Reactjs:

import React, { useState } from "react";

import "./App.css"

import axios from 'axios';

const api = {

  key: "7d4ef5d23648c7c6acd3ccfala0a30b",

  base: "https://api.openweathermap.org/data/2.5/"

}

function App() {

  const [query, setQuery] = useState('');

  const [weather, setWeather] = useState({});

  const search = *evt* => {

    if (*evt*.key === "Enter") {

      fetch(`${api.base}weather?q=${query}&units=metric&APPID=${api.key}`)

        .then(*res* => *res*.json())

        .then(*result* => {

          setWeather(*result*);

          setQuery('');

          console.log(weather)

        });

    }

  }

  const dateBuilder = (*d*) => {

    let months = ["1", "2", "3", "4", "5"];

    let days = ["Sunday", "badday", "happyDay"];

    let day = days[*d*.getDay()];

    let date = *d*.getDay();

    let month = months[*d*.getMonth()];

    let year = *d*.getFullYear();

    return `${day} ${month} ${year}`

  }

  return (

    <div *className*="app">

      <main>

        <div *className*="search-box">

          <input

*className*="search-bar"

*type*="text"

*placeholder*="Search..."

*onChange*={*e* => setQuery(*e*.target.value)}

*value*={query}

*onkeyPress*={search}

          />

        </div>

        {(typeof weather.main != "undefined") ? (

          <div>

            <div *className*="location-box">

              <div *className*="location">{weather.name},{weather.sys.country} </div>

              <div *className*="date">{dateBuilder(**new** *Date*())}</div>

            </div >

              <div *className*="weather-box">

                <div *className*="temp">

              {weather.main.temp} c

                </div>

                <div *className*="weather">{weather.weather[0].main}</div>

              </div>

            </div>

 ) : ('')}

          </main>

</div>

  );

}

export default App;

css:

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

body{

  font-family: 'montseratt', sans-serif;

}

*.app*{

  background-size: cover;

  transition: 0.4 case-out;

}

main {

  min-height: 100vh;

}

*.search-box*{

  width: 100%;

  margin: 0 0 75px;

}

*.search-box* *.search-bar*{

  display: block;

  width: 100%;

  padding: 15px;

  appearance: none;

  background: none;

  border: none;

  outline: none;

  background-color: rgba(225, 225,255,0.5);

  border-radius: 0 0 16px 16px;

  margin-top: -5px;

  box-shadow:  0px 5px rgba(0,0,0,0.2);

  color: #313131;

  font-size: 20px;

  transition: 0.4s ease;

max-width: 400px;

flex: 1;

margin: 0 auto;

}

*.search-box* *.search-bar:focus* {

  background-color: rgba(255, 255,255,0.75);

}

*.location-box* *.location* {

  font-size: 32px;

font-weight: 500;

  text-align: center;

  text-shadow: 3px 3px rgba(50,50,70,0.5) ;

}

*.location-box* *.date*{

  color: #FFF;

  font-size: 20px;

  font-weight: 300;

  text-align: center;

  text-shadow: 2px 2px rgba (50, 50,70, 0.5);

}

*.weather-box*{

  text-align: center;

}

*.weather-box* *.temp*{

  position: relative;

  display: inline-block;

  margin: 30px auto;

  background-color: rgba(255,255,255,0.2);

 border-radius: 16px;

  padding: 15px 25px;

  font-size: 102px;

  font-weight: 900;

  text-shadow: 3px 6px rgba(50,50,70,0.5);

  text-align: center;

  justify-content: center;

  align-items: center;

  box-shadow: 3px 6px rgba(0,0,0,0.2);

  display: flex;

  max-width: 350px;

}

*.weather-box* *.weather*{

  color: black;

  font-size: 48px;

  font-weight: 500px;

  text-shadow: 3px 3px rgba (50,50,70, 0.5);

}

import React, { useState } from 'react';

import axios from 'axios';

const cities = [

  { name: 'New York', code: 'NY' },

  { name: 'Los Angeles', code: 'CA' },

  { name: 'Chicago', code: 'IL' },

  { name: 'Houston', code: 'TX' },

  { name: 'Philadelphia', code: 'PA' },

  { name: 'Phoenix', code: 'AZ' },

  { name: 'San Antonio', code: 'TX' },

  { name: 'San Diego', code: 'CA' },

  { name: 'Dallas', code: 'TX' },

  { name: 'San Jose', code: 'CA' },

  { name: 'Austin', code: 'TX' },

  { name: 'Jacksonville', code: 'FL' },

  { name: 'Fort Worth', code: 'TX' },

  { name: 'Columbus', code: 'OH' },

  { name: 'San Francisco', code: 'CA' },

  { name: 'Charlotte', code: 'NC' },

]

const WeatherApp = () => {

  const [city, setCity] = useState('');

  const [weatherData, setWeatherData] = useState(null);

  const handleInputChange = (*event*) => {

    setCity(*event*.target.value);

  };

  const handleSubmit = async (*event*) => {

*event*.preventDefault();

    const apiKey = '30cca3cf59c040bd80e110005230603';

    const url = `https://api.weatherapi.com/v1/current.json?key=${apiKey}&q=${city}`;

    const response = await axios.get(url);

    setWeatherData(response.data);

  };

  return (

    <div>

      <h1>Weather App</h1>

      <form *onSubmit*={handleSubmit}>

        <label>

          Enter city:

          <input *type*="text" *value*={city} *onChange*={handleInputChange} />

        </label>

        <button *type*="submit">Get Weather</button>

      </form>

      {weatherData && (

        <div>

          <h2>Weather for {weatherData.location.name}</h2>

          <p>Temperature: {weatherData.current.temp\_c}°C</p>

          <p>Condition: {weatherData.current.condition.text}</p>

        </div>

      )}

    </div>

  );

};

export default WeatherApp;

author:

import React, { useState } from 'react';

function App() {

  const [quote, setQuote] = useState({});

  const [authorQuotes, setAuthorQuotes] = useState([]);

  const [authorName, setAuthorName] = useState('');

  const fetchRandomQuote = () => {

    fetch('https://api.quotable.io/random')

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

      .then(*data* => {

        setQuote(*data*);

      });

  };

  const fetchAuthorQuotes = () => {

    fetch(`https://api.quotable.io/quotes?author=${authorName}`)

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

      .then(*data* => {

        setAuthorQuotes(*data*.results);

      });

  };

  const handleRandomClick = () => {

    fetchRandomQuote();

    setAuthorQuotes([]);

    setAuthorName('');

  };

  const handleAuthorClick = () => {

    fetchAuthorQuotes();

  };

  const handleAuthorNameChange = (*event*) => {

    setAuthorName(*event*.target.value);

  };

  return (

    <div *className*="App">

      <h1>Random Quotes</h1>

      <div>

        <h2>{quote.content}</h2>

        <p>- {quote.author}</p>

        <button *onClick*={handleRandomClick}>Random Quote</button>

        {quote.author && <button *onClick*={handleAuthorClick}>View All Quotes by {quote.author}</button>}

      </div>

      <div>

        <h3>Quotes by {quote.author || authorName}</h3>

        {quote.author &&

          <ul>

            {authorQuotes.map(*quote* => (

              <li *key*={*quote*.\_id}>{*quote*.content}</li>

            ))}

          </ul>

        }

        {!quote.author &&

          <div>

            <input *type*="text" *value*={authorName} *onChange*={handleAuthorNameChange} *placeholder*="Enter author name" />

            <button *onClick*={handleAuthorClick}>Search</button>

            {authorQuotes.length > 0 &&

              <ul>

                {authorQuotes.map(*quote* => (

                  <li *key*={*quote*.\_id}>{*quote*.content}</li>

                ))}

              </ul>

            }

          </div>

        }

      </div>

    </div>

  );

}

export default App;