Web assignment – Week 15

Harini S

23BCE5142A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

import React, { useState } from 'react';

function Header({title}){

  return(

    <h1>{title}</h1>

  )

}

function Content(){

  const joke = ["aprilfools", "dad", "dadjoke", "dadjoke", "dadjoke"];

  const [randomnum , setrandomnum ]= useState("")

  const generateRandom = () => {

    const random = Math.floor(Math.random() \* 5);

    setrandomnum(joke[random]);

  }

  return(

    <div>

    <button onClick={generateRandom}>Generate a random joke</button>

    <h2>{randomnum}</h2>

    </div>

  )

}

function Footer(){

  return(

    <h3>Footer</h3>

  )

}

function App(){

  return (

    <div>

      <Header title="Random Joke Generator"/>

      <Content />

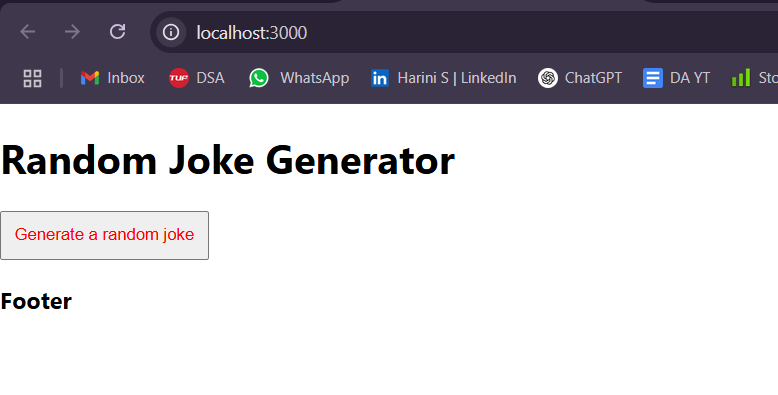
      <Footer />

    </div>

   )

}

export default App;



import React, { useState } from 'react';

function Header({title}){

  return(

    <h1>{title}</h1>

  )

}

function Content(){

  const joke = ["aprilfools", "dad", "dadjoke", "dadjoke", "dadjoke"];

  const [randomnum , setrandomnum ]= useState("")

  const generateRandom = () => {

    const random = Math.floor(Math.random() \* 5);

    setrandomnum(joke[random]);

  }

  return(

    <div>

    <button onClick={generateRandom} style={{fontsize:"10px",padding:"10px",color:"red"}}>Generate a random joke</button>

    <h2>{randomnum}</h2>

    </div>

  )

}

function Footer(){

  return(

    <h3>Footer</h3>

  )

}

function App(){

  return (

    <div>

      <Header title="Random Joke Generator"/>

      <Content />

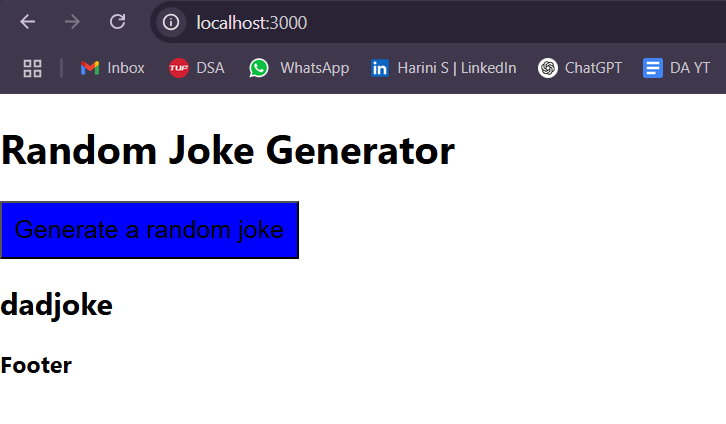
      <Footer />

    </div>

   )

}

export default App;



import React, { useState } from 'react';

function Header({title}){

  return(

    <h1>{title}</h1>

  )

}

function Content(){

  const styles = {

    backgroundColor:"blue",

    padding:"10px",

    fontSize:"20px"

}

  const joke = ["aprilfools", "dad", "dadjoke", "dadjoke", "dadjoke"];

  const [randomnum , setrandomnum ]= useState("")

  const generateRandom = () => {

    const random = Math.floor(Math.random() \* 5);

    setrandomnum(joke[random]);

  }

  return(

    <div>

    <button onClick={generateRandom} style={styles}>Generate a random joke</button>

    <h2>{randomnum}</h2>

    </div>

  )

}

function Footer(){

  return(

    <h3>Footer</h3>

  )

}

function App(){

  return (

    <div>

      <Header title="Random Joke Generator"/>

      <Content />

      <Footer />

    </div>

   )

}

export default App;

A screenshot of a computer

AI-generated content may be incorrect.

button{

  background-color: #4CAF50; /\* Green \*/

  border: none;

  color: white;

  padding: 15px 32px;

  text-align: center;

  text-decoration: none;

  display: inline-block;

  font-size: 16px;

  margin: 4px 2px;

  cursor: pointer;

}

import React, { useState } from 'react';

import './App.css';

function Header({title}){

  return(

    <h1>{title}</h1>

  )

}

function Content(){

  const joke = ["aprilfools", "dad", "dadjoke", "dadjoke", "dadjoke"];

  const [randomnum , setrandomnum ]= useState("")

  const generateRandom = () => {

    const random = Math.floor(Math.random() \* 5);

    setrandomnum(joke[random]);

  }

  return(

    <div>

    <button onClick={generateRandom}>Generate a random joke</button>

    <h2>{randomnum}</h2>

    </div>

  )

}

function Footer(){

  return(

    <h3>Footer</h3>

  )

}

function App(){

  return (

    <div>

      <Header title="Random Joke Generator"/>

      <Content />

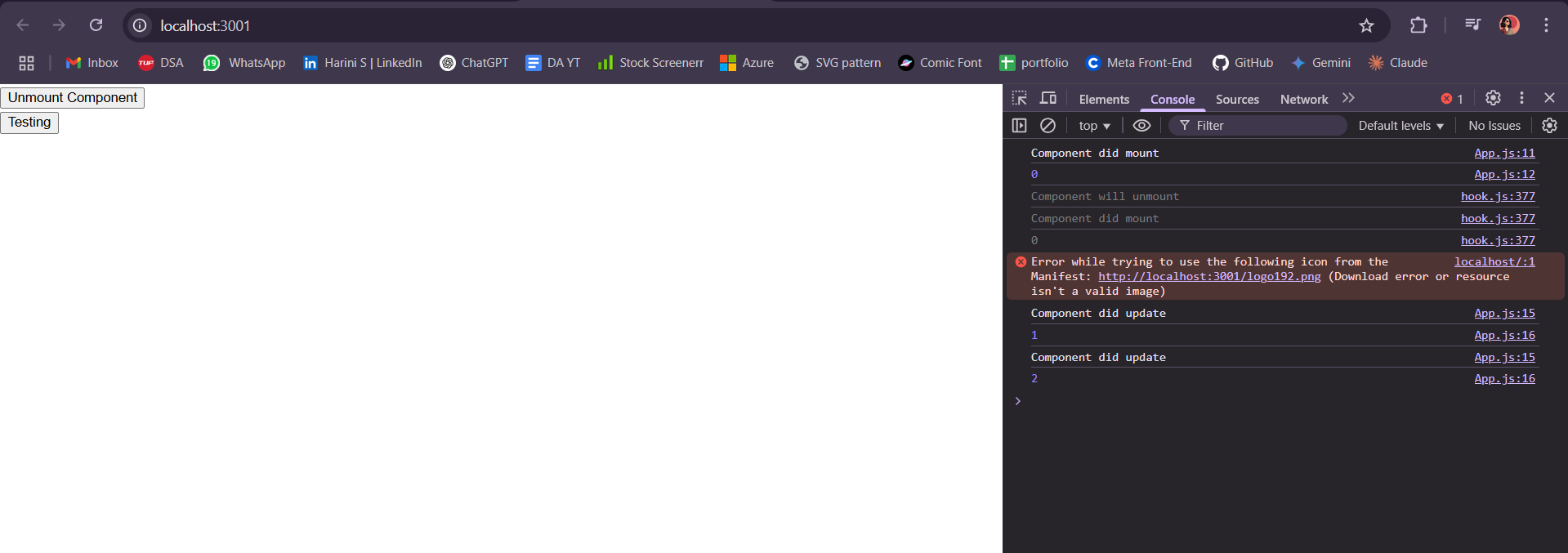
      <Footer />

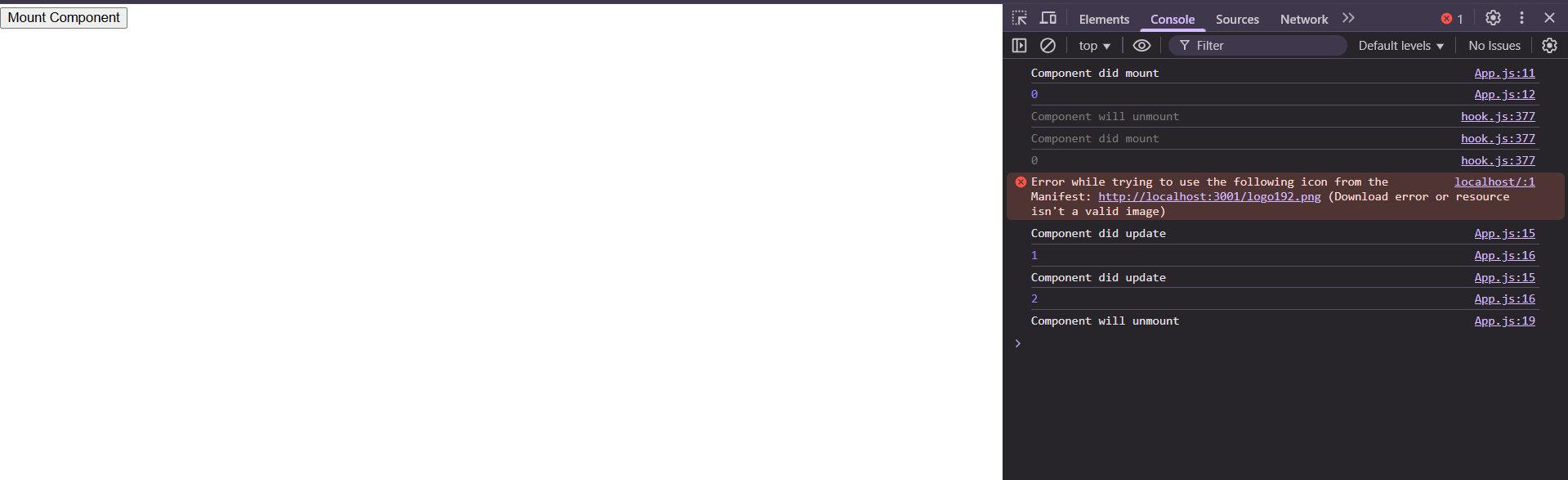
    </div>

   )

}

export default App;





import React,{useState} from 'react';

class LifeCycleClassDemo extends React.Component{

  constructor(props){

    super(props);

    this.state ={

      value : props.value || 0

    }

    this.func = this.func.bind(this);

  }

  componentDidMount(){

    console.log("Component did mount");

    console.log(this.state.value);

  }

  componentDidUpdate(){

    console.log("Component did update");

    console.log(this.state.value);

  }

  componentWillUnmount(){

    console.log("Component will unmount");

  }

  func(){

    this.setState({

      value : this.state.value + 1

    })

  }

  render(){

    return(

      <div>

          <button onClick={this.func}>

            Testing

          </button>

      </div>

    )

  }

}

function ParentComponent() {

  const [showComponent, setShowComponent] = useState(true);

  return (

    <div>

      <button onClick={() => setShowComponent(!showComponent)}>

        {showComponent ? "Unmount Component" : "Mount Component"}

      </button>

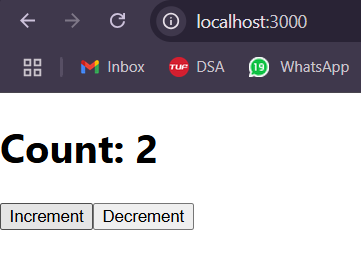
      {showComponent && <LifeCycleClassDemo value={0} />}

    </div>

  );

}

export default ParentComponent;



import {useState} from 'react';

function Counter(){

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

  const increment = () => {

    setCount(count + 1);

  };

  const decrement = () => {

    setCount(count - 1);

  };

  return (

    <div>

      <h1>Count: {count}</h1>

      <button onClick={increment}>Increment</button>

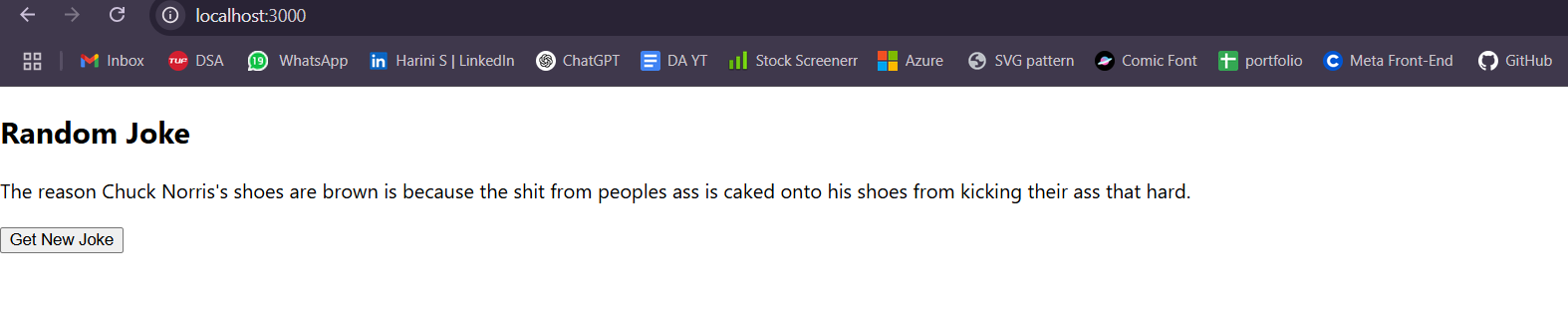
      <button onClick={decrement}>Decrement</button>

    </div>

  );

}

export default Counter



import React, { useState, useEffect } from "react";

function JokeComponent() {

  const [joke, setJoke] = useState("Loading joke...");

  // Function to fetch a new joke

  const fetchJoke = async () => {

    try {

      const response = await fetch("https://api.chucknorris.io/jokes/random");

      const data = await response.json();

      setJoke(data.value);

    } catch (error) {

      setJoke("Failed to fetch joke. Try again!");

      console.error("Error fetching joke:", error);

    }

  };

  // Fetch joke when component mounts

  useEffect(() => {

    fetchJoke();

  }, []);

  return (

    <div>

      <h2>Random Joke</h2>

      <p>{joke}</p>

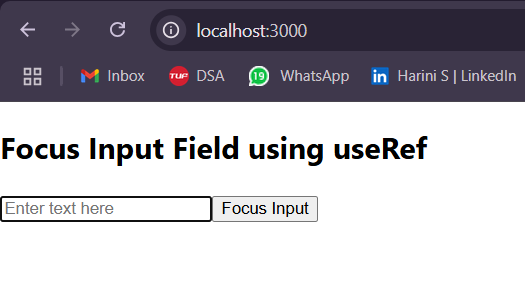
      <button onClick={fetchJoke}>Get New Joke</button>

    </div>

  );

}

export default JokeComponent;



import React, { useRef } from "react";

function InputFocusForm() {

  const inputRef = useRef(null);

  const handleFocus = () => {

    inputRef.current.focus();

  };

  return (

    <div>

      <h2>Focus Input Field using useRef</h2>

      <input ref={inputRef} type="text" placeholder="Enter text here" />

      <button onClick={handleFocus}>Focus Input</button>

    </div>

  );

}

export default InputFocusForm;

**NOTE: UseContext and UseReducer qs were not taught, hence wasn’t done**