Fetch-axios:

App.css:

@import url('https://fonts.googleapis.com/css2?family=Playfair+Display:ital,wght@700;500&display=swap');

.App {

  font-family: 'Playfair Display', serif;

  margin: 20px 15vw;

  }

  h2 {

  font-family: 'Playfair Display', serif;

  font-weight: 700;

  font-size: 3em;

  }

  button {

  padding: 1em 1.2em;

  border: none;

  font-size: 1em;

  background-color: #131212;

  color: #ffffff;

  border-radius:0.5em;

  cursor: pointer;

  }

  button:hover{

  background-color:#3b3b3b;

  }

.watermark{

  position: fixed;

  bottom: 10px;

  right: 10px;

  color: rgba(8, 8, 8, 0.4);

  font-size: 15px;

}

App.js:

import "./App.css";

import { useEffect, useState } from "react";

import axios from "axios";

function App() {

  const [fact, setFact] = useState("");

  const fetchClick = () => {

    fetchFact()

  }

  const axiosClick = () => {

    fetchFact1()

    }

  const fetchFact = () => {

  fetch("https://catfact.ninja/fact")

   .then((response) => response.json())

  .then((data) => setFact(data.fact));

}

const fetchFact1 = () => {

  axios.get("https://catfact.ninja/fact").then((response) => {

  setFact(response.data.fact)

});

  }

useEffect(() => {

  fetchFact()

}, []);

return (

  <div className="App">

    <h2>Press the button for a random cat fact!</h2>

      <hr />

      <button onClick= {() => fetchClick()}>Get Cat fact using fetch</button>&nbsp;&nbsp;&nbsp;

    <button onClick= {() => axiosClick()}>Get Cat fact using axios</button>

    <p>{fact}</p>

    <div className="watermark">

        <p style={{ textAlign: 'right', marginRight: '20px', fontWeight: 'bold'}}>

        Dasari Harika (22BQ1A4235) </p>

        </div>

  </div>

);

}

export default App;

BMI:

App.css:

.App {

  text-align: center;

  font-family: Arial, sans-serif;

}

h1 {

  margin-top: 20px;

  font-size: 24px;

  color: #333;

}

form {

  background-color:none;

  padding: 20px;

  border-radius: 10px;

  display: inline-block;

  text-align: left;

}

label {

  font-size: 16px;

  color: #333;

  font-weight: bold;

}

input[type="text"],

input[type="number"] {

  width: 250px;

  padding: 8px;

  margin: 10px 0;

  border: 1px solid #ccc;

  border-radius: 5px;

  font-size: 14px;

}

input[type="submit"] {

  background-color: #007bff;

  color: white;

  border: none;

  padding: 10px 15px;

  font-size: 16px;

  border-radius: 5px;

  cursor: pointer;

}

input[type="submit"]:hover {

  background-color: #0056b3;

}

.watermark{

  font-size: 20px;

  padding-top: 60px;

}

App.js:

import React, {useState} from 'react';

import './App.css';

function BMICalcDemo() {

const [height,setHeight] = useState(0); const [weight, setWeight] = useState(0); const [name, setName] = useState("");

const calculateBMI = ()=> {

var heightSquared = (height/100 \* height/100); var bmi = weight / heightSquared;

if( bmi < 16)

window.alert("Hi.."+name+"...You are completely UnderWeight (Severe Thinner)"); else

if( bmi >=16 && bmi < 17)

window.alert("Hi.."+name+"...You are Moderately UnderWeight (Moderate Thinner)"); else

if( bmi >=17 && bmi < 18.5)

window.alert("Hi.."+name+"...You are little UnderWeight (Mild Thinness)"); else

if( bmi >= 18.5 && bmi <= 24.99 ){ window.alert("Hi.."+name+"...You are in a healthy weight range");

}

else if(bmi >= 25 && bmi <= 29.9){ window.alert("Hi.."+name+"...You are overweight");

}

else if(bmi >= 30){ window.alert("Hi.."+name+"...You are obese");

}

else if(bmi < 18.5){

window.alert("Hi.."+name+"...You are under weight");

}

bmi = Math.round(bmi \* 100) / 100;

}

const submitMe = (e) =>{ e.preventDefault(); calculateBMI();

}

const handleName = (e) =>{ setName(e.target.value);

}

const handleHeight = (e) =>{ setHeight(e.target.value);

}

const handleWeight = (e) =>{ setWeight(e.target.value);

}

return (

<div className="App">

<h1>BMI Calculator</h1>

<form onSubmit={submitMe}>

<label>

Please enter your name:

</label>

<input type="text" name="name" value={name} onChange={handleName}/>

<br/><br/>

<label>

Enter your height in cm:

</label>

<input type="text" name="height" value={height} onChange={handleHeight}  />

<br/><br/>

<label>

Enter your weight in kg :

</label>

<br/>

<input type="text" name="weight" value={weight} onChange={handleWeight}/>

<br/>

<br/>

<input type="submit" value="Submit"/>

</form>

<div class="watermark">B.Jahnavi (23BQ5A4202)</div>

</div>

);

}

export default BMICalcDemo;

Counter:

App.js

import React, { useState } from "react";

import Button from "./button.jsx";

import "./style.css";

export default function App() {

  const [count, setCount] = useState(0);

  let incrementCount = () => {

    setCount(count + 1);

  };

  let decrementCount = () => {

    setCount(count - 1);

  };

  return (

    <div className="app">

      <div>

        <div className="title-container">

          <h2>COUNTER REACT APP</h2>

          <div className="title-underline"></div>

        </div>

        <div className="count">

          <h3>Count</h3>

          <h1>{count}</h1>

        </div>

        <div className="buttons">

          <Button align="center" title="-" action={decrementCount} />

          <Button align="center" title="+" action={incrementCount} />

        </div>

      </div>

      {/\* Watermark \*/}

      <div className="watermark">DASARI HARIKA</div>

    </div>

  );

}

Button.jsx:

import React from "react";

export default function Button(props){

    let { action, title } = props;

    return <button onClick={action}>{title}</button>

}

Style.css:

body {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    margin: 0;

    text-align: center;

}

.app {

    display: flex;

    flex-direction: column;

    justify-content: center;

    align-items: center;

    margin: 5rem;

}

.app .count {

    display: flex;

    align-items: center;

    justify-content: center;

    margin-bottom: 2rem;

}

.app .count h3 {

    line-height: 8.6rem;

    margin-right: 1rem;

}

.app .count h1 {

    font-size: 5em;

    margin-top: 2rem;

}

.app .buttons {

    display: flex;

    justify-content: center;

    gap: 2rem;

}

.app .buttons button {

    width: 10rem;

    height: 10rem;

    border-radius: 50%;

    border: none;

    font-size: 5em;

    box-shadow: 0 4px 32px 0 rgba(10, 14, 29, 0.2), 0 8px 64px 0 rgba(10, 14, 29, 0.08);

    cursor: pointer;

    transition: font-size 0.2s ease-in-out;

}

.app .buttons button:hover {

    font-size: 7em;

}

.app .buttons button:focus {

    outline: 0;

}

.title-container {

    margin-bottom: 20px;

}

.title-container h2 {

    text-decoration: underline;

    display: inline-block;

    text-align: center;

}

.watermark {

    position: fixed;

    bottom: 10px;

    right: 10px;

    font-size: 12px;

    color: rgba(0, 0, 0, 0.5);

    font-weight: bold;

    user-select: none;

    z-index: 1000;

}