Styling React Components

1. Understanding the need for styling React components

In a React application, components are the building blocks that define the structure and functionality of the user interface. However, without proper styling, even a well-functioning component may look unappealing or confusing to users.  
  
Styling is essential because it:  
- Improves visual appeal  
- Creates a consistent user experience  
- Enhances usability  
- Makes components responsive  
- Supports branding  
  
Therefore, styling React components is as crucial as building their logic, especially in production-level applications.

2. Working with CSS Modules and Inline Styles

React provides multiple ways to apply styles to components. Two commonly used approaches are CSS Modules and inline styles.  
  
CSS Modules:  
- Scoped locally to a specific component  
- Styles are imported as objects in the component  
- Prevents global style pollution  
- Encourages component-based styling  
  
Inline Styles:  
- Defined using JavaScript objects in JSX  
- Accepts camelCased CSS properties (e.g., fontSize, backgroundColor)  
- Suitable for dynamic styles based on props or state

Comparison:  
| Feature | CSS Modules | Inline Styles |  
|----------------------|---------------------------- |---------------------------|  
| Scope | Component-level | Element-specific |  
| Syntax | Traditional CSS | JavaScript object |  
| Dynamic styling | Needs external logic | Built-in support |  
| Maintenance | Easy for large apps | Harder to scale |  
| Use case | Layout and theming | Conditional styles |

CohortDetails.module.css

.box {  
 width: 300px;  
 display: inline-block;  
 margin: 10px;  
 padding: 10px 20px;  
 border: 1px solid black;  
 border-radius: 10px;  
 vertical-align: top;  
}  
  
dt {  
 font-weight: 500;  
}  
  
.title {  
 font-size: 1.1rem;  
 margin-bottom: 10px;  
}

CohortDetails.js

import React from 'react';  
import styles from './CohortDetails.module.css';  
  
function CohortDetails({ cohort }) {  
 const isOngoing = cohort.status.toLowerCase() === 'ongoing';  
 const headingStyle = {  
 color: isOngoing ? 'green' : 'blue',  
 fontWeight: 'bold',  
 };  
  
 return (  
 <div className={styles.box}>  
 <h3 className={styles.title} style={headingStyle}>{cohort.name}</h3>  
 <dl>  
 <dt>Started On</dt>  
 <dd>{cohort.startDate}</dd>  
 <dt>Current Status</dt>  
 <dd>{cohort.status}</dd>  
 <dt>Coach</dt>  
 <dd>{cohort.coach}</dd>  
 <dt>Trainer</dt>  
 <dd>{cohort.trainer}</dd>  
 </dl>  
 </div>  
 );  
}  
  
export default CohortDetails;

App.js

import React from 'react';  
import CohortDetails from './components/CohortDetails';  
  
const cohorts = [  
 {  
 name: 'INTADMDF10 -.NET FSD',  
 startDate: '22-Feb-2022',  
 status: 'Scheduled',  
 coach: 'Aathma',  
 trainer: 'Jojo Jose',  
 },  
 {  
 name: 'ADM21JF014 - Java FSD',  
 startDate: '10-Sep-2021',  
 status: 'Ongoing',  
 coach: 'Apoorv',  
 trainer: 'Elisa Smith',  
 },  
 {  
 name: 'CDBJF21025 - Java FSD',  
 startDate: '24-Dec-2021',  
 status: 'Ongoing',  
 coach: 'Aathma',  
 trainer: 'John Doe',  
 },  
];  
  
function App() {  
 return (  
 <div>  
 <h1>Cohorts Details</h1>  
 {cohorts.map((cohort, index) => (  
 <CohortDetails key={index} cohort={cohort} />  
 ))}  
 </div>  
 );  
}  
  
export default App;

**Output:**



