**Frontend Documentation**

**Personal Fitness Trainer**

This site is made using three components:

1. ReactJS being a popular JavaScript library for building user interfaces that allows for efficient rendering and updating of components. It provides a structured way to create reusable components and manage the state of a website or application.
2. I used HTML because it is the standard markup language used to create web pages, providing the structure and content of a website. It is supported by all web browsers and is essential for building any kind of website.
3. I used CSS to style web pages, allowing for the customization of visual elements such as layout, colors, and fonts. It provided a separation of presentation and content, making it easier to maintain and update a website's appearance.

**JSX File Summary**

**App.jsx** :

This code defines a React function component called App, which renders a website using HTML and CSS. The website includes multiple routes using the React Router library, each of which renders a different component when navigated to. The selectedValue state and setSelectedValue function are passed down as props to the Homepage component, which can be used to update the state of the App component.

**Index.js** :

This code imports React and ReactDOM libraries and renders a React component called App in the ‘root’ element of the HTML document. It also uses React’s StrictMode to highlight potential issues and improvements during development.

**Homepage.jsx** :

This code defines a React function component called Homepage, which renders a form for users to enter their personal details and fitness goals. The form includes input fields for name, age, height, and weight, as well as a dropdown component for selecting a fitness goal. When the user clicks the "submit" button, the handleSubmit function is called to check if all required fields are filled, and if so, it navigates the user to the fitness route using the React Router library. The selectedValue state and setSelectedValue function are passed down as props from the App component, and are used to update the state of the Homepage component.

**Fitness.jsx** :

This code defines a React component called "Fitness". It imports the "useNavigate" and "useState" hooks from React Router and the "Dropdown" component from a local file. Inside the component, there are three "handleSubmit" functions that navigate to different pages when buttons are clicked. The component renders a form with a title and a "Dropdown" component to select a fitness plan. Then, there are two buttons to navigate to different pages for choosing a diet plan or workout routine. Finally, there is a "Submit" button that does not have a specific action defined.

**Mui.jsx** :

The code provides a dropdown component using React and CSS to select fitness plans for diet and workout routines. It uses the useState and useEffect hooks for state management and event handling, respectively. The dropdown options are passed as props, and when an option is selected, the setSelectedValue1 function is called to update the selected value. The selected value is displayed in the dropdown input field, and clicking the input field toggles the visibility of the dropdown menu. The handleSubmit functions navigate to different pages based on the selected fitness plan.

**Ex.jsx** :

This is a functional component in React that creates a button that toggles the display of a list when clicked. It uses the useState hook to keep track of whether the list is currently displayed or hidden. When the button is clicked, the toggleList function is called which updates the showList state variable to the opposite of its current value. If showList is true, the list is displayed, otherwise it is hidden.

Info.jsx : This is a React component named "Info" which imports several other components: ButtonWithList, ButtonWithListSupp, ButtonWithListMeal, and ButtonWithListEx. It also imports some CSS files.

In the component's code, there are several functions that handle button clicks and use the useNavigate hook from the react-router-dom package to navigate to different pages. The handleSubmit function navigates to "/fitness", handleSubmit1 navigates to "/progress", handleSubmit2 navigates to "/workout", handleSubmit3 navigates to "/muscle", and handleSubmit4 navigates to "/faq".

The return statement of the component contains a form with several buttons and text that display information and lists when clicked. The buttons trigger the corresponding handle functions to navigate to other pages or show information on the same page.

**Dropdown.jsx :**

The Dropdown component is a custom dropdown menu that allows the user to select from a list of options. It takes in several props, including a placeholder for the initial display, a selected value, and a function to set the selected value. The component uses state to manage whether or not the dropdown menu is currently displayed, and it uses the useEffect hook to add an event listener to hide the menu when the user clicks outside of it. The options for the dropdown are defined as an array of objects with values and labels, and the selected value is displayed in the dropdown input. When the user clicks on the input, the dropdown menu is displayed with the available options. When an option is clicked, it is set as the new selected value and the menu is hidden.

**Diet.jsx :**

This code defines a functional component called Diet. It imports the useNavigate hook from the react-router-dom library, a Dropdown component from another file, and the useState hook from the react library.

The Diet component returns a form with a dropdown menu and a submit button. The dropdown menu is created using the Dropdown component, and the selected value from the dropdown is stored in the component's state using the useState hook.

When the submit button is clicked, the handleSubmit function is called, which uses the useNavigate hook to navigate to the /fitness route.

**Muscle.jsx**

This is a functional component named "Muscle" that allows users to input a muscle group and outputs a list of exercises corresponding to that muscle group. The component uses the useState hook to manage the input and output values. It has a form with an input field that allows users to enter a muscle group, and a button to submit the form.

The component has a handleChange function that sets the input value whenever the input field changes. It also has a handleSubmit function that is called when the form is submitted. The handleSubmit function checks the input value and sets the output value accordingly using if/else statements.

The output value is then displayed on the screen along with the input field and submit button.

**Table.js**

This is a React component called "Table" that creates a form with three input fields: "Weight", "Target Weight", and "Body Fat Percent", and a button to submit the form. When the form is submitted, a new object is created with the values from the input fields, and it is added to an array of entries using the "setEntries" function provided by the "useState" hook.

Below the form, there is a table that displays all the entries added so far. It has a header row and a body where each row corresponds to an entry, with the values displayed in separate columns. The table is generated using the "map" function, which iterates over the entries array and generates a row for each entry. The "key" prop is set to the index of the entry in the array to help React keep track of the elements.

**Routine.jsx**

This is a React functional component that renders a form with a dropdown menu component. The component imports useNavigate and useState hooks from React, and the Dropdown component from a local file named Roudiet.

The useNavigate hook is used to navigate to a specific route in the application when the form is submitted, and the useState hook is used to manage the state of the selected value in the dropdown menu.

The handleSubmit function is called when the submit button is clicked, and it uses the navigate function from the useNavigate hook to redirect the user to the "/fitness" route.

The component returns a div that contains the form with a dropdown menu rendered by the Dropdown component, and a submit button. The Dropdown component is passed the selectedValue1 and setSelectedValue1 props, which are used to manage the state of the selected value in the dropdown menu.

**Workout.js**

The code appears to be a functional component in React that renders a form for entering data into a table, as well as the table itself.

The useState hook is used to define a state variable called entries, which is initialized to an empty array. When the form is submitted, the handleSubmit function is called, which creates a new object representing the entry and adds it to the entries array using the setEntries function. The event.target.reset() statement resets the form to its initial state.

The table is defined using the table, thead, tbody, tr, and th elements in HTML, and it is rendered using JSX. The entries array is mapped over to create a table row for each entry, which is then displayed in the table.

The input fields are defined using the input element in HTML, and they allow the user to enter the date, routine, and time for each workout. When the form is submitted, the data is used to create a new entry in the entries array, which is then displayed in the table.

Overall, this component seems to be a basic implementation of a form that allows users to enter data and display it in a table format.

**WEBSITE NAVIGATION**

The web application can be initiated by executing the command 'npm run start', which will start a development server and launch the React application. Upon successful initialization, the application will direct the User to the home webpage, where they can enter their personal data and select a desired Fitness Goal from the available options to proceed.

After providing the necessary input and submitting the data, the application will redirect the User to a fitness page, where they can provide additional information regarding their specific fitness plan, diet plan, and workout routine. Upon submitting this information, the application will generate personalized guidance for the User based on their individual fitness goal, including information about recommended supplements, meal plans, and exercise routines.

The application also includes a progress log feature that allows Users to track their progress over time by storing information about their weight, target weight, and body fat percentage. Additionally, the workout log feature enables Users to record the details of their workout routines and monitor their regularity.

For those seeking targeted exercises for specific muscle groups, the application includes a section dedicated to this purpose. Finally, a comprehensive FAQ section is provided at the end to address general queries and provide clarity regarding the workings of the web application.

In summary, the application provides Users with a user-friendly interface to input personal data, establish their fitness goals, and receive personalized guidance to achieve their desired outcomes. The progress log and workout log features enable Users to monitor their progress and stay on track, while the targeted exercises and FAQ sections offer further support and guidance as needed