Name : Khushi Ingale
Assignment 34 - React.js

1. Building a List of Items Using JSX, Passing Data from a JavaScript Array
->

```
import React from 'react';

const ItemList = () => {
  const items = ['Apple', 'Banana', 'Cherry', 'Date', 'Elderberry'];

  return (
    <div>
      <h2>Fruit List</h2>
      <ul>
        {items.map((item, index) => (
          <li key={index}>{item}</li>
        ))}
      </ul>
    </div>
  );
}

export default ItemList;
```

2. Implementing a Basic Counter That Updates the Display When a Button is Clicked
->

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

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

  const increment = () => {
    setCount(count + 1);
  };

  return (
    <div>
      <h2>Counter: {count}</h2>
      <button onClick={increment}>Increase Count</button>
    </div>
  );
}

export default Counter;
```

3. Creating a Form with Input Fields and a Submit Button, Handling User Input and Displaying the Data
->

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

const Form = () => {
  const [inputValue, setInputValue] = useState('');

  const handleSubmit = (e) => {
    e.preventDefault();
    alert(`Submitted: ${inputValue}`);
  };

  return (
    <div>
      <h2>Submit Form</h2>
      <form onSubmit={handleSubmit}>
        <input
```

```
          type="text"
          value={inputValue}
          onChange={(e) => setInputValue(e.target.value)}
        />
        <button type="submit">Submit</button>
      </form>
    </div>
  );
}

export default Form;
```

4. Fetching Data from an API and Displaying it in a List

->

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

const FetchData = () => {
  const [data, setData] = useState([]);

  useEffect(() => {
    fetch('https://jsonplaceholder.typicode.com/posts')
      .then(response => response.json())
      .then(data => setData(data));
  }, []);

  return (
    <div>
      <h2>Fetched Posts</h2>
      <ul>
        {data.map(post => (
          <li key={post.id}>
            <h3>{post.title}</h3>
            <p>{post.body}</p>
          </li>
        ))}
      </ul>
    </div>
  );
}

export default FetchData;
```

5. Building a Simple To-Do List Application with Adding, Deleting, and Marking Tasks as Complete

->

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

const TodoList = () => {
  const [tasks, setTasks] = useState([]);
  const [taskInput, setTaskInput] = useState('');

  const addTask = () => {
    if (taskInput) {
      setTasks([...tasks, { text: taskInput, completed: false }]);
      setTaskInput('');
    }
  };

  const deleteTask = (index) => {
    const updatedTasks = tasks.filter((_, i) => i !== index);
    setTasks(updatedTasks);
  };

  const toggleTaskCompletion = (index) => {
    const updatedTasks = tasks.map((task, i) =>
      i === index ? { ...task, completed: !task.completed } : task
    );
    setTasks(updatedTasks);
  };

  return (
    <div>
      <h2>Todo List</h2>
      <input
        type="text"
        value={taskInput}
        onChange={(e) => setTaskInput(e.target.value)}
        placeholder="Add a new task"
      />
      <button onClick={addTask}>Add Task</button>

      <ul>
        {tasks.map((task, index) => (
          <li key={index} style={{ textDecoration: task.completed ? 'line-
through' : 'none' }}>
            <span onClick={() => toggleTaskCompletion(index)}>{task.text}</span>
            <button onClick={() => deleteTask(index)}>Delete</button>
          </li>
        ))}
      </ul>
    </div>
  );
}

export default TodoList;
```