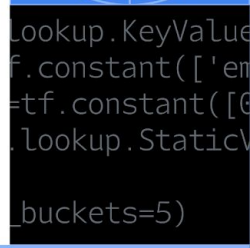
Google Developer Groups

Session starting soon

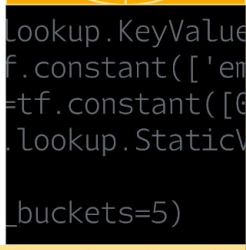


Google Developer Groups

Welcome To Day <5/>

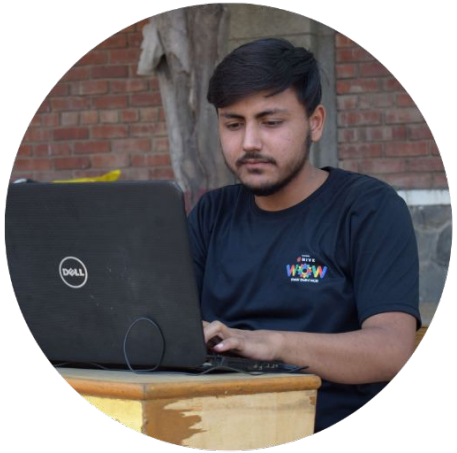
<TECH WINTER BREAK/>

Level up your skills during winter chillllsss



Google Developer Groups

About the Mentor



Sagar Kumar Jha is an accomplished professional with diverse experience across **high-impact organizations.**

- Interned at **DRDO**, focusing on **APK & API Security**.
- Interned at the **Special Protection Groups(SPG)**, specializing in **Radar Communication Systems**.
- Interned at **NITI Aayog**, contributing to **Software Engineering projects**.

Google Developer Groups

About the Instructor

Divyansh Raj is a skilled professional currently serving as the Technical Lead of GDGC and an intern at Delhi Government University, showcasing his expertise in technical leadership and project execution.

Google Developer Groups

Connect With Us !

Sagar Kumar Jha
(Mentor)

Divyansh Raj
(Instructor)

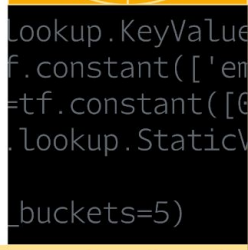
```
lookup.KeyValue
f.constant(['em
=tf.constant([0
.lookup.StaticV

_buckets=5)
```

Google Developer Groups

Day 5: React JS

Build Fast, Scale Smart, Code Seamlessly!



<TECH WINTER BREAK/>

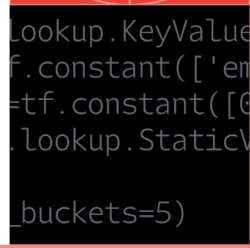
Level up your skills during winter chillllsss

Google Developer Groups

Let's Start!

React JS: An open-source JavaScript library for building user interfaces.

Applications are built using reusable and self-contained components, allowing for modular and maintainable code.



1. Introduction to React Js

What is ReactJS?

React.js is a popular open-source JavaScript library for building user interfaces, particularly for single-page applications.

Key Features

- Component based architecture
- Declarative Syntax
- Virtual DOM for efficient updates
- Unidirectional data flow

2. React JS Basics

Setting Up the Environment
- Install Node.js and npm
- Create a new React project using Create React App:

```
npx create-react-app my-app
cd my-app
npm start
```

Virtual DOM

- Definition: A lightweight copy of the actual DOM used by React to optimize updates.

- Benefits: minimizes DOM manipulations, speeds up rendering process.

ES6+ Features in React

- Arrow Functions: `const add = (a, b) => a + b;`

- Template Literals: `const greeting = \`Hello, ${name}\`;`

- Destructuring: `const { name, age } = person;`

JSX(Javascript XML)

- Syntax Extension for JavaScript.
- Advantages: clean syntax for combining HTML and JavaScript.
- Prevents XSS attacks by escaping values.

Example:

```
const element = <h1>Hello, world!</h1>;
ReactDOM.render(element, document.getElementById('root'));
```

3. Props and States

Props:
Read-only data passed from parent to child components.
Example:

```
function Greeting({ name }) {
  return <h1>Hello, {name}!</h1>;
}
```

State:
Local data storage managed within a component.
Example:

```
const [count, setCount] = useState(0);
```

4. Event Handling

Event Handling

- Example:

```
function handleClick() {
  console.log('Button clicked!');
}

<button onClick={handleClick}>Click Me</button>
```

Conditional Rendering

- Example:

```
function Greeting({ isLoggedIn }) {
  return isLoggedIn ? <h1>Welcome back!</h1> : <h1>Please sign in.</h1>;
}
```

5 . Advanced Concepts

React Hooks:
- Introduced in React 16.8
- useState: State management in function components
- useEffect: Handling side effects like data fetching
- Additional Hooks: useContext, useReducer, useRef.

6 . Events and Forms

Events and Forms

Handling Events
- Example:

```
function handleClick() {
  console.log('Button clicked!');
}
```

Controlled vs Uncontrolled Components
- Controlled: Form data is handled by React.
- Uncontrolled: Form data is handled by DOM.
- Example:

```
const [value, setValue] = useState('');
function handleChange(event) {
  setValue(event.target.value);
}
```

7. Routing with react

React Router
- Installing: npm install react-router-dom
- Example:

```
import { BrowserRouter, Routes, Route } from 'react-router-dom';

function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
      </Routes>
    </BrowserRouter>
  );
}
```

8. API Integration

Fetching Data
- Using fetch:

```
useEffect(() => {
  fetch('https://api.example.com/data')
    .then(response => response.json())
    .then(data => setData(data));
}, []);
```

Handling Errors:
- Example:

```
try {
  const response = await fetch(url);
} catch (error) {
  console.error('Error:', error);
}
```

9. Deploying React Application

Building for Production
- npm run build

Deploying
- Platforms: Netlify, Vercel, GitHub Pages.

Google Developer Groups

Thank You!

That’s all for today!

Happy New Year everyone,

The fun is in the journey, success is just a place to be at.