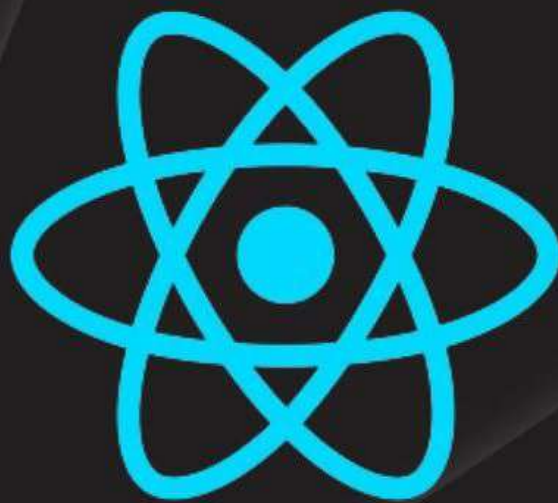


# Axios



@developer.mohit

# What is Axios?

- **Axios is a popular JavaScript library used to make HTTP requests from both the browser and Node.js**
- **It provides a simple and intuitive API for performing **CRUD** operations (Create, Read, Update, Delete) on web resources.**
- **npm install axios**



@developer.mohit

# GET and DELETE request

```
1  import axios from 'axios'
2
3  function App() {
4
5      const handleClick = () => {
6          // get using GET method
7          // delete using DELETE method
8          axios
9              .get('http://your-end-point')
10             // .delete('http://your-end-point')
11             .then((response) => {
12                 console.log(response.data)
13                 // data is retrieved from .data
14             })
15             .catch((err) => {
16                 console.log(err.message)
17                 // instead of whole red paragraph it will print the message only
18             })
19         }
20
21         return (
22             <>
23             <button onClick={handleClick}>Click me to Fetch</button>
24             </>
25         )
26     }
27
28     export default App
```



@developer.mohit

# POST and PUT request

```
import axios from 'axios'

function App() {
  const [data, setData] = useState({ key: 'abc' })

  const handleClick = () => {
    // send using POST method
    // put using PUT method
    axios
      .post('http://your-end-point', data)
      // .put('http://your-end-point', data)
      .then((response) => {
        console.log(response.data)
        // data is retrieved from .data
      })
      .catch((err) => {
        console.log(err.message)
        // instead of whole red paragraph it will print the message only
      })
  }

  return (
    <>
    <button onClick={handleClick}>Click me to Fetch</button>
    </>
  )
}

export default App
```



@developer.mohit

# Async/await

- I used Promises to **fetch** API, you can do with **async/await**



@developer.mohit

# Response object properties

```
function App() {
  const handleClick = () => {
    axios
      .get('https://api.example.com/data')
      .then((response) => {
        console.log(response.data) // Response data
        console.log(response.status) // HTTP status code
        console.log(response.statusText) // HTTP status message
        console.log(response.headers) // Response headers
        console.log(response.config) // Request config
      })
      .catch((error) => {
        console.error(error)
      })
  }

  return (
    <>
    <button onClick={handleClick}>Click me</button>
    </>
  )
}
```



@developer.mohit

# Error handling properties

```
function App() {  
  const handleClick = () => {  
    axios  
      .get('https://api.example.com/data')  
      .then((response) => {  
        console.log(response.data)  
      })  
      .catch((error) => {  
        if (error.response) {  
          // The request was made, and the server responded with a status code that falls out of the range of 2xx  
          console.log(error.response.data)  
          console.log(error.response.status)  
          console.log(error.response.headers)  
        } else if (error.request) {  
          // The request was made, but no response was received  
          console.log(error.request)  
        } else {  
          // Something happened in setting up the request that triggered an Error  
          console.log('Error', error.message)  
        }  
        console.log(error.config)  
      })  
  }  
}
```

It is more than enough,



@developer.mohit