provider provides status of payment via API. You can check either payment is made successfully or not. They also provide a complete guide of it.

***Note: Make sure CURL is enabled on your web server or on your localhost when you are testing demo.***

**Implementation:**

Create a REST API using php.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is REST API?

# Suggested Reference:

# <https://www.allphptricks.com/create-and-consume-simple-rest-api-in-php/>)

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 19**

**Create an Image slider using jQuery.**

**Date:**

**Competency and PracticalSkills:**

**Relevant CO:** 5

**Objectives:**

1. To understand how JQuery Works.

**Theory:**

JQUERY

The purpose of jQuery is to make it much easier to use JavaScript on your website. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code. jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

1. HTML/DOM manipulation
2. CSS manipulation
3. HTML event methods
4. Effects and animations
5. AJAX

There are several ways to start using jQuery on your web site.

You can:

* Download the jQuery library from jQuery.com
* Include jQuery from a CDN, like Google

|  |
| --- |
| *<head> <script src="jquery-3.6.4.min.js"></script> </head>*  *OR*  *<head>*  *<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js">*  *</script>*  *</head>* |

The jQuery syntax is tailor-made for **selecting** HTML elements and performing some **action** on the element(s).

Basic syntax is:

|  |
| --- |
| *$(selector).action()* |

* A $ sign to define/access jQuery
* A (*selector*) to "query (or find)" HTML elements
* A jQuery *action*() to be performed on the element(s)

Examples:

|  |
| --- |
| *$(this).hide() - hides the current element.*  *$("p").hide() - hides all <p> elements.*  *$(".test").hide() - hides all elements with class="test".*  *$("#test").hide() - hides the element with id="test".* |

All jQuery methods in our examples, are inside a document ready event:

|  |
| --- |
| *$(document).ready(function(){    // jQuery methods go here…  });* |

This is to prevent any jQuery code from running before the document is finished loading (is ready). It is good practice to wait for the document to be fully loaded and ready before working with it. This also allows you to have your JavaScript code before the body of your document, in the head section.

jQuery selectors allow you to select and manipulate HTML element(s).

jQuery selectors are used to "find" (or select) HTML elements based on their name, id, classes, types, attributes, values of attributes and much more. It's based on the existing [CSS Selectors](https://www.w3schools.com/cssref/css_selectors.asp), and in addition, it has some own custom selectors.

All selectors in jQuery start with the dollar sign and parentheses: $().

When a user clicks on a button, all <p> elements will be hidden:

|  |
| --- |
| *$(document).ready(function(){   $("button").click(function(){     $("p").hide();   }); });* |

The jQuery #id selector uses the id attribute of an HTML tag to find the specific element.

An id should be unique within a page, so you should use the #id selector when you want to find a single, unique element.

To find an element with a specific id, write a hash character, followed by the id of the HTML element:

|  |
| --- |
| $("#test") |

When a user clicks on a button, the element with id="test" will be hidden:

|  |
| --- |
| *$(document).ready(function(){   $("button").click(function(){     $("#test").hide();   }); });* |

All the different visitors' actions that a web page can respond to are called events. An event represents the precise moment when something happens.

Examples:

* moving a mouse over an element
* selecting a radio button
* clicking on an element

The term **"fires/fired"** is often used with events. Example: "The keypress event is fired, the moment you press a key". Here are some common DOM events:

|  |  |  |  |
| --- | --- | --- | --- |
| **Mouse Events** | **Keyboard Events** | **Form Events** | **Document/Window Events** |
| Click | Keypress | submit | Load |
| Dblclick | Keydown | change | Resize |
| Mouseenter | Keyup | focus | Scroll |
| Mouseleave |  | blur | Unload |

In jQuery, most DOM events have an equivalent jQuery method. To assign a click event to all paragraphs on a page, you can do this:

|  |
| --- |
| *$("p").click();* |

The next step is to define what should happen when the event fires. You must pass a function to the event:

|  |
| --- |
| *$("p").click(function(){   // action goes here!! });* |

## Commonly Used jQuery Event Methods

**$(document).ready()**

The $(document).ready() method allows us to execute a function when the document is fully loaded.

**click()**

The click() method attaches an event handler function to an HTML element. The function is executed when the user clicks on the HTML element. The following example says: When a click event fires on a <p> element; hide the current <p> element:

|  |
| --- |
| *$("p").click(function(){   $(this).hide(); });* |

**dblclick()**

The dblclick() method attaches an event handler function to an HTML element. The function is executed when the user double-clicks on the HTML element:

|  |
| --- |
| *$(*"p"*).*dblclick*(*function*(){* *$(*this*).*hide*();* *});* |

**mouseenter()**

The mouseenter() method attaches an event handler function to an HTML element. The function is executed when the mouse pointer enters the HTML element:

|  |
| --- |
| *$("#p1").mouseenter(function(){   alert("You entered p1!"); });* |

**Implementation:**

Create an Image slider using jQuery.

|  |
| --- |
|  |

**Output:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is jquery?

# Javascript Vs. Jquery

# Suggested Reference:

# <https://www.w3schools.com/jquery/jquery_intro.asp>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |

**Experiment No: 20**

**Cookie Example**

Create HTML form with one textbox and button. Keep button label as SAVE. User will enter color name in textbox and click on save button. On save, the value of textbox color name should be saved in COOKIE. Whenever user opens page again, the background color should be same as saved in cookie. Whenever user opens page again, the background color should be same as saved in cookie.

**Date:**

**Competency and PracticalSkills:**

**Relevant CO: 5**

**Objectives:**

1. To understand use of COOKIES.

**Theory:**

**Conclusion:**

|  |
| --- |
|  |

**Quiz:**

# What is cookie?

# What is the life of cookie?

# Suggested Reference:

# <https://www.w3schools.com>

# References used by the students:

# Rubric wise marks obtained:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | Total |
| Marks |  |  |  |  |