



* Aim: Create a webpage using basic HTML tags.

* Theory:

• HTML :

→ It stands for HyperText Markup Language. It's the standard language used to create and design web pages. HTML consists of various elements and tags that structure the content and define its appearance on web browser.

Basic tags:

- ① <html> : Defines root of an HTML document.
- ② <head> : Contains meta information about HTML document, such as title, links to stylesheets and scripts.
- ③ <title> : Sets title of HTML document, which appears in browser's title bar or tab.
- ④ <body> : Contains content of HTML document, such as text, images, link, etc.
- ⑤ <h1> to <h6> : Defines headings of different levels (most important '<h1>' to least important '<h6>')



- ⑥ `<p>` : Defines a paragraph
- ⑦ `<a>` : Defines a hyperlink.
- ⑧ `` : Inserts an image into webpage.
- ⑨ `
` : Inserts a single line break.
- ⑩ `<hr>` : Defines thematic break, often as horizontal line.

* Program:

```

<!DOCTYPE html>
<html>
<head>
    <title> My First Webpage </title>
</head>
<body>
    <h1> Welcome to My Webpage </h1>
    <p> This is paragraph </p>

    <img src = "hello.jpg" alt = "hello">
    <br>
</body>
</html>

```

* Result:

→ Hence, webpage using basic HTML tags created successfully.



* Aim: Create a Resume using basic HTML tags.

* Theory:

 : Emphasizes the text within as bold.

 : Defines Unordered List

 : Represents a list item within .

* Program:

```
<!DOCTYPE html>
<html>
<head>
    <title> Resume - Yash Pounikar </title>
</head>
<body>
    <header>
        <h2> Yash Pounikar </h2>
        <p> 2nd Year B.Tech in Artificial Intelligence </p>
    </header>

    <section>
        <h2> Contact Information </h2>
        <p> Email: pounikaryash01@gmail.com </p>
        <br>
    </section>

    <section>
        <h2> Education </h2>
```



Date :

<p> Bachelor of Technology (BTech)
in Artificial Intelligence </p>
<p> Pariyadarshini T.L. College of Engineering, Nagpur </p>
<p> Expected Graduation: 2026 </p>
</section>

<section>
<h2> Skills </h2>

 Programming Languages: C, Python, Java,
HTML, CSS,
JavaScript
 Web Development
 Problem Solving and Critical Thinking

</section>

<section>
<h2> Experience </h2>
<p> NO professional experience yet, but
actively working on personal projects
related to web </p>
</section>
</body>
</html>

* Result:

Hence, the resume using basic HTML tags
created successfully.



Date :

Practical No.3

* Aim: Create a Table using different table tags.

* Theory:

Table tags in HTML are used to create a table in webpage.

<table> : Defines table .

<tr> : Defines row within table .

<th> : Defines a header cell in table

<td> Defines a standard cell in table .

<colspan> : allows span area for column .

* Program:

```
<!DOCTYPE html>
<html>
<head>
    <title> W3 Weekly Timetable </title>
    <style>
        table {
            width: 100%; 
            border-collapse: collapse;
        }
        th, td {
            border: 2px solid black;
            padding: 8px;
        }
    </style>
</head>
<body>
    <h1> Weekly Timetable </h1>
    <table>
        <thead>
            <tr>
                <th> Subject </th>
                <th> Day </th>
                <th> Time </th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td> Mathematics </td>
                <td> Monday </td>
                <td> 9 AM - 11 AM </td>
            </tr>
            <tr>
                <td> English </td>
                <td> Tuesday </td>
                <td> 10 AM - 12 PM </td>
            </tr>
            <tr>
                <td> Physics </td>
                <td> Wednesday </td>
                <td> 11 AM - 1 PM </td>
            </tr>
            <tr>
                <td> Chemistry </td>
                <td> Thursday </td>
                <td> 12 PM - 2 PM </td>
            </tr>
            <tr>
                <td> Computer </td>
                <td> Friday </td>
                <td> 1 PM - 3 PM </td>
            </tr>
        </tbody>
    </table>
</body>
</html>
```

text-align: center;

}

th.day {

width: 14%;

}

th.time {

width: 12%;

}

th.lab {

width: 24%;

}

</style>

</head>

<body>

<h1> Weekly Timetable </h1>

<table>

<tr>

<th class = "day"> </th>

<th class = "time"> ~~10:15am - 11:15am~~

</th>

<th class = "time"> 11:15 am - 12:15 pm

</th>

<th class = "time"> 12:15 p.m - 1:15pm

</th>

<th class = "time"> 1:15pm - 2:00pm

</th>

<th class = "time"> 2:00pm - 3:00pm

</th>

<th class = "time"> 3:00pm - 4:00pm

</th>

<th class = "time"> 4:00pm - 5:00pm

</th>

</tr>

<tr>

<th class = "day"> Monday </th>

<td> IAI </td>

<td> CN </td>

<td> FIPR </td>

<td> colspan = "3" rowspan = "5" > Recess </td>

<td> LAPS </td>

<td> colspan = "2" rowspan = "1" > CNL/CWS/PS-II

</td>

</tr>

<tr>

<th class = "day"> Tuesday </th>

<td> PP </td>

<td> NPTEL </td>

<td> IR </td>

<td> LAPS </td>

<td> colspan = "2" rowspan = "1" > CNL/CWS/PS-II

</td>

</tr>

<tr>

<th class = "day"> Wednesday </th>

<td> CN </td>

<td> IAI </td>

<td> TOC </td>

<td> PP </td>

<td> IR </td>

<td> Library </td>

</tr>

<tr>

<th class = "day" > Thursday </th>

<td> IR </td>

<td colspan = "2" > CNL/CWS/PS-II </td>

<td> TOC </td>

<td> LAPS </td>

<td> Technical Exposure </td>

</td>

<tr>

<th class = "day" > Friday </th>

<td> LAPS </td>

<td> CN </td>

<td> FJPR </td>

<td> PP </td>

<td> TOC </td>

<td> IAI </td>

</td>

<tr>

<th class = "day" > Saturday </th>

~~<td>~~ <td colspan = "7" > </td>

</td>

</table>

</body>

</html>

*

Result:

Hence, the table using different table tag created successfully.



Practical No. 4

Aim: Create a account on GIT & GITHUB and upload a file on it.

Theory:

Git:

It's a distributed version control system that allows multiple developers to collaborate on projects. It tracks changes to files over time, enabling users to revert to previous version, merge changes from different sources and manage concurrent work efficiently.

Github:

On the other hand, Github is web based platform built around Git. It provides hosting for Git repositories and offers additional features such as issue tracking, project management tools, and collaboration features like pull requests and code reviews. It facilitates collaboration among developers by providing a centralized platform for storing, sharing and contributing to code bases.



Date :

* Steps to upload file on github :

Step ① : Sign in to Github.

Step ② : Create a New Repository

Step ③ : Navigate to Repository

Step ④ : Upload files using "Add files" option
in that using "upload files".
Drag and drop file you want to
upload or click "choose your files"
to browse and select.

Step ⑤ : Commit changes.

* Result:

Hence, creating account on Github
and uploading file completed
successfully.



Practical No. 5

* Aim: Write a different commands on GIT / GITHUB .

* Theory:

Git/Github are version control systems that allows multiple developer to collaborate on projects. It track changes made in repositories.

commands:

- ① 'git init': Initializes a new Git repository.
- ② "git clone" [repository URL]: Clones a repository from remote url to your local machine ,
- ③ 'git add [file]': Adds a file to staging area .
- ④ 'git commit -m "[commit message]"': Commits staged changes with descriptive message ,
- ⑤ 'git push' : Pushes committed changes to remote repository .
- ⑥ 'git pull' : fetches changes from a remote repository and merges them into current branch .



Date :

- ⑦ 'git branch': Lists existing branches or creates a new branch.
- ⑧ 'git checkout [branch]': Switches to different branch.
- ⑨ 'git merge [branch]': Merges changes from a specified branch into current branch.
- ⑩ 'git status': Displays status of working directory.
- ⑪ 'git fork': Creates copy of repository on Github under your own account.

* Result:

Hence, different commands for GIT/GITHUB executed successfully.



Date :

```
background-color: #f2f2f2;
}

header {
    background-color: #333;
    color: #fff;
    padding: 20px;
    text-align: center;
}

nav {
    background-color: #666;
    color: #fff;
    padding: 10px;
    text-align: center;
}

div section {
    padding: 20px;
    margin: 20px;
    background-color: #fff;
}

footer {
    background-color: #333;
    color: #fff;
    padding: 10px;
    text-align: center;
}

.highlight {
    color: #ff4500;
    font-weight: bold;
}

</style>
</head>
```



Date :

```
<body>
  <header>
    <h1> My Website </h1>
  </header>
  <nav>
    <span> Home <span> | <span> About </span> |
    <span> Contact </span>
  </nav>
  <div>
    <section>
      <h2> Welcome to my Website </h2>
      <p> This is a <span class = "highlight">
        demo </span> webpage designed
        using <span class = "highlight"> div
        and <span class = "highlight"> span
        </span> tags </p>
    </div>
  *
```

</body>

</html>

* Result:

Hence, webpage using div and span tags
designed successfully.

Practical No. 6



* Aim: Design a webpage using Div & span Tags.

* Theory:

Div:

In HTML, <div> element is generic container that allows you to group and organize content.

Span:

In HTML, element is an inline container used to apply styles or manipulate small pieces of content within a larger block of text.

* Program:

```
<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "UTF-8">
    <meta name = "viewport" content = "width =
        device-width , initial-scale = 1.0">
    <title> webpage design </title>
    <style>
        body{
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
```



Practical No. 7

* Aim: Design webpage using CSS.

* Theory:

CSS:

It stands for cascading style sheets. It's a styling language for web pages. It defines how HTML elements look and behave, including layout, colors, fonts, and spacing. Selectors target specific elements, properties define their appearance, and values set specific styles. It can be applied externally, internally or inline within HTML.

* Program:

```
<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "UTF - 8">
    <meta name = "viewport" content = "width =
        device-width, initial-scale = 1.0">
    <title> More Attractive Webpage Design </title>
    <style>
        body {
            font-family: 'Segoe UI', Tahoma,
            Geneva, Verdana, sans-serif;
            margin: 0;
        }
    </style>
</head>
<body>
    <h1> Welcome to My Website! </h1>
    <p> This is a sample webpage designed using CSS.
        It features a header, a main content area, and a footer.
        The content includes a heading, some descriptive text, and a
        button for navigation. The design is clean and modern, utilizing
        responsive design principles for better user experience on
        different devices.
    </p>
    <button> Go Back </button>
</body>
</html>
```

padding: 0;

background-color: #f9f9f9;

color: #333;

text-align: center;

}

• container {

max-width: 800px;

margin: 50px auto;

padding: 30px;

background-color: #fff;

border-radius: 10px;

box-shadow: 0 0 20px rgba(0,0,0,0.1);

}

h2 {

margin-bottom: 20px;

color: #333;

}

p {

color: #666;

line-height: 1.6;

}

button {

padding: 10px 20px;

background-color: #007bff;

color: #fff;

border: none;

border-radius: 5px;

cursor: pointer;

transition: background-color 0.3s ease;

}



button { hover {

background-color: #0056b3;

}

.feature {

margin-top: 40px;

display: flex;

justify-content: center;

align-items: center;

}

.feature img {

width: 100px;

border-radius: 50%;

margin-right: 20px;

box-shadow: 0 0 10px rgba(0,0,0,0.2);

}

.feature p {

text-align: left;

}

</style>

</head>

<body>

<div class = "container">

<h1> Welcome </h1>

<p> This website is designed
using CSS </p>

<button> Learn More </button>

<div class = "feature">

</div>

</div>

```
</body>  
</html>
```

* Result!

Hence, webpage designed successfully using
CSS .



* Aim: Write a HTML code for Redirection Button.

* Theory:

'

* Program:

index.html:

```

<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "UTF-8">
    <meta name = "viewport" content = "width =
        device-width, initial scale = 1.0">
    <title> Redirect Example </title>
</head>
<body>
    <a href = "pjtec.html"> <button> Go to
        PJTCE Page </button> </a>
</body>
</html>

```



Date :

pjlee.html

```
<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "UTF-8">
    <meta name = "viewport" content = "width=device-width, initial scale = 1.0">
    <title> PJLCE </title>
</head>
<body>
    <h1> Welcome to PJLCE </h1>
    <p> An Autonomous Institute </p>
</body>
</html>
```

* Result:

Hence, HTML code for Redirection Button executed successfully.



* Aim: Create a Docker file & project hosted on GITHUB.

* Theory:

Docker:

→ It's a platform that allows you to package, distribute and run applications in lightweight, portable containers. These containers include everything needed to run application, such as code, runtime system tools, libraries and settings.

steps to create Docker file:

Step ① : Create new directory for your project.

Step ② : Create your HTML file.

Step ③ : Write your Dockerfile.

Step ④ : Write Dockerfile content.

Step ⑤ : Build your image
of docker build -t my-html-app .

Step ⑥ : Run your Docker container
of docker run -d -p 8080:80 myhtmlapp



Date :

Step ⑦: Access your HTML file!

Open web browser and navigate to
`http://localhost:8080`

→ Steps to host project on Github:

Step ① : Create Github repository.

Step ② : Push code to repository.

Step ③ : Enable Github pages.

Step ④ : Access your project.

`https://<username>.github.io/<repository>`.

* Result:

Hence, docker file created and project hosted on github successfully.



Date :

Practical No. 10

* Aim: Design a webpage using Javascript.

* Theory:

JavaScript is a versatile programming language primarily used for adding interactivity and dynamic behavior to web pages. It is often executed in web browser, but it can also run on servers through platform like Node.js.

* Program:

```
<!DOCTYPE html>

<html> lang = "en">
  <head>
    <meta charset = "UTF-8" >
    <meta name = "viewport" content = "width =
      device-width , initial scale = 1.0" >
    <title> Toggle Text </title>
    <style>
      body {
        font-family : Arial , sans-serif ;
        text-align : center ;
      }
      #content {
        margin-top : 50px ;
      }
      button {
        padding : 10px 20px ;
        font-size : 16px ;
      }
    </style>
  </head>
  <body>
    <h1> Welcome to my Website ! </h1>
    <p> Click the button below to toggle the text ! </p>
    <button id = "toggle"> Click Me ! </button>
    <script>
      document.getElementById("toggle").addEventListener("click", function() {
        let content = document.getElementById("content");
        if (content.style.display === "block") {
          content.style.display = "none";
        } else {
          content.style.display = "block";
        }
      });
    </script>
  </body>
</html>
```



Date :

```
background-color: #007bff;
color: #fff;
cursor: pointer;
border-radius: 5px;

</style>

</head>
<body>
    <h1> Welcome to Toggle Text </h1>
    <div> id = "content">
        <p> id = "dynamicText" > Click the button
            to toggle text </p>
        <button onclick = "toggleText()"> Toggle Text
        </button>
    </div>
    <script>
        var textToggle = true;
        function toggleText() {
            var paragraph = document.querySelector
                ("#dynamicText");
            if (textToggle) {
                paragraph.textContent = "Text
                    changed to first option";
            }
            else {
                paragraph.textContent = "Text
                    changed to second option";
            }
            textToggle = !textToggle;
        }
    </script>
```



Date :

```
</body>  
</html>
```

* Result!

Hence , webpage using Javascript designed successfully .