Anikets :-

import \* as React from 'react';

import './App.css';

import Avatar from '@mui/material/Avatar';

import Button from '@mui/material/Button';

import CssBaseline from '@mui/material/CssBaseline';

import TextField from '@mui/material/TextField';

import FormControlLabel from '@mui/material/FormControlLabel';

import Checkbox from '@mui/material/Checkbox';

import Link from '@mui/material/Link';

import Grid from '@mui/material/Grid';

import Box from '@mui/material/Box';

import LockOutlinedIcon from '@mui/icons-material/LockOutlined';

import Typography from '@mui/material/Typography';

import Container from '@mui/material/Container';

import { createTheme, ThemeProvider } from '@mui/material/styles';

function Copyright(props) {

  return (

    <Typography variant="body2" color="text.secondary" align="center" {...props}>

      {'Copyright © '}

      <Link color="inherit" href="aniketvaidya18@gmail.com">

      aniketvaidya18@gmail.com

      </Link>{' '}

      {new Date().getFullYear()}

      {'.'}

    </Typography>

  );

}

*// TODO remove, this demo shouldn't need to reset the theme.*

const defaultTheme = createTheme();

export default function SignIn() {

  const handleSubmit = (event) => {

    event.preventDefault();

    const data = new FormData(event.currentTarget);

    console.log({

      email: data.get('email'),

      password: data.get('password'),

    });

  };

  return (

    <ThemeProvider theme={defaultTheme}>

      <Container component="main" maxWidth="xs">

        <CssBaseline />

        <Box

          sx={{

            marginTop: 8,

            display: 'flex',

            flexDirection: 'column',

            alignItems: 'center',

          }}

        >

          <Avatar sx={{ m: 1, bgcolor: 'secondary.main' }}>

            <LockOutlinedIcon />

          </Avatar>

          <Typography component="h1" variant="h5">

            Sign in

          </Typography>

          <Box component="form" onSubmit={handleSubmit} noValidate sx={{ mt: 1 }}>

            <TextField

              margin="normal"

              required

              fullWidth

              id="email"

              label="Email Address"

              name="email"

              autoComplete="email"

              autoFocus

            />

            <TextField

              margin="normal"

              required

              fullWidth

              name="password"

              label="Password"

              type="password"

              id="password"

              autoComplete="current-password"

            />

            <FormControlLabel

              control={<Checkbox value="remember" color="primary" />}

              label="Remember me"

            />

            <Grid container>

              <Grid item xs

              sx={{ mt: 0, mb: 0, ml: 35, mr: 0 }}>

                <Link href="#" variant="body2">

                  Forgot password?

                </Link>

              </Grid>

            </Grid>

            <Button

              type="submit"

              halfWidth

              variant="contained"

              sx={{ mt: 8, mb: 2 }}

            >

              Sign In

            </Button>

            <Button

              type="submit"

              sx={{ mt: 8, mb: 2, ml: 5, mr: 5 }}

              variant="contained"

            >

              Log In

            </Button>

          </Box>

        </Box>

        <Copyright sx={{ mt: 8, mb: 4 }} />

      </Container>

    </ThemeProvider>

  );

}

Css code :-

.App {

  text-align: left;

}

.App-logo {

  height: 40vmin;

  pointer-events: none;

}

@media (prefers-reduced-motion: no-preference) {

  .App-logo {

    animation: App-logo-spin infinite 20s linear;

  }

}

.App-header {

  background-color: #282c34;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: rgb(186, 186, 186);

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}