Creating Project:

Npx create-react-app linkedin –template redux

Dependencies:

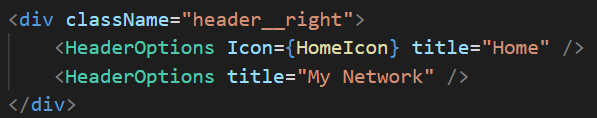
Yarn add @material-ui/core

Yarn add @material-ui/icons

Yarn add firebase@^8.10.0

To pass Icon as props or component:

Make Icon (i) capital as we are passing material-ui Icon



import React from 'react'

import './HeaderOptions.css';

function HeaderOptions({Icon, title}) {

    return (

        <div className="headerOption">

            {Icon && <Icon className="headerOption\_\_icon"/>}

            <h3>{title}</h3>

        </div>

    )

}

export default HeaderOptions

align items work with flexbox, flex direction row or colum

.headerOption{

    display: flex;

    flex-direction: column;

    align-items: center;

    margin-right: 20px;

    color: gray;

    cursor: pointer;

}

Function inside function declare and calling

import { Avatar } from '@material-ui/core'

import React from 'react'

import './Sidebar.css'

function Sidebar() {

    const recentItem = (topic) => (

        <div className="sidebar\_\_recentItem">

            <span className="sidebar\_\_hash">#</span>

            <p>{topic}</p>

        </div>

    )

    return (

        <div className="sidebar">

            <div className="sidebar\_\_bottom">

                <p>Recent</p>

                {recentItem('reactjs')}

                {recentItem('programming')}

                {recentItem('fullstack')}

                {recentItem('design')}

                {recentItem('developer')}

            </div>

        </div>

    )

}

export default Sidebar

To limit text with in area or div or body:

Used overflow-wrap: anywhere;

            <div className="post\_\_body">

                <p>{message}</p>

            </div>

.post\_\_body{

    overflow-wrap: anywhere;

}

Firebase for authentication and db:

import firebase from 'firebase';

const firebaseConfig = {

  apiKey: "AIzaSyDUdfLCHFAHvs9sc5h7O-jJfJ0P8n1U5BY",

  authDomain: "linkedin-311e5.firebaseapp.com",

  projectId: "linkedin-311e5",

  storageBucket: "linkedin-311e5.appspot.com",

  messagingSenderId: "650316104831",

  appId: "1:650316104831:web:55793589eabf36d1b16b4a"

};

// Initialize Firebase

const firebaseApp = firebase.initializeApp(firebaseConfig);

const db = firebaseApp.firestore();

const auth = firebase.auth();

export {db, auth};

Firebase update and collection:

Adding data to collection

    const sendPost = (e) => {

        e.preventDefault()

        db.collection('posts').add({

            name: "Muneeb",

            description: "test",

            message: input,

            photourl: "",

            timestamp: firebase.firestore.FieldValue.serverTimestamp(),

        })

    }

Getting and updating data

    useEffect(()=>{

        db.collection('posts').onSnapshot(snapshot=> {

            setPosts(snapshot.docs.map(doc=>(

                {

                    id: doc.id,

                    data: doc.data()

                }

            )))

        })

    }, [])

Now using data from collection

            {posts.map( ({ id, data: { name, description, message, photourl }} )=>{

                <Post

                    key= {id}

                    name={name}

                    description={description}

                    message={message}

                    photourl={photourl}

                />

            })}