

## 1. React-api-demo

### App.js

```
import React, { useEffect, useState } from "react";
import axios from "axios";

function App() {
  const [posts, setPosts] = useState([]);
  const [title, setTitle] = useState("");

  useEffect(() => {
    axios.get("https://jsonplaceholder.typicode.com/posts?_limit=5")
      .then(res => setPosts(res.data))
      .catch(err => console.log(err));
  }, []);

  const addPost = () => {
    axios.post("https://jsonplaceholder.typicode.com/posts", {
      title,
      body: "Demo post"
    }).then(res => alert("Post Added: " + res.data.title));
  };

  return (
    <div>
      <h1>API Calling using Axios</h1>
      <input
        value={title}
        onChange={(e) => setTitle(e.target.value)}
        placeholder="Enter title"
      />
    </div>
  );
}
```

```

/>
<button onClick={addPost}>Add Post</button>

<h2>Posts:</h2>
{posts.map(p => (
  <p key={p.id}>{p.title}</p>
))}
</div>

);
}
export default App;

```

## OUTPUT:

### API Calling using Axios

#### [Input] [Add Post Button]

#### Posts:

- sunt aut facere repellat
- qui est esse
- ea molestias quasi
- eum et est occaecati
- nesciunt quas odio

## 2. Hook-demo

```

import React, { useState, useEffect, useRef } from "react";

function App() {
  const [count, setCount] = useState(0);

```

```

const inputRef = useRef();

useEffect(() => {
  console.log("Count updated:", count);
}, [count]);

const focusInput = () => {
  inputRef.current.focus();
};

return (
  <div>
    <h1>React Hooks Demo</h1>

    <h2>Count: {count}</h2>
    <button onClick={() => setCount(count + 1)}>Increase</button>

    <br /><br />

    <input ref={inputRef} placeholder="UseRef Example" />
    <button onClick={focusInput}>Click to Focus</button>
  </div>
);
}

export default App;

```

## OUTPUT:

Count: 0  
 [Increase Button]  
 [Input Field] [Focus Button]

Click Increase → Count updates  
 Click Focus → Input automatically gets focus

## 3. Dynamic-style-demo

```

import React, { useState } from "react";
import "bootstrap/dist/css/bootstrap.min.css";

function App() {

```

```

const [isDark, setIsDark] = useState(false);

const changeTheme = () => setIsDark(!isDark);

return (
  <div className={isDark ? "bg-dark text-white p-5" : "bg-light p-5"}>
    <h1>Dynamic Class Rendering</h1>
    <h3>Expression Example: 5 + 5 = {5 + 5}</h3>

    <button className="btn btn-primary" onClick={changeTheme}>
      Toggle Theme
    </button>
  </div>
);
}

export default App;

```

### **OUTPUT:**

```

import React, { useState } from "react";
import "bootstrap/dist/css/bootstrap.min.css";

function App() {
  const [isDark, setIsDark] = useState(false);

  const changeTheme = () => setIsDark(!isDark);

  return (
    <div className={isDark ? "bg-dark text-white p-5" : "bg-light p-5"}>
      <h1>Dynamic Class Rendering</h1>
      <h3>Expression Example: 5 + 5 = {5 + 5}</h3>

      <button className="btn btn-primary" onClick={changeTheme}>
        Toggle Theme
      </button>
    </div>
  );
}

export default App;

```