

10. officespacereentalapp

Objectives

- Define JSX
- Explain about ECMA Script
- Explain React.createElement()
- Explain how to create React nodes with JSX
- Define how to render JSX to DOM
- Explain how to use JavaScript expressions in JSX
- Explain how to use inline CSS in JSX

In this hands-on lab, you will learn how to:

- Use JSX syntax in React applications
- Use inline CSS in JSX

Prerequisites

The following is required to complete this hands-on lab:

- Node.js
- NPM
- Visual Studio Code

Notes

Estimated time to complete this lab: 60 minutes.

Create a React Application named “officespacerentalapp” which uses React JSX to create elements, attributes and renders DOM to display the page.

Create an element to display the heading of the page.

Attribute to display the image of the office space

Create an object of office to display the details like Name, Rent and Address.

Create a list of Object and loop through the office space item to display more data.

To apply Css, Display the color of the Rent in Red if it's below 60000 and in Green if it's above 60000.

CODE:

App.js

```
import React from 'react';
```

```
function App() {
```

```
  const office = {
```

```
    name: "IndiQube Alpha",
```

```
    rent: 45000,
```

```
    address: "Koramangala, Bangalore",
```

```
    image: "https://picsum.photos/id/1011/300/150",
```

```
  };
```

```
  const rentStyle = {
```

```
    color: office.rent > 60000 ? "green" : "red",
```

```
    fontWeight: "bold",
```

```
};
```

```
return (
```

```
<div style={{ padding: "20px", fontFamily: "Arial" }}>
```

```
<h1 style={{ textAlign: 'center' }}>Office Space Rental</h1>
```

```
<div style={{
```

```
  border: "1px solid #ccc",
```

```
  padding: "10px",
```

```
  borderRadius: "8px",
```

```
  width: "300px",
```

```
  margin: "0 auto"
```

```
}}>
```

```
<img
```

```
  src={office.image}
```

```
  alt={office.name}
```

```
  style={{ width: "100%" }}
```

```
  onError={(e) => {
```

```
    e.target.onerror = null;
```

```
    e.target.src = "https://via.placeholder.com/300x150?text=No+Image";
```

```
  }}
```

```
/>
```

```
<h2>{office.name}</h2>
```

```
<p><strong>Address:</strong> {office.address}</p>
```

```
<p style={rentStyle}>
```

```

    <strong>Rent:</strong> ₹{office.rent}

  </p>

</div>

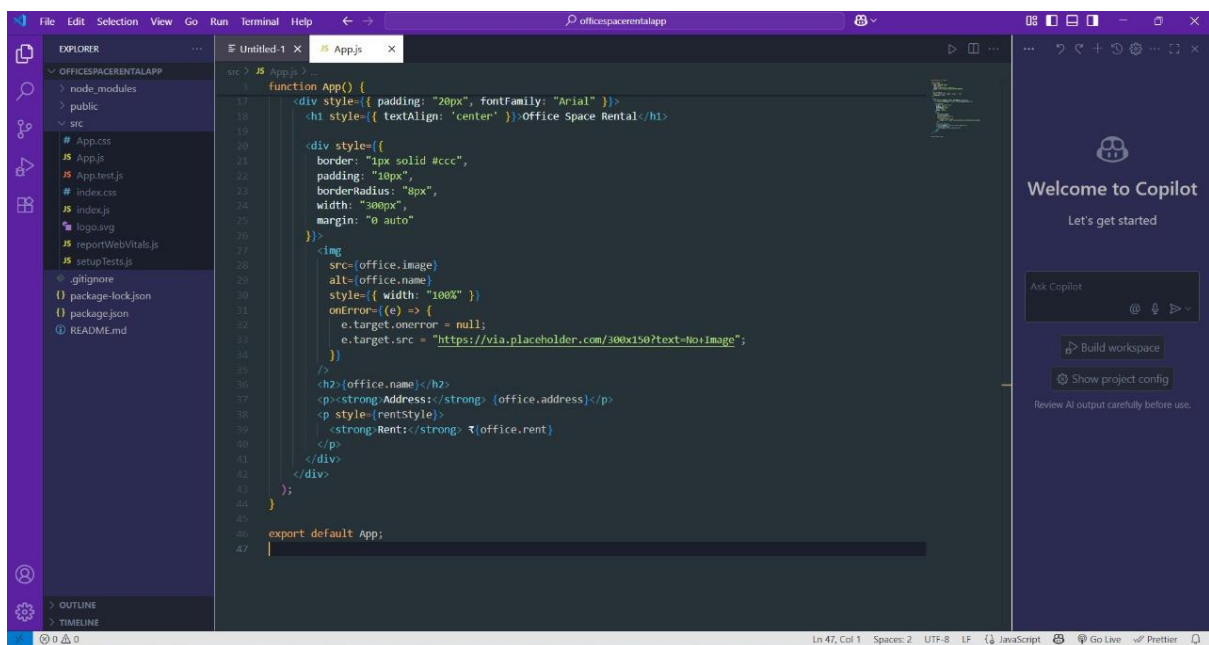
</div>

);

}

```

export default App;



```
Windows PowerShell

npm run build
  Bundles the app into static files for production.

npm test
  Starts the test runner.

npm run eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

cd officespacerentalapp
npm start

Happy hacking!

C:\Users\mkanm>cd officespacerentalapp
C:\Users\mkanm\officespacerentalapp>npm start
> officespacerentalapp@0.1.0 start
> react-scripts start

(node:24072) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationWarning: 'onAfterSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
(Use 'node --trace-deprecation ...' to show where the warning was created)
(node:24072) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] DeprecationWarning: 'onBeforeSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
Starting the development server...
Compiled successfully!

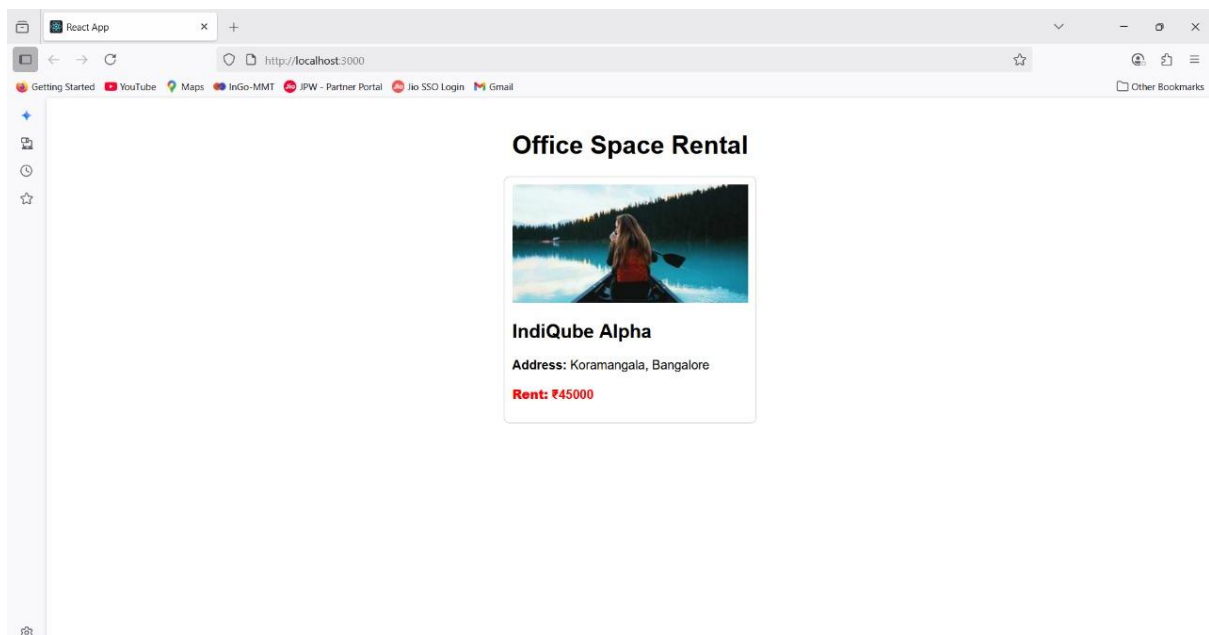
You can now view officespacerentalapp in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.135:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
Compiling...
Compiled successfully!
```

Output:



KANMANI MURUGHAIYAN

SUPERSET ID: 6407636

