**Geco ReactJS Training**

|  |  |
| --- | --- |
| ReactJS Assignment |  |
| Name: | Ho Choo Geok |
| Date of Issue: | 1st November 2022 |
| Date of Submission: | 1st November 2022 |
| Cohort No.: | 27 |
| Week No.: | 4 |
| Session No.: | 14 |

1. Create a to-do application with Storybook and recoil.

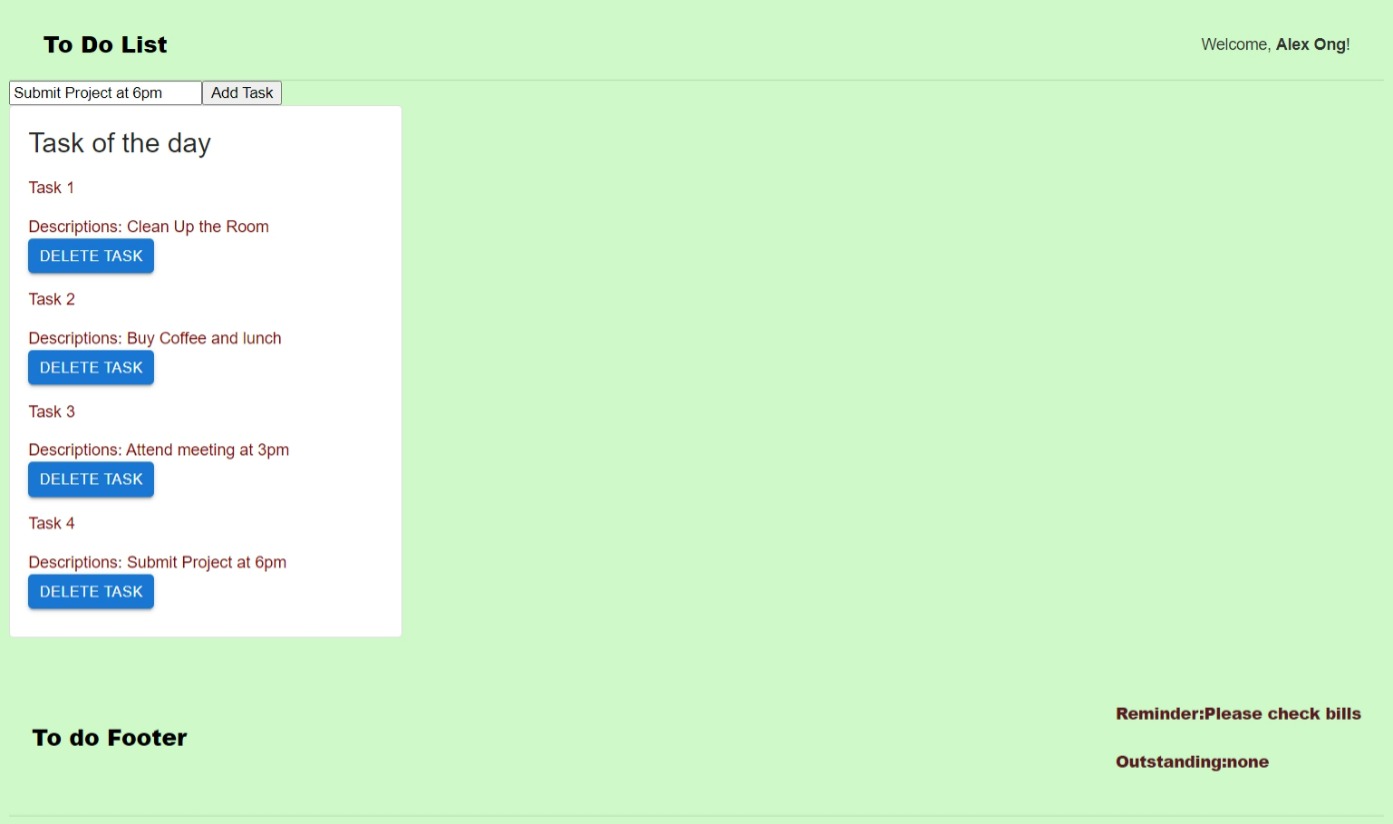
It will contain header, footer, form in the UI.

Capture data from user and display below in the form of cards.

There should be delete button on each card, when user click on it delete the record user has clicked on.

Implement tree-shaking.

Header in Storybook

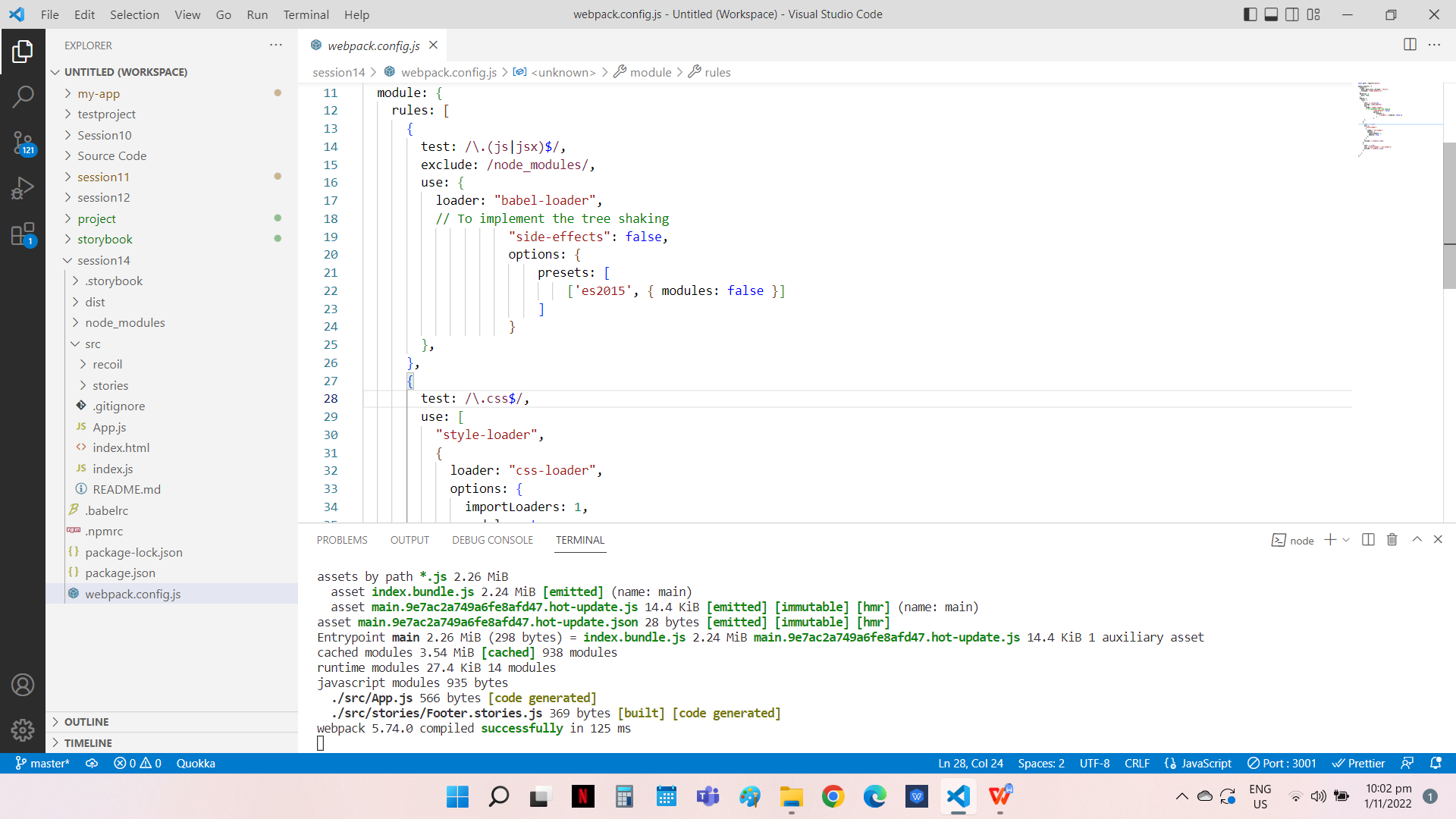


Footer in Storybook

Card display of task. Click delete task to delete a task.

Forms input for Task description

**Tree shaking**



**Source Code**

**File Name: atomData.js**

import {atom} from "recoil";

export const atomData = atom({

    key:'todoListState',

    default:[]

});

**File Name: DisplayData.js**

import React from 'react';

import {useRecoilState} from "recoil";

import {atomData} from './atomData';

import {Form} from "./Form";

import {Card,CardContent,Button,Typography} from '@mui/material/';

export const DisplayData = () => {

    const[list,setList] = useRecoilState(atomData);

    console.log(list);

    const handleUpdate = (e,id) => {

        const newList = list.filter( (info => {

            return info.id !== id;

        }));

        console.log(id)

        setList(newList);

    }

    return(

        <div>

            <Form />

        <Card sx={{ maxWidth: 345 }} variant="outlined">

        <CardContent>

        <Typography gutterBottom variant="h5" component="div">

          Task of the day

        </Typography>

        <Typography variant="body2" color="darkred">

         {list.length>0 ? (list.map(info => {

                return(

                    <React.Fragment key={info.id}>

                       <p> Task {info.id}  </p>

                       <span> Descriptions: {info.text}  </span>

                       <Button size="small" variant="contained" onClick={(e) => handleUpdate(e,info.id)} >Delete TasK</Button>

                        {/\* <Button size="Large" onClick={(e) => handleUpdate(e,info.id)}>Delete</Button> \*/}

                       {/\* <p onClick={(e) => handleUpdate(e,info.id)}> {info.text} </p> \*/}

                       </React.Fragment>

             )})) : (<> </>)}

        </Typography>

      </CardContent>

    </Card>

        </div>

    )

};

**File Name: Form.js**

import React, {useState} from 'react';

import {atomData} from "./atomData";

import {useRecoilState} from 'recoil';

export const Form = () => {

    function getid(){

        setId(id+1)

        return id  ;

    }

    const [inputValue,setInputValue] = useState("");

    const [id,setId] = useState(0);

    const [val,setVal] = useRecoilState(atomData);

    const handleChange = (e) =>{

        if (e.target.value !==''){

        setInputValue(e.target.value)

        }

    }

    const addData = () => {

        if (inputValue) {setVal( (list) => [

            ...list,

            {text:inputValue,

                id: getid(),

            }

        ])}

       }

    return(

        <div>

            <input type="text" value={inputValue} onChange={handleChange} placeholder="Task"/>

            <button onClick={addData}> Add Task </button>

        </div>

    )

};

**File Name: App.js**

import React from 'react';

import {Name} from "./stories/Header.stories";

import {Display} from "./stories/Footer.stories";

import {DisplayData} from './recoil/DisplayData';

function App() {

  return (

     <>

     <Name username="Alex Ong" />

     <DisplayData />

     <br/>

     <Display reminder="Please check bills" outstanding="none" />

    </>

  );

};

export default App;

**File Name: Header.js**

import React from 'react';

import PropTypes from 'prop-types';

import './header.css';

export const Header = ({username}) => (

  <header>

    <div className="wrapper">

      <div>

       <h1>To Do List </h1>

      </div>

      <div>

             <span className="welcome">

              Welcome, <b>{username}</b>!

             </span>

      </div>

    </div>

  </header>

);

Header.propTypes = {

  username: PropTypes.string.isRequired,

};

Header.defaultProps = {

  username: null,

 };

**File Name: Header.stories.js**

import React from 'react';

import { Header } from './Header';

export default {

  title: 'Example/Header',

  component: Header,

  parameters: {

    layout: 'fullscreen',

  },

};

const Template = (args) => <Header {...args} />;

export const Name = Template.bind({});

Name.args = {

  Name : {},

  };

**File Name: Footer.stories.js**

import React from 'react';

import { Footer } from './Footer';

export default {

  title: 'Example/Footer',

  component: Footer,

  parameters: {

    layout: 'fullscreen',

  },

  };

const Template = (args) => <Footer {...args} />;

export const Display = Template.bind({});

Display.args = {

  reminder:{} ,

  outstanding:{},

  };

**File Name: Footer.js**

import React from 'react';

import PropTypes from 'prop-types';

import './footer.css';

export const Footer = ({reminder,outstanding}) => (

   <footer>

      <div className="wrapper">

      <div>

       <h3>To do Footer</h3>

      </div>

      <div className='text'>

      <h5> Reminder:{reminder}</h5>

      <h5> Outstanding:{outstanding}</h5>

      </div>

    </div>

  </footer>

);

Footer.propTypes = {

  reminder: PropTypes.string,

  outstanding: PropTypes.string,

  };

Footer.defaultProps = {

  reminder: "Reminder",

  outstanding: "NA",

};