React Fundamental

EASY
Level

Task 1 : Create a Basic Functional Component
Simple JavaScript function returning JSX.

Output : “ Your Name “

Task 2 : Create a Basic Class Component
React component using class component with render() method.

Output : “ Your Position “

Task 3 : Display Name in Functional Component
Show a static name in a simple component.

Hint : const name = "Amro";
Output : Welcome, Amro

Task 4 : Adding Inline Styles to a Functional Component
Apply inline styles to a component for custom design.

Hint :Use the style attribute with an object to style elements. “ const style = { };“

Output : Orange **Coding** *Academy*

Task 5 : Add Multiple Inline Styles in a Functional Component
Apply different styles to elements within a component.

Hint : Create separate style objects for each element and apply them using style attribute
Output :

Task 6 : Applying External JSON Styles in a Component
Use JSON file for styling in a React component.

Hint : Import the JSON file and apply styles through the style attribute { "con1": { }, "con2": { }, }
Output :

orange™
دايماً معاك

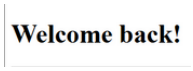
React Fundamental

INTERMIDATE
Level

Task 7 : Conditional Rendering in a Functional Component

Show different content based on a condition " Display a welcome or login prompt based on login status. "

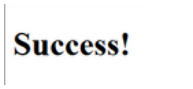
Hint : ***isLoggedIn={true}***
Output :



Task 8 : Conditional Rendering in a Class Component

Display different messages based on a condition " Show "Success!" or "Error!" based on isSuccess prop. "

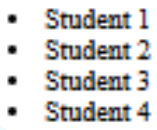
Hint : ***isSuccess={true}***
Output :



Task 9 : Rendering List Items Dynamically in React

Render a list of items passed via props in React.

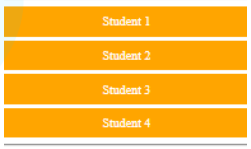
Hint : Use .map() to render list items , <Task9 items={students} />
Output :



Task 10 : Creating a Styled List Component with Inline Styles in React

Render and style a list of items with inline styles **using MAP**

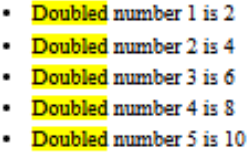
Hint : {items.map((item, index) => ( ))} Use .map() to render list items.
Output :



Task 11 : Displaying Doubled Numbers from an Array in React

Render an array of numbers, displaying each number doubled.

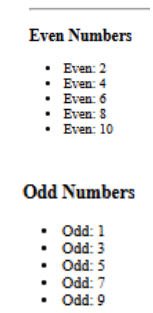
Hint : Use .map() to transform and display array values , const numbers = [1, 2, 3, 4, 5];
Output : **Doubled 1 => 2 , Doubled 2=> 8 .....**



Task 12 : Displaying Even and Odd Numbers

Display static cards with title, description, and image.

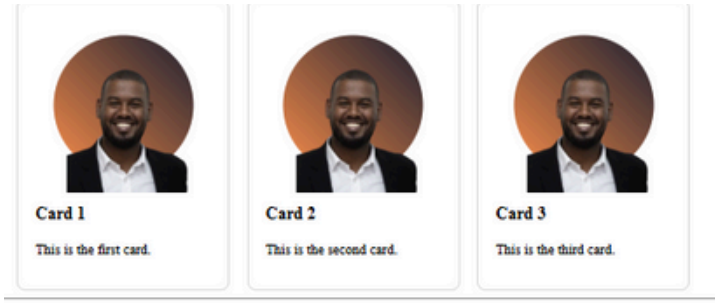
Hint : We use .map() to iterate over the numbers array , const numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
Output : Even Number **as list (2 , 4 , 6 , 8 , 10)** , Odd Number **as list (1 , 3 , 5 , 7 , 9)**



Task 13 : Rendering Static Cards with Title, Description, and Image in React

Dynamically render cards with title, description, and image using .map().

Hint : Use .map() to display each card dynamically
Output :



Task 14 : Implementing Custom Toast Notifications with react-toastify

Use the react-toastify library to display customizable toast notifications in a React app with different styles for the parent and child components.

You'll create a parent and child component, each triggering different toasts with custom styles. Customize the appearance using inline styles or CSS classes.

Customize position, duration, and appearance by passing options like position, autoClose, and style to toast()

Hint : toast("message") , Add <ToastContainer /> , dont forget import pacakge and ToastContainer

Raect - toastify