

Assignment: Fetching Data from an API and Displaying It in Cards Using React

Objective:

The aim of this assignment is to enhance your skills in fetching data from a public API and displaying that data in a well-structured card layout using React. You will learn how to manage state, handle side effects with `useEffect`, and present data dynamically in a card format.

Task:

1. Create a React Component:

- Create a new React component called `ProductCardList`.

2. Set Up State:

- Use the `useState` hook to create two pieces of state:
 - `products` (an array to hold the fetched product data).
 - `loading` (a boolean to manage the loading state).

3. Fetch Data with `useEffect`:

- Use the `useEffect` hook to fetch data from the following API endpoint when the component mounts:
 - API: `https://fakestoreapi.com/products`
 - While the data is being fetched, set the `loading` state to `true`. Once the data is successfully fetched, update the `products` state with the fetched data and set the `loading` state to `false`.

4. Create a Card Component:

- Create a child component called `ProductCard` that accepts a product as a prop and displays the following details in a card layout:
 - Product image
 - Product title
 - Product price
 - Product description (limited to 100 characters)

- Style the card with basic CSS to ensure it looks visually appealing.

5. Render the Cards:

- In the `ProductCardList` component, map over the `products` array and render a `ProductCard` for each product.
- If `loading` is `true`, display a loading message.

6. Error Handling (Bonus):

- Optionally, handle any errors that might occur during the fetch operation by displaying an error message.

Example Output:

The component should display a 'Loading...' message while the data is being fetched. Once the data is fetched, it should display a grid of product cards, each showing the product image, title, price, and truncated description.

Requirements:

- You must use `useState` to manage the `products` and `loading` states.
- You must use `useEffect` to fetch data from the API when the component mounts.
- The `ProductCard` component should be reusable and accept product details as props.

Deliverables:

A React component that fetches product data from the API and displays it in a series of cards. The card layout should be visually structured and responsive.