VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University) Hyderabad
– 500031

DEPARTMENT OF : **CSE**

NAME OF THE LABORATORY : **FSWD**

Name: Roll No. Page No.

1)React code to build a simple search filter functionality to display a filtered list based on the search query entered by the user.

```
import React, { useState } from "react";
export default function App() {
 const itemList = [
  "Apple",
  "Orange",
  "Banana",
  "Cherry",
  "Milk",
  "Peanuts",
  "Butter",
  "Tomato"
 ];
 const [filteredList, setFilteredList] = new useState(itemList);
 const filterBySearch = (event) => {
  // Access input value
  const query = event.target.value;
  // Create copy of item list
  var updatedList = [...itemList];
  // Include all elements which includes the search query
  updatedList = updatedList.filter((item) => {
```

```
  return item.toLowerCase().indexOf(query.toLowerCase()) !== -1;
  });
  setFilteredList(updatedList);
 };
 return (
  <div className="App">
   <div className="search-header">
    <div className="search-text">Search:</div>
    <input id="search-box" onChange={filterBySearch} />
   </div>
   <div id="item-list">
    <ol>
     {filteredList.map((item, index) => (
      <li key={index}>{item}</li>
     ))}
    </ol>
   </div>
  </div>
 );
}
```

VASAVI COLLEGE OF ENGINEERINGS

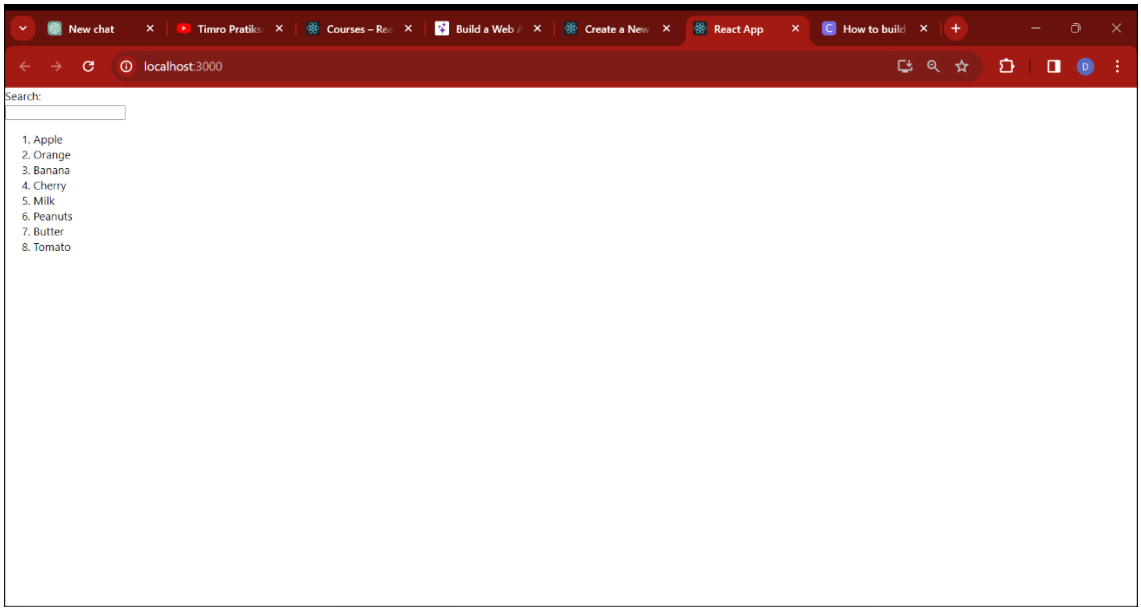
(AUTONOMOUS)
(Affiliated to Osmania University) Hyderabad
– 500031

DEPARTMENT OF : **CSE**

NAME OF THE LABORATORY : **FSWD**

Name: Roll No. Page No.

OUTPUT:



VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University) Hyderabad
– 500031

DEPARTMENT OF : **CSE**

NAME OF THE LABORATORY : **FSWD**

Name: Roll No. Page No.

2) Creating a simple counter using React which increments or decrements count dynamically on-screen as the user clicks on the button. This exercise requires knowledge of fundamental React concepts such as State, Component, etc.

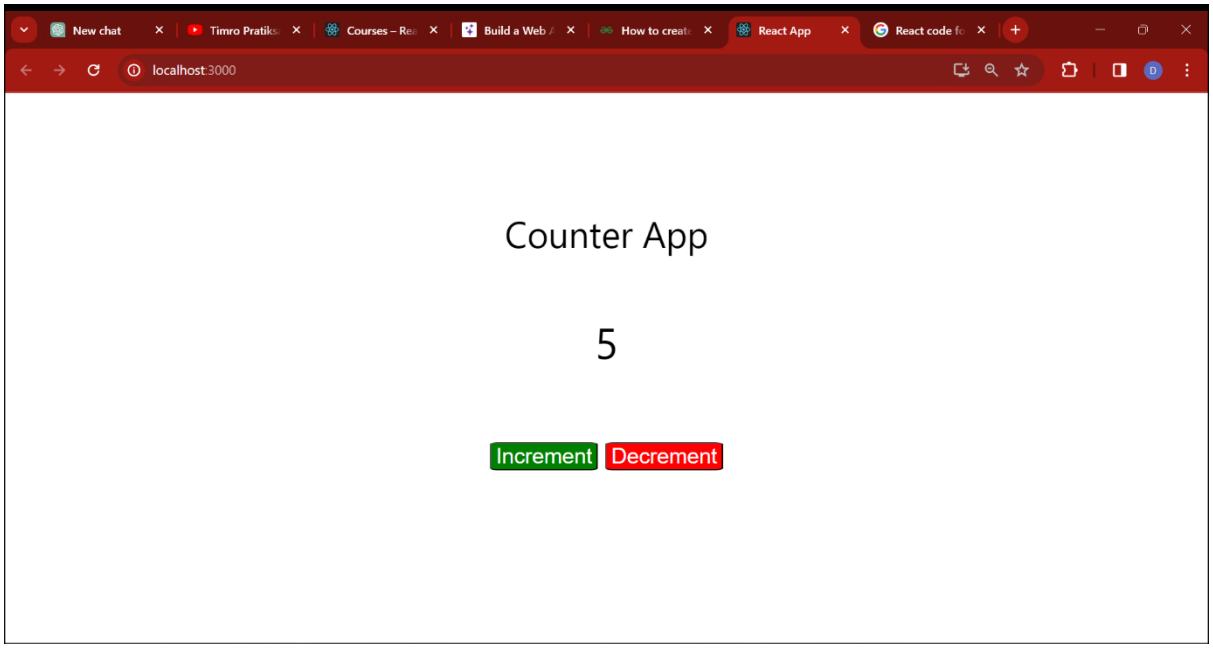
```
import React, { useState } from "react";
import "./App.css";
const App = () => {
  const [counter, setCounter] = useState(0);

  const handleClick1 = () => {

    setCounter(counter + 1);
  };

  const handleClick2 = () => {
    setCounter(counter - 1);
  };

  return (
    <div
      style={{
        display: "flex",
        flexDirection: "column",
        alignItems: "center",
```

```
justifyContent: "center",
          fontSize: "300%",
          position: "absolute",
          width: "100%",
          height: "100%",
          top: "-15%",
        }}
      >
        Counter App
        <div
          style={{
            fontSize: "120%",
            position: "relative",
            top: "10vh",
          }}
        >
          {counter}
        </div>
        <div className="buttons">
          <button
            style={{
              fontSize: "60%",
              position: "relative",
              top: "20vh",
```

```
marginRight: "5px",
            backgroundColor: "green",
            borderRadius: "8%",
            color: "white",
          }}
          onClick={handleClick1}
        >
          Increment
        </button>
        <button
          style={{
            fontSize: "60%",
            position: "relative",
            top: "20vh",
            marginLeft: "5px",
            backgroundColor: "red",
            borderRadius: "8%",
            color: "white",
          }}
          onClick={handleClick2}
        >
          Decrement
        </button>
      </div>
```

```
 </div>
  );
};
export default App;
```

OUTPUT:



VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University) Hyderabad
– 500031

DEPARTMENT OF : **CSE**

NAME OF THE LABORATORY : **FSWD**

Name: Roll No. Page No.

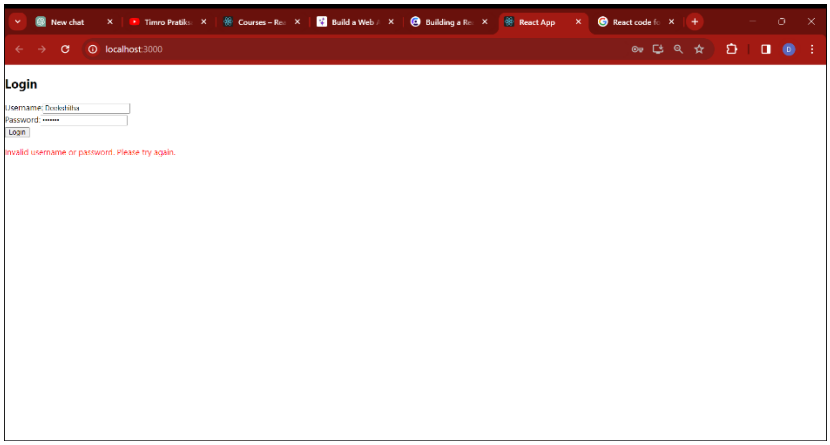
3) React code for simple login form where the user login by entering their username and password. The form inputs are validated to check if correct information is entered and the error messages are the validation fails. The login form is hidden and the “Welcome, ${name}” message is shown when the user login is successful.

```
import React, { useState } from 'react';


const LoginForm = () => {
 // State variables
 const [username, setUsername] = useState('');
 const [password, setPassword] = useState('');
 const [error, setError] = useState('');
 const [isLoggedIn, setIsLoggedIn] = useState(false);


 // Function to handle form submission
 const handleLogin = () => {
  // Validation logic
  if (!username || !password) {
   setError('Please enter both username and password.');
   return;
  }


  // Simulating authentication (you can replace this with actual authentication logic)
  if (username === 'user' && password === 'password') {
   setIsLoggedIn(true);
   setError('');
```

```
} else {
    setError('Invalid username or password. Please try again.');
  }
 };


 return (
  <div>
   {!isLoggedIn ? (
    <div>
     <h2>Login</h2>
     <div>
      <label>Username:</label>
      <input
       type="text"
       value={username}
       onChange={(e) => setUsername(e.target.value)}
      />
     </div>
     <div>
      <label>Password:</label>
      <input
       type="password"
       value={password}
       onChange={(e) => setPassword(e.target.value)}
```

```
/>
        </div>
        <button onClick={handleLogin}>Login</button>
        {error && <p style={{ color: 'red' }}>{error}</p>}
      </div>
    ) : (
      <div>
        <h2>Welcome, {username}!</h2>
        {/* Add your content for logged-in users here */}
      </div>
    )}
  </div>
 );
};
export default LoginForm;
```

OUTPUT:



VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

(Affiliated to Osmania University)

Hyderabad – 500 031.

DEPARTMENT OF : _____

NAME OF THE LABORATORY : _____

Name_____Roll No._____Page No._____

1)React exercise to create an image slide, where users can view multiple images with next/previous buttons. Additionally, there is also an option to select an image from any index of the list through a click-on option circle.

```
import React, { useState } from 'react';

const ImageSlider = ({ images }) => {
 const [currentIndex, setCurrentIndex] = useState(0);

 const goToNext = () => {
  setCurrentIndex((prevIndex) => (prevIndex + 1) % images.length);
 };

 const goToPrevious = () => {
  setCurrentIndex((prevIndex) => (prevIndex - 1 + images.length) % images.length);
 };

 const goToIndex = (index) => {
  setCurrentIndex(index);
 };

 return (
  <div className="image-slider">
   <div className="image-container">
    <img src={images[currentIndex]} alt={`Image ${currentIndex}`} />
   </div>
   <div className="controls">
    <button onClick={goToPrevious}>Previous</button>
    <button onClick={goToNext}>Next</button>
   </div>
   <div className="option-circles">
    {images.map((image, index) => (
     <div
      key={index}
      className={`option-circle ${index === currentIndex ? 'active' : ''}`}
      onClick={() => goToIndex(index)}
     />
    ))}
   </div>
  </div>);};
```

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

(Affiliated to Osmania University)

Hyderabad – 500 031.

DEPARTMENT OF : _____

NAME OF THE LABORATORY : _____

Name_____Roll No._____Page No._____

```
                        const App = () => {
  const images = [
    'https://img.freepik.com/free-photo/painting-mountain-lake-with-mountain-
background_188544-9126.jpg',
    'https://images.pexels.com/photos/268533/pexels-photo-
268533.jpeg?auto=compress&cs=tinysrgb&dpr=1&w=500',
    'https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcQ9nv6zOdBK7FnnGLttVekKF7SXQnOwNrgrIb1xQnx_Uq2N
qIi5eRJQEahRJw&s',
    // Add more image URLs as needed
  ];

  return (
    <div className="app">
      <h1>Image Slider</h1>
      <ImageSlider images={images} />
    </div>
  );
};

export default App;
```

OUTPUT:

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

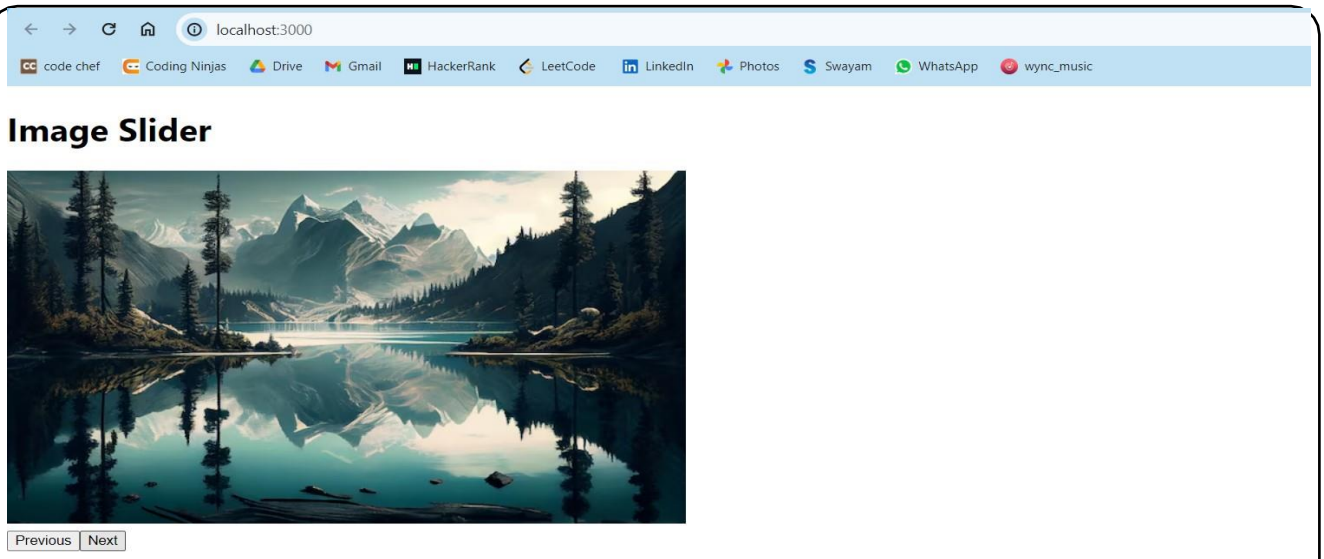
(Affiliated to Osmania University)

Hyderabad – 500 031.

DEPARTMENT OF : _____

NAME OF THE LABORATORY : _____

Name_____Roll No._____Page No._____



2)React code to display a checklist with multiple options that can select and the selected options are dynamically displayed on the screen. React State is used to keep track of checked options and onChange Event handler is triggered to alter the state whenever an option is checked or unchecked.

```
import React, { useState } from 'react';

const Checklist = () => {
  const [checkedOptions, setCheckedOptions] = useState([]);

  const handleCheckboxChange = (option) => {
    if (checkedOptions.includes(option)) {
      // If the option is already checked, remove it from the array
      setCheckedOptions((prevOptions) => prevOptions.filter((item) => item !== option));
    } else {
      // If the option is not checked, add it to the array
      setCheckedOptions((prevOptions) => [...prevOptions, option]);
    }
  };

  return (
    <div>
      <h2>Checklist</h2>
      <form>
        {['Option 1', 'Option 2', 'Option 3', 'Option 4'].map((option, index) => (
```

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

(Affiliated to Osmania University)

Hyderabad – 500 031.

DEPARTMENT OF : _____

NAME OF THE LABORATORY : _____

Name_____Roll No._____Page No._____

```
<div key={index} className="checkbox">
          <input
            type="checkbox"
            id={option}
            checked={checkedOptions.includes(option)}
            onChange={() => handleCheckboxChange(option)}
          />
          <label htmlFor={option}>{option}</label>
        </div>
      ))}
    </form>
    <div>
      <h3>Selected Options:</h3>
      <ul>
        {checkedOptions.map((option, index) => (
          <li key={index}>{option}</li>
        ))}
      </ul>
    </div>
  </div>
 );
};

const App = () => {
 return (
  <div className="app">
    <h1>React Checklist</h1>
    <Checklist />
  </div>
 );
};

export default App;
```

OUTPUT:

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

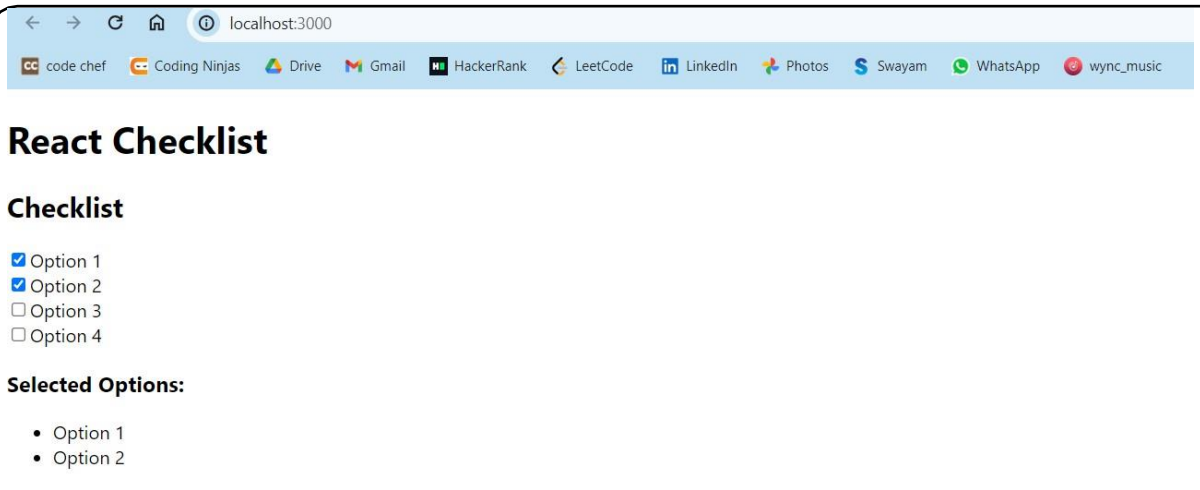
(Affiliated to Osmania University)

Hyderabad – 500 031.

DEPARTMENT OF : _____

NAME OF THE LABORATORY : _____

Name_____Roll No._____Page No._____



3)Creating a simple counter using React which increments or decrements count dynamically on-screen as the user clicks on the button. This exercise requires knowledge of fundamental React concepts such as State, Component, etc. We can complete the simple counter exercise with the following steps:

1. Create React state to store the count value.
2. Declare JS functions to incement or decrement the value through setState.

Add HTML buttons with on Click to JSX code.

```
import React, { useState } from 'react';

const Counter = () => {
  // Step 1: Create React state to store the count value
  const [count, setCount] = useState(0);

  // Step 2: Declare JS functions to increment or decrement the value through setState
  const increment = () => {
    setCount((prevCount) => prevCount + 1);
  };

  const decrement = () => {
    setCount((prevCount) => (prevCount > 0 ? prevCount - 1 : 0));
  };

  return (
    <div>
      <h2>Counter App</h2>
```

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

(Affiliated to Osmania University)

Hyderabad – 500 031.

DEPARTMENT OF : _____

NAME OF THE LABORATORY : _____

Name_____Roll No._____Page No._____

```
    <p>Count: {count}</p>
  {/* Add HTML buttons with onClick to JSX code */}
    <button onClick={increment}>Increment</button>
    <button onClick={decrement}>Decrement</button>
   </div>
  );
};

const App = () => {
  return (
   <div className="app">
    <h1>React Counter Example</h1>
    <Counter />
   </div>
  );
};

export default App;
```

OUTPUT:

React Counter Example

Counter App

Count: 6

Increment Decrement

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad – 500031

DEPARTMENT OF : **CSE**
NAME OF THE LABORATORY : **FSWD**

Name _____Roll No._____Page No.

1.Write a program to create,read,write,delete and append files using node js.

To create file:

```
var fs = require('fs');

fs.appendFile('mynewfile1.txt', 'Hello content!', function (err) {
  if (err) throw err;
  console.log('Saved!');
});
```

Output:

```
Saved!
```

To read file:

```
var fs=require('fs');

fs.readfile('mynewfile1.txt','Utf-8',(err,data) => {
iff(err) {
console.error(err);
return ;
}
console.log(data);
});
```

Output:

To write in to a file:

```
var fs = require('fs');

fs.writeFile('mynewfile3.txt', 'Hello content!', function (err) {
  if (err) throw err;
  console.log('Saved!');
});
```

Output:

```
Saved!
```

To update a file:

```
var fs = require('fs');

fs.appendFile('mynewfile1.txt', ' This is my text.', function (err) {
  if (err) throw err;
  console.log('Updated!');
});
```

Output:

```
Updated!
```

To delete a file:

```
var fs = require('fs');

fs.unlink('mynewfile2.txt', function (err) {
  if (err) throw err;
  console.log('File deleted!');
});
```

Output:

```
File deleted!
```

To rename a file:

```
var fs = require('fs');

fs.rename('mynewfile1.txt', 'myrenamedfile.txt', function (err) {
  if (err) throw err;
  console.log('File Renamed!');
});
```

Output:

```
File Renamed!
```

2.Write a program to create and run express module.

```
var express = require('express');
var app = express();
app.get('/', function (req, res) {
  res.send('Welcome to JavaTpoint!');
});
var server = app.listen(8000, function () {
  var host = server.address().address;
  var port = server.address().port;
  console.log('Example app listening at http://%s:%s', host, port);
});
```

Output: