



L O V E L Y  
P R O F E S S I O N A L  
U N I V E R S I T Y

**SEMINAR ON SUMMER TRAINING**

**Seven Week Training Report**

**On**

**Front End Web Development Training**

**Submitted By**

**MALIK SARTAJ MUSHTAQ**

**Registration No-12221774**

**Under the guidance of**

**Gokboru Tech Pvt Ltd.School**

**Of Computer Application**

**LOVELY PROFESSIONAL UNIVERSITY**

**(JUNE-JULY,2024)**

S.No	Title	Page
1.	Declaration by Student	3
2.	Training Certification from organization	4
3.	Acknowledgment	5
4. 4.1 4.2 4.3	Introduction of course Technical Learning Introduction of mini project and technical assignment. Details of mini-Project	6-7
5.	Interface Design	7-11
6.	Code snippet	12-38
7.	Bibliography	39
8.		

## **STUDENT DECLARATION**

**To whom it may concern**

**I, MALIK SARTAJ MUSHTAQ, Registration no-12221774 of the student, hereby declare that the work done by me on “Topic of the work” from June 2024 to July 2024, Year, is a record of original work for the partial fulfilment of the requirements for the award of the degree, degree name.**

**Name of the Student: Malik Sartaj Mushtaq**

**Registration No: 12221774**

Dated:30/09/2024

## **CERTIFICATE OF TRAINING:**



GOKBORU  
Tech

# CERTIFICATE OF COMPLETION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

*Malik Sartaj Mushtaq*

For successfully completing the 7-week Front End Web Development training, conducted by Gokboru Tech Pvt Ltd. This intensive program covered key aspects of front end technologies, including web design, user experience, and development strategies.

*Yash Raj*

Yash Raj  
SME



Muhammad Samir Akhtar  
DIRECTOR & CEO

#startupindia

STARTUP  
PUNJAB  
BUSINESS FIRST

START  
INUP



Recognised by

**ACKNOWLEDGEMENT :**

I am deeply appreciative of the internship opportunity provided by GOKBORU TECH PVT LTD, which has significantly contributed to my learning and professional growth. I consider myself fortunate to have been part of this experience and to have learned from the esteemed professionals guiding me throughout the internship. I extend my sincere gratitude to the Training and Placement Coordinator of the School of Computer Application at Lovely Professional University for facilitating this opportunity. I am especially thankful for the invaluable advice and guidance that ensured a productive and enriching internship experience.

**MALIK SARTAJ MUSHTAQ**

**12221774**

## **INTRODUCTION OF THE COURSE**

**In this Live course, you will start by learning the fundamentals of Git and Front-End Web Development. The course will then advance to cover two of the leading libraries—React JS and Node JS. You will have the opportunity to develop an industry-ready single-page web application. By the end of the course, you will complete a major project, such as creating a platform similar to Instagram or Facebook. Additionally, you will gain experience in deploying and maintaining applications using a comprehensive DevOps workflow. Various deployment methods will be explored, including AWS, Netlify, and GitHub Pages, along with CI/CD Pipelines like Travis-CI and GitHub Actions. You can attend this course from anywhere and ask questions without hesitation.**

### **Technical Learning:**

**1.Learn scripting language HTML.**

**2.Front-End Web UI Frameworks and Tools: Bootstrap 4 and above, responsive design, JavaScript components, CSS**

**3.Front-End Web Development**

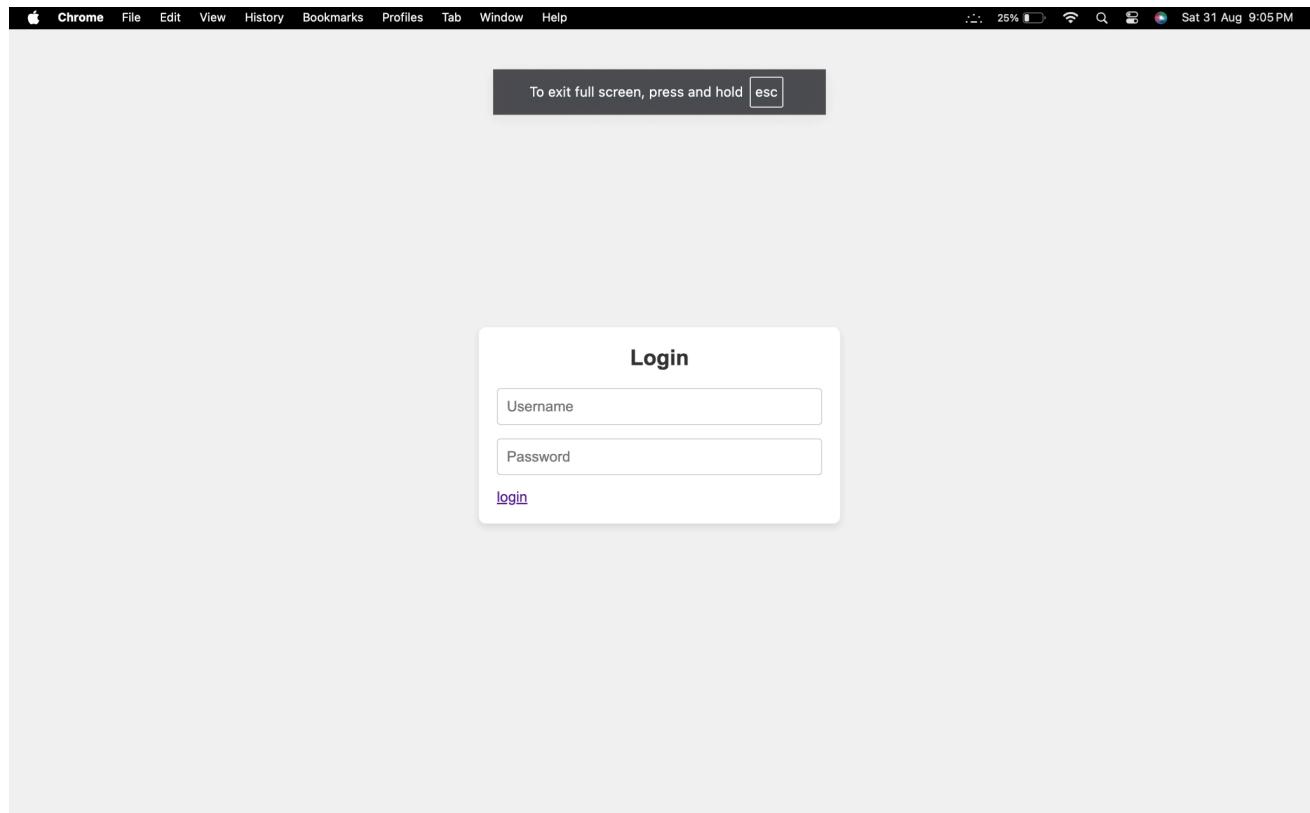
**4.1 minor project on a walkthrough to a web application creation.**

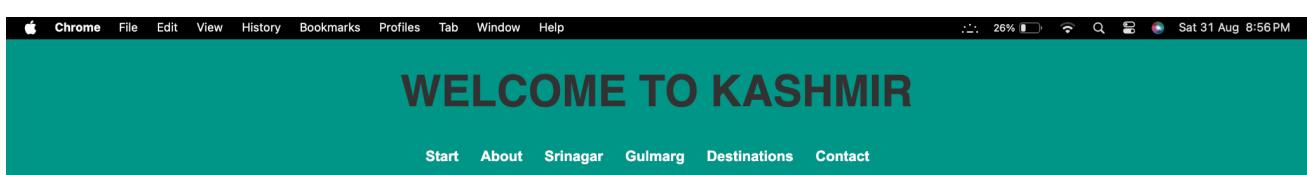
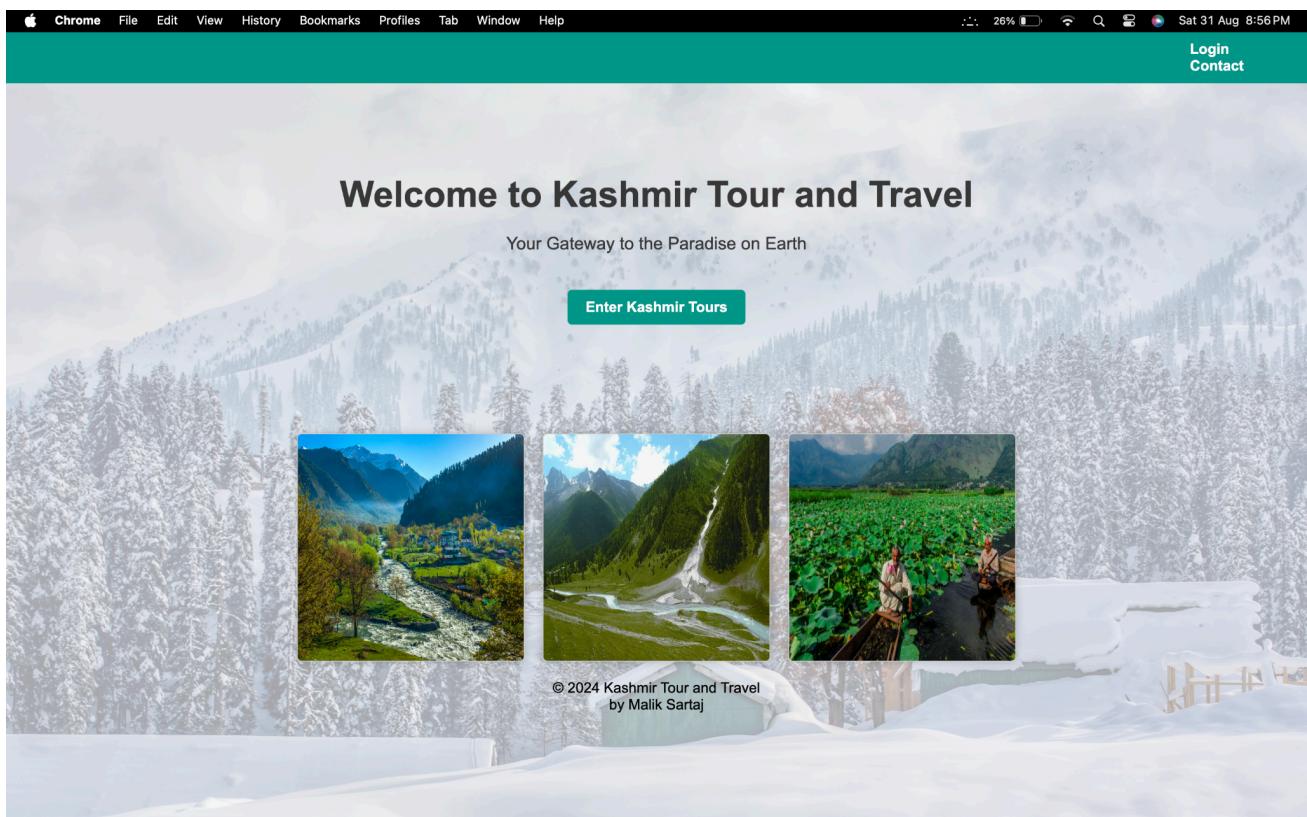
**5.1 major project based on real-world applications to put your theoretical knowledge to use**

**This project, which I have developed, is a 'Travel and Tourism Website' focused on Kashmir. Utilizing HTML, CSS, and JavaScript libraries like Bootstrap and jQuery, the site provides**

**detailed information about various travel destinations in Kashmir. Users can explore different attractions, find insights into their preferred sites, and discover local cuisine. Additionally, the website features a 'Favorites' section where users can bookmark their favorite destinations and experiences by clicking on a star icon next to each item.**

### **Interface Design:**





#### About Kashmir

Discover the beauty of Kashmir, known for its picturesque landscapes, serene lakes, and vibrant culture.

Kashmir, often hailed as the "Crown Jewel of India," is a realm where Mother Nature herself seems to have painted her most exquisite masterpiece. Nestled in the lap of the mighty Himalayas, this paradisiacal land is a symphony of unparalleled beauty, where every dawn and dusk is a painting, every valley a poem, and every lake a shimmering jewel. ...

#### Popular Destinations



Popular Destinations



Srinagar

Explore the capital city with its beautiful Dal Lake and Mughal Gardens. ...



Explore the capital city with its beautiful Dal Lake and Mughal Gardens. ...



Gulmarg



A screenshot of a web browser window. The title bar shows 'Chrome' and the date 'Sat 31 Aug 8:58 PM'. The main content area has a teal header with the text 'SRINAGAR, KASHMIR' in large white letters. Below the header is a navigation menu with links: 'Home', 'Overview', 'Attractions', and 'Contact'. The rest of the page is blank.

## Overview

Srinagar, often referred to as the "Venice of the East," is the summer capital of Jammu and Kashmir. It is known for its enchanting Dal Lake, houseboats, Mughal Gardens, and serene ambiance.

Srinagar, the jewel of Kashmir, is a city where nature's beauty and human artistry converge in harmonious splendor. Nestled amidst the Himalayan foothills, Srinagar boasts the enchanting Dal Lake, where shimmering waters reflect the surrounding snow-capped peaks and vibrant houseboats drift serenely. Mughal Gardens like Nishat Bagh and Shalimar Bagh offer meticulously manicured paradises with terraced lawns and exquisite flora. The city's historic old quarter showcases intricate woodwork and architectural marvels, while the warm hospitality of its people adds a layer of cultural richness to this breathtaking canvas. Srinagar is a destination that captivates the heart with its serene beauty and timeless charm.

## Top Attractions



### Dal Lake

Dal Lake is Srinagar's iconic attraction, famous for its shikara rides and houseboats. It's a serene oasis in the heart of the city.

It is often referred to as the "Jewel in the Crown of Kashmir," is a natural wonder that epitomizes the enchanting beauty of the region. This pristine, mirror-like lake is surrounded by lush, green meadows and towering mountains, creating a picturesque backdrop that seems straight out of a fairy tale. Houseboats, intricately designed with vibrant colors, line the shores, offering a unique and tranquil accommodation experience. A shikara ride on Dal Lake is a must-do activity, allowing you to immerse yourself in the serene ambiance, observe local life on floating gardens, and witness breathtaking sunsets that paint the sky in hues of orange and pink.



file:///Users/sartaj/Desktop/html sem4/dal\_lake\_booking.html



### Nishat Bagh

Nishat Bagh, one of the Mughal Gardens of Srinagar, offers breathtaking views of the Zabarwan Range and Dal Lake. It's a perfect spot for a leisurely stroll is a horticultural marvel that unfurls like a cascading green paradise against the backdrop of the Zabarwan Range and Dal Lake. This terraced garden, meticulously landscaped with vibrant flowerbeds, cascading fountains, and meticulously manicured lawns, offers a glimpse into the opulence of the Mughal era. As you stroll through Nishat Bagh's pathways, the soothing sound of water and the fragrance of blooming flowers envelop your senses, creating a serene and romantic atmosphere. It's a place where one can find solace and appreciate the timeless beauty that has been meticulously preserved for centuries.



file:///Users/sartaj/Desktop/html sem4/nishat\_bagh\_booking.html

Chrome File Edit View History Bookmarks Profiles Tab Window Help 26% Sat 31 Aug 8:58 PM

## CONTACT US

[Go To Home](#)

### Get in Touch

If you have any questions or inquiries, please feel free to contact us using the form below:

Name:

Email:

Message:

[Submit](#)



The screenshot shows a dark teal header bar with the title "GULMARG, KASHMIR" in large white capital letters. Below the title is a navigation menu with links for "Home", "Overview", "Attractions", and "Contact". The browser's top bar includes standard menu items like "File", "Edit", "View", "History", "Bookmarks", "Profiles", "Tab", "Window", and "Help", along with system status icons.

## Overview

Gulmarg, a paradise for nature lovers and adventure enthusiasts, is located in the Pir Panjal range. It's famous for its stunning landscapes and exciting activities. It is a veritable gem ensconced amidst the Himalayan grandeur, a poetic fusion of natural splendor and alpine sophistication. This idyllic retreat, nestled within the Pir Panjal range, boasts panoramic vistas of pristine meadows and towering peaks, where time seems to stand still. Renowned as a pristine sanctuary for all seasons, Gulmarg enchants visitors with its lush emerald expanses in summer and transforms into a winter wonderland, where glistening snow blankets the landscape. The Gulmarg experience extends beyond mere aesthetics; it beckons adventure enthusiasts with world-class skiing, trekking, and golfing opportunities, while the tranquil atmosphere soothes the soul. The Gulmarg Gondola, one of the highest cable cars globally, whiskers travelers to lofty altitudes, unveiling a tapestry of ethereal beauty and untamed wilderness. In Gulmarg, nature's symphony harmonizes with the melodies of leisure and recreation, crafting an unforgettable ode to serenity and grandeur.

## Top Attractions



This screenshot shows the same browser interface as the first one, but the main content area displays a large image of a green hillside with a road and a vehicle. The URL in the address bar is "gulmargkashmir.com/attractions".

## Top Attractions



### Gulmarg Meadows

Nestled within the bosom of the Pir Panjal range, Gulmarg Meadows beckon with an exquisite tapestry of nature's grandeur. These emerald expanses, resembling nature's own Persian rug, unfurl gracefully beneath azure skies. Verdant meadows, adorned with delicate wildflowers, roll gently towards the distant horizon, where snow-crowned peaks whisper secrets to the heavens. Here, the earth breathes tranquility, and the senses are serenaded by the symphony of rustling leaves and the delicate perfume of Alpine blooms. In this sylvan sanctuary, one finds refuge from the world's cacophony, embracing the unspoken poetry of the Himalayan meadows.



Nestled within the bosom of the Pir Panjal range, Gulmarg Meadows beckon with an exquisite tapestry of nature's grandeur. These emerald expanses, resembling nature's own Persian rug, unfurl gracefully beneath azure skies. Verdant meadows, adorned with delicate wildflowers, roll gently towards the distant horizon, where snow-crowned peaks whisper secrets to the heavens. Here, the earth breathes tranquility, and the senses are serenaded by the symphony of rustling leaves and the delicate perfume of Alpine blooms. In this sylvan sanctuary, one finds refuge from the world's cacophony, embracing the unspoken poetry of the Himalayan meadows.



#### Gulmarg Gondola

The Gulmarg Gondola, a marvel of modern engineering amid ancient peaks, offers a passage to celestial realms. Ascending to heights that graze the very edges of the Himalayan sky, this aerial journey unfolds a visual symphony of snow-draped summits, whispering pine forests, and boundless vistas. As the cable car soars higher, the world below transforms into a masterpiece of contrasts, where the verdant valleys merge with the icy grandeur of alabaster mountains. With each ascent, one transcends the earthly realm and enters a dreamscape where the sublime meets the surreal. Gulmarg Gondola rides are a

Chrome File Edit View History Bookmarks Profiles Tab Window Help 26% Sat 31 Aug 8:58 PM

## CONTACT US

[Go To Home](#)



### Get in Touch

If you have any questions or inquiries, please feel free to contact us using the form below:

Name:

Email:

Message:

[Submit](#)



## Some code snippets of my project:

### Enter.html:

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="=device-, initial-scale=1.0">

    <link rel="stylesheet" href="styles5.css">

    <title>Kashmir Tour and Travel</title>

</head>

<body>

    <header>

        <nav>

            <ul>

                <li><a href="login.html">Login</a></li>

                <li><a href="contact.html">Contact</a></li>

            </ul>

        </nav>

    </header>
```

```
<section class="hero">

    <h1>Welcome to Kashmir Tour and Travel</h1>

    <p>Your Gateway to the Paradise on Earth</p>

    <a href="ca4.html" class="explore-button">Enter Kashmir
Tours</a>

</section>

<section class="image-gallery">

    <div class="image">
        
    </div>

    <div class="image">
        
    </div>

    <div class="image">
        
    </div>

</section>

<footer>

    <p style="text-align: center;">&copy; 2024 Kashmir Tour
and Travel <br>
```

```

        <span style="text-align: center;">by Malik
Sartaj</span></p>

</footer>

</body>

</html>

```

### style5.css

```

body, h1, h2, h3, p, ul, li {

    margin: 0;
    padding: 0;
}

body {

    font-family: 'Arial', sans-serif;
    background-color: #f2f2f2;
}

```

```

body::before {

    content: '';
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
}

```

```
background: url('login_backgroung.jpg') center/cover
no-repeat;
opacity: 0.3;
z-index:-1; /* Adjust the opacity as needed */

}

header {
background-color: #009688;
color: #fff;
padding: 10px 0;
}

header nav ul {
list-style-type: none;
margin: 0;
padding-left: 91%;
}

header nav ul li {
display: inline;
margin-right: 20px;
}

header nav a {
```

```
text-decoration: none;  
  
color: #fff;  
  
font-weight: bold;  
}  
  
  
.hero {  
  
text-align: center;  
  
padding: 100px 0;  
  
color: #3c3939;  
}  
  
  
.hero h1 {  
  
font-size: 2.5rem;  
  
margin-bottom: 20px;  
}  
  
  
.hero p {  
  
font-size: 1.2rem;  
  
margin-bottom: 40px;  
}  
  
  
.explore-button {  
  
display: inline-block;  
  
padding: 10px 20px;  
  
background-color: #009688;
```

```
    color: #fff;
    text-decoration: none;
    font-weight: bold;
    border-radius: 5px;
    margin: 0 10px;
    transition: background-color 0.3s ease-in-out;
}

.explore-button:hover {
    background-color: #007e6e;
}

.image-gallery {
    display: flex;
    flex-wrap: wrap;
    justify-content: center;
    gap: 20px;
    padding: 20px;
}

.image {
    width: 250px;
    height: 250px;
    border: 1px solid #ccc;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
}
```

```
        overflow: hidden;  
  
        border-radius: 5px;  
  
        transition: transform 0.3s ease-in-out;  
    }  
  
  
.image img {  
  
    max-width: 100%;  
  
    height: 250px;  
}  
  
  
.footer {  
  
    background-color: #333;  
  
    color: #fff;  
  
    text-align: center;  
  
    padding: 20px 0;  
}
```

### **srinagar:**

```
<!DOCTYPE html>  
  
<html lang="en">  
  
<head>  
  
    <meta charset="UTF-8">  
  
    <meta name="viewport" content="width=device-width,  
initial-scale=1.0">  
  
    <link rel="stylesheet" href="styles3.css">  
  
    <title>Srinagar - Kashmir Tour</title>
```

```
</head>

<body>

    <header>

        <h1>Srinagar, Kashmir</h1>

        <nav>

            <ul>

                <li><a href="ca4.html">Home</a></li>

                <li><a href="#overview">Overview</a></li>

                <li><a href="#attractions">Attractions</a></li>

                <li><a href="contact.html">Contact</a></li>

            </ul>

        </nav>

    </header>

    <section id="overview">

        <h2>Overview</h2>

        <p>Srinagar, often referred to as the "Venice of the East," is the summer capital of Jammu and Kashmir. It is known for its enchanting Dal Lake, houseboats, Mughal Gardens, and serene ambiance.</p>

        <p>Srinagar, the jewel of Kashmir, is a city where nature's beauty and human artistry converge in harmonious
```

splendor. Nestled amidst the Himalayan foothills, Srinagar boasts the enchanting Dal Lake, where shimmering waters reflect the surrounding snow-capped peaks and vibrant houseboats drift serenely. Mughal Gardens like Nishat Bagh and Shalimar Bagh offer meticulously manicured paradises with terraced lawns and exquisite flora.

The city's historic old quarter showcases intricate woodwork and architectural marvels, while the warm hospitality of its people adds a layer of cultural richness to this breathtaking canvas. Srinagar is a destination that captivates the heart with its serene beauty and timeless charm.</p>

</section>

<section id="attractions">

<h2>Top Attractions</h2>

<div class="attraction">

<a href="dal\_lake\_booking.html">

<h3>Dal Lake</h3></a>

<p>Dal Lake is Srinagar's iconic attraction, famous for its shikara rides and houseboats.

It's a serene oasis in the heart of the city.</p>

<p>It is often referred to as the "Jewel in the Crown of Kashmir," is a natural wonder that epitomizes the enchanting beauty of the region. This pristine, mirror-like lake is

surrounded by lush, green meadows and towering mountains, creating a picturesque backdrop that seems straight out of a fairy tale.

Houseboats, intricately designed with vibrant colors,

line the shores, offering a unique and tranquil accommodation experience.

A shikara ride on Dal Lake is a must-do activity, allowing you to immerse

yourself in the serene ambiance, observe local life on floating gardens, and witness

breathtaking sunsets that paint the sky in hues of orange and pink.</p>

</div>

<div class="attraction">

<a href="nishat\_bagh\_booking.html">



<h3>Nishat Bagh</h3></a>

<p>Nishat Bagh, one of the Mughal Gardens of Srinagar, offers breathtaking views of the Zabarwan Range

and Dal Lake. It's a perfect spot for a leisurely stroll is a horticultural marvel

that unfurls like a cascading green paradise against the backdrop of the Zabarwan

Range and Dal Lake. This terraced garden, meticulously landscaped with vibrant

flowerbeds, cascading fountains, and  
meticulously manicured lawns, offers a glimpse  
into the opulence of the Mughal era. As you  
stroll through Nishat Bagh's pathways,  
the soothing sound of water and the  
fragrance of blooming flowers envelop your  
senses, creating a serene and romantic  
atmosphere. It's a place where one can  
find solace and appreciate the timeless  
beauty that has been meticulously  
preserved for centuries.</p>

</div>

<!-- Add more attractions -->

</section>

</frames>

<footer>

<p>&copy; 2023 Kashmir Tour and Travel by Sahil Hamid</p>

<p style="text-align: center;"><a href="contact.html"><h4>Contact Us</h4></a></p>

</footer>

</body>

</html>

## **GULMARG:**

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width,
initial-scale=1.0">

    <link rel="stylesheet" href="styles3.css">

    <title>Gulmarg - Kashmir Tour</title>

</head>

<body>

    <header>

        <h1>Gulmarg, Kashmir</h1>

        <nav>

            <ul>

                <li><a href="ca4.html">Home</a></li>

                <li><a href="#overview">Overview</a></li>

                <li><a href="#attractions">Attractions</a></li>

                <li><a href="contact.html">Contact</a></li>

            </ul>

        </nav>

    </header>

    <section id="overview">
```

## <h2>Overview</h2>

<p>Gulmarg, a paradise for nature lovers and adventure enthusiasts, is located in the Pir Panjal range.

It's famous for its stunning landscapes and exciting activities.

It is a veritable gem ensconced amidst the Himalayan grandeur,

is a poetic fusion of natural splendor and alpine sophistication.

This idyllic retreat, nestled within the Pir Panjal range,

boasts panoramic vistas of pristine meadows and towering peaks,

where time seems to stand still. Renowned as a pristine sanctuary

for all seasons, Gulmarg enchants visitors with its lush emerald expanses

in summer and transforms into a winter wonderland, where glistening snow blankets the landscape.

The Gulmarg experience extends beyond mere aesthetics; it beckons adventure enthusiasts

with world-class skiing, trekking, and golfing opportunities, while the tranquil atmosphere

soothes the soul. The Gulmarg Gondola, one of the highest cable cars globally,

whisks travelers to lofty altitudes, unveiling a tapestry of ethereal beauty

and untamed wilderness. In Gulmarg, nature's symphony harmonizes with the melodies of leisure and recreation, crafting an unforgettable ode to serenity and grandeur.</p>

</section>

```
<section id="attractions">

    <h2>Top Attractions</h2>

    <div class="attraction">

        <a href="gulmarg_meadows_booking.html">

            <h3>Gulmarg Meadows</h3></a>

        <p>Nestled within the bosom of the Pir Panjal range,
Gulmarg Meadows beckon
            with an exquisite tapestry of nature's grandeur.
These emerald expanses,
            resembling nature's own Persian rug, unfurl
gracefully beneath azure skies.

        Verdant meadows, adorned with delicate
wildflowers, roll gently towards the
            distant horizon, where snow-crowned peaks
whisper secrets to the heavens.

        Here, the earth breathes tranquility, and the
senses are serenaded by the
            symphony of rustling leaves and the delicate
perfume of Alpine blooms.

        In this sylvan sanctuary, one finds refuge
from the world's cacophony.
```

embracing the unspoken poetry of the  
Himalayan meadows.</p>

</div>

<div class="attraction">

<a href="gulmarg\_gondola\_booking.html">

<h3>Gulmarg Gondola</h3></a>

<p>The Gulmarg Gondola, a marvel of modern engineering  
amid ancient peaks,

offers a passage to celestial realms. Ascending  
to heights that graze

the very edges of the Himalayan sky, this aerial  
journey unfolds a visual

symphony of snow-draped summits, whispering pine  
forests, and boundless vistas.

As the cable car soars higher, the world below  
transforms into a masterpiece

of contrasts, where the verdant valleys merge  
with the icy grandeur of alabaster

mountains. With each ascent, one transcends  
the earthly realm and enters a

dreamscape where the sublime meets the  
surreal. Gulmarg Gondola rides are a

celestial odyssey, an ethereal dance between  
earth and sky, where nature's

grandeur unfurls in all its majesty.</p>

</div>

<!-- Add more attractions -->

```
</section>

<footer>
    <p>&copy; 2023 Kashmir Tour and Travel by Sahil Hamid</p>
    <p style="text-align: center;"><a href="contact.html"><h4>Contact Us</h4></a></p>
</footer>
</body>
</html>
```

## **TECHNOLOGIES LEARNT:**

### **Technology Learnt**

**It had 16 units which was further divided into chapters and then topics so during my whole 7- week course I learned the following:**

#### **1st Week:**

##### **1.)Introduction to HTML and CSS**

**HTML is the standard markup language used to create the structure of web pages.**

**It defines the content and its organization on a webpage.**

**CSS is used to style and layout web pages — for example, to alter the colors, fonts, and spacing, or to arrange content on the page.**

- Introduction to HTML (Structure, Tags, Elements)**

**HTML, or Hypertext Markup Language, is the foundational language used to create**

**and structure content on the web. It provides the basic framework of a webpage, allowing developers to define elements such as headings, paragraphs, links, images, and more. HTML uses tags, which are enclosed in angle brackets (e.g.,<tag>), to define elements. These tags come in pairs, with an opening tag (e.g.,<p>) and a closing tag (e.g.,</p>), which surrounds the content. Each tag represents a different type of content, like <h1> for a heading or <a> for a link. HTML elements can be nested within each other to create complex layouts and structures. Although HTML primarily focuses on content structure, it works in tandem with CSS and JavaScript to create visually appealing and interactive web pages. Understanding HTML is essential for anyone looking to build websites or engage in web development.**

- **More HTML (Forms, Tables, Semantics)**

**HTML provides a range of elements that extend beyond basic content structure, including forms, tables, and semantic tags. Forms in HTML are used to collect user input and typically include elements like text fields, checkboxes, radio buttons, and submit buttons. The form element(<form>) wraps around input elements and specifies how the data should be submitted, either via GET or POST methods. Tables, defined with the<table> tag, are used to display data in a grid format with rows<tr>, columns(<td>), and headers(<th>), making it easier to organize and present structured data**

**Semantic HTML is about using HTML tags that have meaningful names to describe their purpose. Tags like <header>,<footer>,<article> and<section> provide a clearer structure for both developers and browsers, enhancing accessibility and search engine optimization. These tags define the different parts of a webpage, making the content easier to understand and navigate. By utilizing forms, tables, and semantic tags, HTML enables developers to create more interactive, structured, and accessible web pages.**

- **Introduction to CSS (Syntax, Selectors, Box Model)**

**CSS, or Cascading Style Sheets, is a language used to control the appearance and layout of HTML elements on a webpage. It allows developers to apply styles, such as colors, fonts, spacing, and positioning, to create visually appealing and responsive designs. The basic syntax of CSS consists of selectors and declarations. Selectors target the HTML elements you want to style, while declarations specify the styles to apply. A declaration consists of a property (e.g.,color) and a value (e.g.,blue), enclosed in curly braces.**

**CSS offers a variety of selectors to target elements, such as class selectors(.class), ID selectors(#id), and more complex ones like attribute selectors or pseudo-classes. The box model is a fundamental concept in CSS that describes how elements are structured on a page. Every HTML element is treated as a rectangular box, consisting of content, padding, border, and margin. Understanding the box model is crucial for managing the spacing and layout of elements. CSS provides the flexibility to create intricate designs, making it an essential tool for modern web development.**● **CSS Layout (Flexbox, Grid)**

**CSS layout techniques like Flexbox and Grid revolutionize how developers create complex and responsive web designs. Flexbox, short for Flexible Box Layout, is a one-dimensional layout system that helps in distributing space along a single row or column. It is ideal for creating layouts where the alignment, spacing, and distribution**

of elements needed to be managed dynamically. Flexbox simplifies the process of centering items, adjusting the size of elements based on available space, and creating flexible, responsive designs without using float or positioning hacks.

CSS Grid, on the other hand, is a two-dimensional layout system that provides even more control over the layout structure. With Grid, developers can create layouts using rows and columns, allowing for more complex designs with precise control over the placement of elements. Grid enables the creation of entire page layouts with ease, from simple grids to intricate and dynamic layouts. Both Flexbox and Grid are powerful tools that complement each other, making CSS layouts more intuitive and versatile, especially in responsive web design.

- **CSS Positioning (Relative, Absolute, Fixed)**

CSS positioning is a powerful feature that allows developers to control the placement of elements on a webpage with precision. The position property in CSS defines how an element is positioned relative to its parent, the viewport, or other elements. The three most commonly used positioning types are relative, absolute, and fixed.

**Relative Position:** positions an element relative to its normal position in the document flow. When you set an element's position to relative, you can move it around using the top, right, bottom, and left properties, without affecting the layout of surrounding elements.

**Absolute Position:** completely removes an element from the normal document flow and positions it relative to its closest positioned ancestor (i.e., an ancestor element with a position value other than static). This allows you to place the element anywhere on the page, but it won't influence or be influenced by other elements in the layout.

**Fixed Position:** locks an element in place relative to the browser window or viewport, meaning it stays in the same position even when the page is scrolled. This is often used for sticky headers, footers, or other elements that need to remain visible regardless of scrolling.

## 2.) Advanced CSS

- **Responsive Design Principles**

Responsive design in advanced CSS is a crucial principle for ensuring that websites adapt seamlessly to different screen sizes and devices. At its core, responsive design aims to create web pages that look and function well on a variety of devices, from desktops to smartphones. This is achieved using techniques like fluid grids, flexible images, and media queries.

Fluid grids use relative units, like percentages, instead of fixed units like pixels, allowing elements to resize proportionally to the screen. Flexible ranges also resize dynamically, ensuring that visual content fits within the layout without breaking it. Media Queries are a powerful CSS tool that enables developers to apply different styles based on the screen size, orientation, or other device characteristics. By setting breakpoints, developers can adjust layouts, font sizes, and other elements to suit various devices.

Advanced responsive design goes beyond simple resizing, incorporating techniques like CSS Grid and Flexbox for creating fluid layouts, as well as leveraging view port unit and rem for scalable typography. Other considerations include touch-friendly design, ensuring tap targets are appropriately sized and spaced, and performance

**optimization, so that images and resources load efficiently on slower networks. By following these responsive design principles, developers can create websites that provide an optimal user experience across all devices and screen sizes.**

- **Media Queries and Viewport Meta Tag**

The **viewport meta tag and media queries are crucial components in responsive web design, enabling websites to adapt seamlessly to various screen sizes and devices.**

The **viewport meta tag, placed within the <head> section of an HTML document, controls the layout of a webpage on different devices, particularly mobile devices. By setting width=device-width, the tag ensures that the page width matches the device's screen width, preventing automatic scaling that can make a site appear too small. The initial-scale=1.0 parameter maintains the default zoom level when the page is loaded, ensuring a consistent user experience across different devices.**

**Media queries, on the other hand, allow developers to apply specific CSS styles based on the characteristics of the user's device, such as screen width, height, orientation, and resolution. By using media queries, you can tailor the layout, font sizes, and other design elements to fit devices ranging from mobile phones to desktop monitors. For example, a media query might apply one set of styles for screens smaller than 600px wide and another set for larger screens. Together, the viewport meta tag and media queries enable web developers to create flexible, responsive designs that provide an optimal viewing experience on any device.**

- **CSS Preprocessors (e.g., Sass)**

**CSS preprocessors, like Sass (Syntactically Awesome Stylesheets), are powerful tools that extend the capabilities of traditional CSS, making it more efficient and maintainable for developers. They introduce features such as variables, nested rules, mixins, and functions, which streamline the styling process and promote code reuse. For instance, variables allow you to store values like colors or font sizes in one place, making global updates easy. Nested rules in Sass mirror the structure of HTML, improving code readability and organization.**

**Mixins in Sass enable the creation of reusable code blocks, reducing redundancy, while functions allow for more complex calculations and operations within your stylesheets. Additionally, Sass preprocessors support partials, which help in breaking down large CSS files into smaller, more manageable pieces, enhancing modularity. Once written, Sass code is compiled into standard CSS, ensuring compatibility with all browsers. By leveraging a CSS preprocessor like Sass, developers can write more dynamic, scalable, and maintainable stylesheets, ultimately improving the development workflow.**

- **CSS Frameworks (e.g., Bootstrap, Tailwind CSS)**

**CSS frameworks, such as Bootstrap and Tailwind CSS, are collections of pre-designed CSS components and utilities that simplify the process of building responsive and visually appealing web interfaces. These frameworks provide a foundation of reusable code, enabling developers to create consistent designs quickly without writing CSS from scratch.**

**Bootstrap is one of the most popular CSS frameworks, known for its extensive library of pre-built components like navigation bars, buttons, forms, and grids. It follows a mobile-first approach, ensuring that designs are responsive out of the box. Bootstrap also includes JavaScript components for interactive elements, making it a**

**comprehensive solution for building modern websites and web applications. By using Bootstrap's grid system, developers can create flexible layouts that adapt to different screen sizes, reducing the need for custom CSS**

**Tailwind css, on the other hand, takes a utility-first approach. Instead of providing pre-styled components, Tailwind offers a wide range of utility classes that can be combined to create custom designs. This approach gives developers more control and flexibility over the design, while still speeding up the development process. Tailwind's emphasis on utility classes allows for quick prototyping and encourages the reuse of small, single-purpose classes to achieve complex layouts.**

- **CSS Animations and Transitions**

**CSS animations and transitions are powerful techniques that enhance the user experience by adding dynamic effects to web elements. They allow for smooth changes in style properties, making interactions more engaging and visually appealing.**

**Css Transition enable elements to change from one style to another over a specified duration. This effect is achieved by defining a transition property, which specifies the CSS properties to animate, along with the duration, timing function, and delay. For example, you can create a smooth color change or size adjustment when a user hovers over a button. Transitions are relatively straightforward and ideal for simple, state-based changes, such as when elements move or change color on user interaction.**

**Css Animation offers more control and complexity compared to transitions. They involve defining keyframes, which are the stages of the animation, and then specifying how styles should change at each stage. Animations can run continuously, loop, or occur once, depending on the defined behavior. This makes them suitable for more intricate effects, like a loading spinner or a bouncing ball. CSS animations provide greater flexibility, allowing you to animate multiple properties simultaneously and create complex sequences of movements.**

**Both CSS animations and transitions enhance user interaction and can be used to draw attention to elements, guide users through actions, or simply add a touch of visual flair. By incorporating these techniques thoughtfully, developers can create more engaging and responsive user interfaces.**

**3rd Week :**

### **3. JavaScript Fundamentals**

- **Introduction to JavaScript (Variables, Data Types, Operators)**

**JavaScript is a powerful programming language used to add interactivity and dynamic features to web pages, and understanding its core elements—variables, data types, and operators—is fundamental for effective coding. Variables in JavaScript are used to store data values and can be declared using var, let, or const. While var is the traditional keyword, let and const are preferred for their block-scoping capabilities, with let allowing for reassignment and const creating immutable variables. Data types in JavaScript define the kind of data a variable can hold, including strings for text, numbers for numerical values, booleans for true/false values, objects for collection of properties, arrays for ordered lists, and special types like undefined and null for uninitialized and intentionally absent values, respectively.**

**Operators in JavaScript perform operations on variables and values and come in**

**several types. Arithmetic operators such as + and - handle basic mathematical operations, while assignment operators like = and += assign and update values. Comparison operators, including == and ===, compare values and return boolean results, and logical operators such as && and || perform logical operations to determine boolean outcomes. Mastering these elements enables developers to write effective JavaScript code, creating responsive and interactive web applications.**

- **Control Flow (Conditionals, Loops)**

**JavaScript fundamentals include control flow mechanisms such as conditionals and loops, which are essential for directing the execution of code based on specific conditions and for performing repetitive tasks. Conditionals allow developers to execute different blocks of code depending on whether certain conditions are met. The most common conditional statements are if, else-if, and else, which evaluate expressions and determine which code block to run. For example, an if statement can check if a user's age is over 18 to decide whether to grant access to certain content. The switch statement is another form of conditional used to handle multiple potential values of a single expression, offering a more organized way to handle multiple cases. Loops, on the other hand, are used to repeat a block of code multiple times, which is useful for iterating over arrays or performing repeated tasks. JavaScript supports several types of loops, including the for loop, which iterates a set number of times based on a condition; the while which continues as long as a specified condition remains true; and the do while loop, which ensures that the code block executes at least once before checking the condition. Each of these loops provides a way to efficiently handle repetitive operations and manage control flow in JavaScript programs. Mastering these control flow constructs allows developers to build more dynamic and responsive applications by handling various scenarios and processing data efficiently.**

- **Functions and Scope**

**In JavaScript, functions and scope are fundamental concepts that play a crucial role in writing organized and efficient code. Functions are reusable blocks of code designed to perform specific tasks. They can be declared using function declarations, which are hoisted and available throughout their scope, or function expressions, which are not hoisted and only available after their definition. Functions can accept parameters and return values, allowing for flexible and modular code. They help avoid redundancy by encapsulating logic that can be called multiple times from different parts of the code. Scope refers to the accessibility of variables and functions in different parts of the code. JavaScript uses lexical scoping, meaning the scope of a variable is determined by its location within the source code. There are two primary types of scope: global and local . global scope encompasses the entire codebase, meaning variables declared in the global context are accessible from any part of the code. Local scope, on the other hand, is confined to the block or function where a variable is declared. Variables defined within a function or block are not accessible outside of it, which helps in managing and encapsulating data.**

- **Arrays and Objects**

**In JavaScript, arrays and objects are fundamental data structures that allow developers to store and organize data efficiently. Arrays are ordered collections of elements, where each element is assigned a numeric index starting from 0. They are**

**commonly used to store lists of data, such as numbers, strings, or even other arrays and objects. Arrays come with various built-in methods, such as push to add elements, pop to remove elements, and foreach or map for iterating through the array's elements, making it easy to manipulate and process data.**

**Objects, on the other hand, are collections of key-value pairs, where each key is a unique identifier (often referred to as a property), and the value can be any data type, including arrays and other objects. Objects are used to represent more complex data structures, such as a user profile or a product with multiple attributes. You can access and modify object properties using dot notation (e.g., object.property) or bracket notation (e.g., object["property"]).**

**Both arrays and objects are versatile and integral to JavaScript programming. Arrays are ideal for managing lists and sequences of data, while objects provide a flexible way to structure more complex data with meaningful keys. Mastering these data structures is essential for effectively handling and organizing data in JavaScript applications.**

- **DOM Manipulation (Selecting Elements, Manipulating Styles and Content)**

**DOM manipulation is a key aspect of JavaScript that allows developers to interact with and modify the structure, style, and content of web pages dynamically. The Document Object Model (DOM) represents the HTML of a web page as a tree of nodes, where each element, attribute, and piece of content is a node that can be accessed and changed using JavaScript.**

**DOM manipulation also allows for the addition or removal of elements from the page, making web pages more dynamic and interactive. Understanding these techniques enables developers to create responsive web pages that adjust in real-time to user actions or other events.**

**Selecting elements is the first step in DOM manipulation. JavaScript provides methods like get element by id to select an element by its ID and queryselector to select elements example, document.getelementbyid("header") would select an element with the ID "header", while document.querySelector("btn") would select the first element with the class "btn". Methods like query selector all allow you to select multiple elements that match a selector.**

## **4th Week**

### **4. Intermediate JavaScript**

- **Events and Event Listeners**

**In JavaScript, events are actions or occurrences like clicks, key presses, or page loads that happen in the browser, while event listeners are functions that respond to these events, making web pages interactive. To create an event listener, you first select the element you want to monitor, then use the addEventListener method to specify the event type, such as "click" or "keydown", and define the function that will execute when the event occurs. For example, you might add a click event listener to a button so that when the button is clicked, an alert message appears. Event listeners can be applied to various events, from mouse movements to form submissions, and multiple listeners can be attached to a single element. They can also be removed when no longer needed using removeEventListener. By using events and event listeners, developers can build web applications that dynamically respond to user actions,**

enhancing interactivity and user experience.

- **Form Validation with JavaScript**

Form validation with JavaScript is a crucial technique for ensuring that users provide the correct and complete information in forms before submitting them. By validating forms on the client side, you can improve the user experience by catching errors early, reducing the need for server-side checks, and preventing incomplete or incorrect data from being sent.

To validate a form, you typically check the values entered by the user against certain criteria, such as ensuring that required fields are not left empty, email addresses are in the correct format, and passwords meet strength requirements. JavaScript provides several methods for accessing form elements and their values, like `document.getelementbyid` or `document.`, which you can use to retrieve user input.

JavaScript also allows for real-time validation, where feedback is provided as the user fills out the form. This can include showing error messages or changing the styling of input fields to indicate whether the entered data is valid or not.

- **Asynchronous JavaScript (Callbacks, Promises)**

Asynchronous JavaScript enables developers to perform tasks like fetching data, reading files, or handling timers without blocking the main thread, ensuring that the rest of the code can continue running smoothly. Callbacks are one of the earliest methods used for handling asynchronous operations, where a function is passed as an argument and executed once a task is complete. While effective, callbacks can lead to "callback hell," a situation where deeply nested functions make code difficult to read and maintain. To improve this, Promises were introduced, offering a more structured way to manage asynchronous operations. A Promise represents the eventual completion or failure of a task and can be handled using `.then()` for success and `.catch()` for errors, making the code more linear and easier to follow. By using callbacks and promises, developers can create responsive and efficient web applications that handle tasks concurrently without blocking the main execution thread.

- **Introduction to ES6+ (Arrow Functions, Destructuring, Modules)**

ES6, or ECMAScript 2015, introduced several features that have improved JavaScript development by making the code more concise and manageable. Arrow function provide a shorter syntax for writing functions and automatically bind the `this` value from their surrounding context, making them ideal for callbacks and simplifying function expressions. Destructuring allows developers to extract values from arrays or properties from objects and assign them to variables in a cleaner, more concise way, reducing redundancy. Additionally, modules enable the organization of code into separate files, allowing for the import and export of functionality, which helps manage large codebases and keeps the global scope clean. These ES6+ features are widely used in modern JavaScript, enhancing the development process and resulting in more efficient and maintainable code.

Additionally, ES6+ features help align JavaScript with modern development practices, fostering better collaboration and scalability in projects. Arrow function simplify function syntax and reduce verbosity, making the codebase more readable and easier to manage. Destructuring enhances the clarity of data extraction and manipulation,

**enabling developers to work with complex data structures more intuitively. Modules promote modular programming, facilitating code reuse and maintenance by allowing developers to break down functionality into distinct, manageable parts. These improvements not only streamline development but also enhance performance and maintainability, making ES6+ features integral to contemporary JavaScript development.**

**• Project: Build a To-Do List App**

**Build using html,css and javascript**

**5th Week**

**5. Front-End Frameworks**

**• Introduction to React.js**

**React.js is a widely-used JavaScript library created by Facebook for building user interfaces, especially for single-page applications. It enables developers to construct interactive and dynamic UIs using reusable components that manage their own state, making the development process more efficient. React's declarative approach allows developers to describe how the UI should look based on the application's state. The library relies on a virtual DOM, which is a lightweight representation of the actual DOM. When the state changes, React updates the virtual DOM and then efficiently synchronizes these updates with the real DOM, enhancing performance by reducing direct DOM manipulation. React promotes a component-based architecture, where the UI is divided into small, reusable pieces, simplifying development, testing, and maintenance. With its component-based design and robust ecosystem, React.js supports modern web development and provides a scalable solution for building complex applications.**

**• Components and Props in React** In React, components and props are fundamental concepts that play a crucial role in

**building and managing user interfaces. components are the building blocks of a React application. They are reusable pieces of code that encapsulate both the structure (HTML) and behavior (JavaScript) of a part of the UI. Components can be either class-based or functional . Class-based components use ES6 classes and can manage their own state and lifecycle methods, while functional components, introduced with React Hooks, are simpler and allow state management and side effects through hooks.**

**Props (short for "properties") are used to pass data from parent components to child components. They enable component reusability and customization by allowing components to receive dynamic data and behavior. Props are read-only, meaning that a child component cannot modify its own props but can use them to render content or trigger functions. For example, a parent component might pass a user's name as a prop to a child component that displays a greeting message.**

**By using components and props effectively, developers can create modular, maintainable, and dynamic user interfaces in React applications, ensuring that each component is responsible for a specific piece of the UI and can be easily reused and updated.**

**• State and Lifecycle Methods**

**In React, state and lifecycle methods are essential for managing data and controlling component behavior throughout their existence. State refers to the internal data of a component that can change over time and affect how the UI is rendered. In class-**

**based components, state is managed using the this.state object and updated will this.setState(), while in functional components, the useState hook provides a way to declare and manage state.** For example, state might be used to track user input or toggle a feature on and off. Lifecycle in class-based components, such as componentDidMount, componentDidUpdate , and componentWillUnmount, allow developers to run code at specific stages of a component's lifecycle, such as when it is first rendered or about to be removed. Functional components use the useEffect hook to handle side effects and manage cleanup tasks. Together, state and lifecycle methods enable React components to respond to changes, perform necessary operations at various points in their life, and maintain a dynamic and interactive user interface.

- **React Router (Client-Side Routing)**

React Router is a powerful library used in React applications to handle client-side routing, enabling seamless navigation between different views or pages without requiring a full page reload. It allows developers to create a single-page application (SPA) where the content updates dynamically based on the user's interactions with the application. React Router provides components like <BrowserRouter>,<router><link>, which manage the routing logic and URL changes within the app. The <BrowserRouter> component wraps the entire application and enables the use of HTML5 history API for cleaner URLs, while the <route> component defines the path and component to render for each route. The <link> component is used to navigate between different routes, replacing traditional anchor tags with React-friendly navigation. By using React Router, developers can build complex and responsive applications with multiple views, improve user experience by providing instant navigation, and maintain a smooth, single-page application experience.

- **Project: Building a Single Page Application (SPA) using React**

## 6th Week

### 6 Advanced React and State Management

- **React Hooks (useState, useEffect, etc.)**

React Hooks are a set of functions introduced in React 16.8 that enable developers to use state and other React features within functional components, which previously required class-based components. The useState hook allows functional components to manage state by declaring state variables and providing a function to update them, simplifying state management and re-rendering. For example, const [count, setCount] = useState(0) initializes a state variable and a function to update it, ensuring the component re-renders with the new state. The useEffect hook manages side effects, such as data fetching or subscriptions, by running a function after the component renders. Dependencies can be specified to control when the effect runs, optimizing performance. For instance, useEffect(() => {fetchData();}, []); runs once when the component mounts. React Hooks also include useContext for accessing context, useSelector for managing complex state logic, and use and useCallback for performance optimization. These hooks provide a more streamlined and readable approach to handling state and side effects, enhancing both development efficiency and code maintainability.

- **Context API for State Management**

The context API in React provides a way to manage state and share data across a component tree without having to pass props down manually through each level of the hierarchy. This feature is particularly useful for handling global state that needs to be accessed by multiple components at different levels. Using the Context API involves creating a context object with `react.createContext()`, which provides a provider component and a consumer component. The provider component wraps around the part of the component tree where you want the context to be accessible and supplies the context value. The consumer component or the `useState` hook can then be used within the tree to access the context value. This approach simplifies state management and reduces the complexity of prop drilling, making it easier to maintain and scale applications by centralizing the state and avoiding repetitive prop passing.

- **Redux Fundamentals**

Redux is a predictable state management library for JavaScript applications, often used with React but applicable to other frameworks as well. It helps manage the state of an application in a centralized store, making it easier to maintain and debug. The core principles of Redux include having a single source of truth for the state, ensuring that the state is read-only, and using pure functions called reducers to manage state changes.

In Redux, the state of the application is stored in a single JavaScript object called the store. Actions are plain JavaScript objects that describe changes or events in the application, and they are dispatched to the store. Reducers are functions that take the current state and an action as arguments, and return a new state based on the action's type. This ensures that state transitions are predictable and consistent.

- **Redux Middleware (Thunk, Saga)**

Redux Thunk and Redux Saga are middleware solutions used with Redux to handle asynchronous actions and side effects, enabling actions like data fetching and complex business logic. By following Redux's principles and using its middleware, developers can build scalable and maintainable applications with predictable state management.

Moreover, Redux Thunk and Redux Saga are middleware libraries used in conjunction with Redux to handle asynchronous operations and manage side effects in a more structured way.

Redux Thunk is a middleware that allows action creators to return functions instead of plain action objects. This enables the handling of asynchronous logic, such as API calls or complex computations, within action creators. For example, an action creator using Redux Thunk might dispatch a loading action before making an API request and then dispatch a success or failure action based on the response. This approach simplifies dealing with async actions by keeping them within the action creators and maintaining a clear flow of data.

Redux Saga is another middleware that provides a more advanced way to handle side effects using generator functions. Redux Saga allows for the management of complex async flows and concurrency by writing sagas, which are generator functions that can pause and resume their execution. This makes it easier to handle actions like debouncing, throttling, and orchestrating multiple concurrent operations. Sagas are typically used to handle more intricate business logic and interactions with external

systems in a more readable and maintainable manner.

## **BIBLIOGRAPHY:**

- 1) Gokboru Tech for React Learning and getting experience.**
- 2) React documentation and manuals for learning how to use their tools.**
- 3) Project images are the screenshot of the project.**
- 4) Code was written and helped by Trainer Mr. Yash Raj.**
- 5) The official React documentation provides comprehensive guides and tutorials on using React, including basic concepts, hooks, and advanced**

Thank you 😊