



Assignment 3

Name: Mubeen imran

Roll: 01-135221-067

For your project, create all the components of your landing page.

These are the components:

1. **Todo.js**
2. **Todoform.js**
3. **Todolist.js**

1. **Todo.js**

Code:

```
import React, { useState } from 'react';
import TodoForm from './TodoForm';
import { RiCloseCircleLine } from 'react-icons/ri';
import { TiEdit } from 'react-icons/ti';

const Todo = ({ todos, completeTodo, removeTodo, updateTodo }) => {
  const [edit, setEdit] = useState({
    id: null,
    value: ''
  });

  const submitUpdate = value => {
    updateTodo(edit.id, value);
    setEdit({
      id: null,
      value: ''
    });
  };

  if (edit.id) {
    return <TodoForm edit={edit} onSubmit={submitUpdate} />;
  }

  return todos.map((todo, index) => (
    <div
      className={todo.isComplete ? 'todo-row complete' : 'todo-row'}
      key={index}
    >
```

```

    <div key={todo.id} onClick={() => completeTodo(todo.id)}>
      {todo.text}
    </div>
    <div className='icons'>
      <RiCloseCircleLine
        onClick={() => removeTodo(todo.id)}
        className='delete-icon'
      />
      <TiEdit
        onClick={() => setEdit({ id: todo.id, value: todo.text })}
        className='edit-icon'
      />
    </div>
  </div>
));
};

export default Todo;

```

2. Todoform.js

Code:

```

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

function TodoForm(props) {
  const [input, setInput] = useState(props.edit ? props.edit.value : '');

  const inputRef = useRef(null);

  useEffect(() => {
    inputRef.current.focus();
  });

  const handleChange = e => {
    setInput(e.target.value);
  };

  const handleSubmit = e => {
    e.preventDefault();
  };

```

```

    props.onSubmit({
      id: Math.floor(Math.random() * 10000),
      text: input
    });
    setInput('');
  };

  return (
    <form onSubmit={handleSubmit} className='todo-form'>
      {props.edit ? (
        <>
          <input
            placeholder='Update your item'
            value={input}
            onChange={handleChange}
            name='text'
            ref={inputRef}
            className='todo-input edit'
          />
          <button onClick={handleSubmit} className='todo-button edit'>
            Update
          </button>
        </>
      ) : (
        <>
          <input
            placeholder='Add a todo'
            value={input}
            onChange={handleChange}
            name='text'
            className='todo-input'
            ref={inputRef}
          />
          <button onClick={handleSubmit} className='todo-button'>
            Add todo
          </button>
        </>
      )}
    </form>
  );
}

export default TodoForm;

```

3. Todolist.js

Code:

```
import React, { useState } from 'react';
import TodoForm from './TodoForm';
import Todo from './Todo';

function TodoList() {
  const [todos, setTodos] = useState([]);

  const addTodo = todo => {
    if (!todo.text || /\s*$/.test(todo.text)) {
      return;
    }

    const newTodos = [todo, ...todos];

    setTodos(newTodos);
    console.log(...todos);
  };

  const updateTodo = (todoId, newValue) => {
    if (!newValue.text || /\s*$/.test(newValue.text)) {
      return;
    }

    setTodos(prev => prev.map(item => (item.id === todoId ? newValue : item)));
  };

  const removeTodo = id => {
    const removedArr = [...todos].filter(todo => todo.id !== id);

    setTodos(removedArr);
  };

  const completeTodo = id => {
    let updatedTodos = todos.map(todo => {
      if (todo.id === id) {
        todo.isComplete = !todo.isComplete;
      }
      return todo;
    });
    setTodos(updatedTodos);
  };
}
```

```
return (  
  <>  
    <h1>What's the Plan for Today?</h1>  
    <TodoForm onSubmit={addTodo} />  
    <Todo  
      todos={todos}  
      completeTodo={completeTodo}  
      removeTodo={removeTodo}  
      updateTodo={updateTodo}  
    />  
  </>  
);  
}  
  
export default TodoList;
```