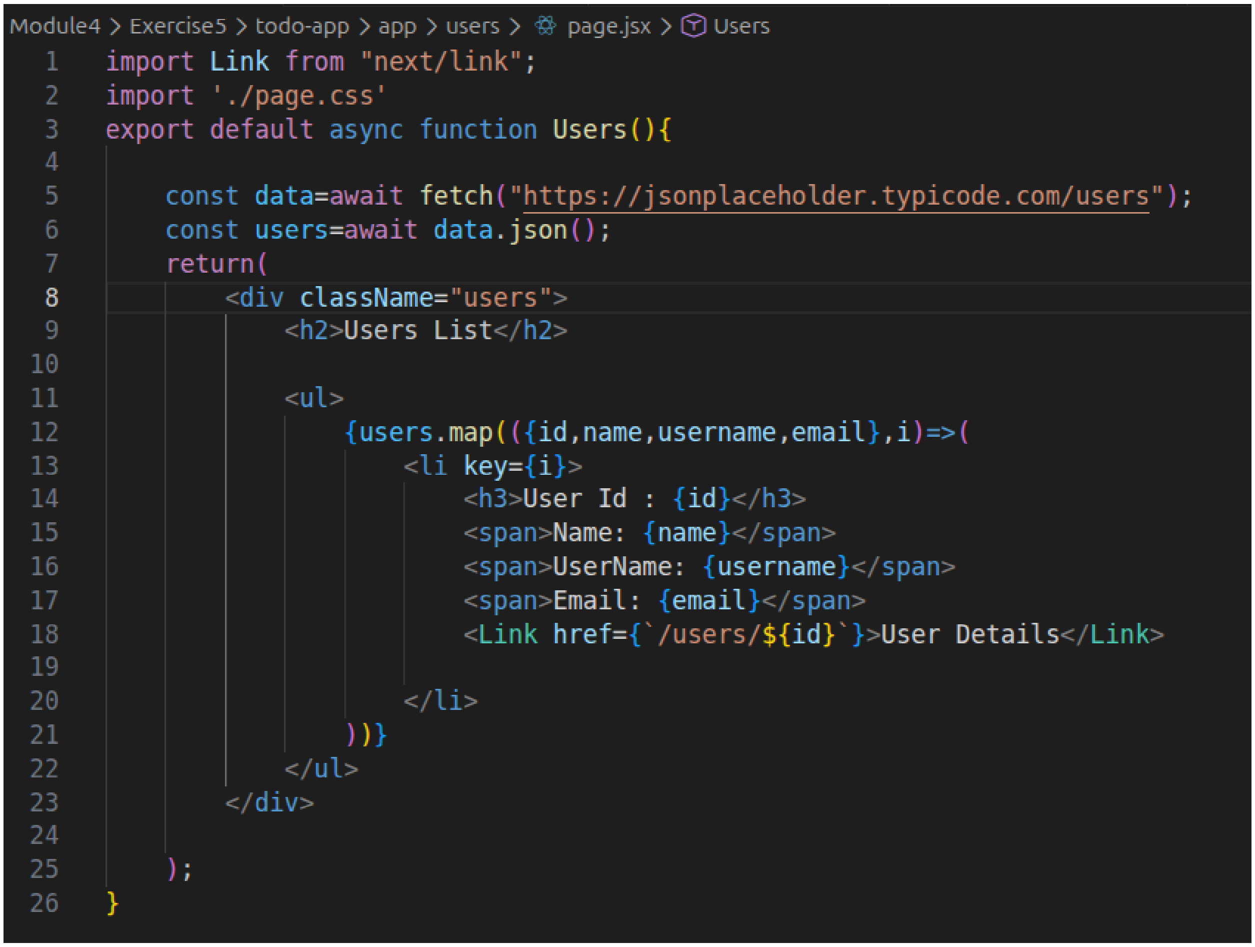
Module 4

Exercise 5 (NextJs Page & Routing)

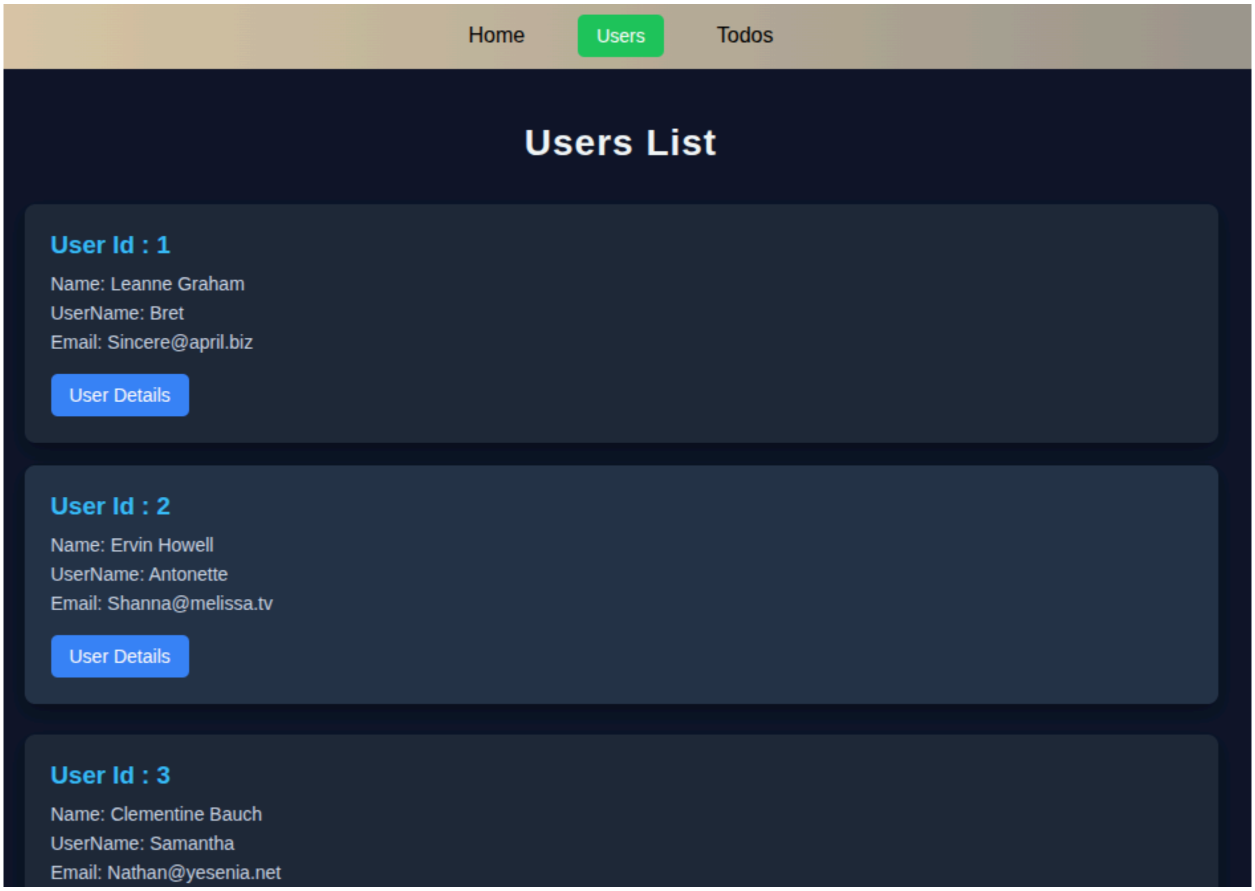
Q1.) Create a user listing page using fetch data of users using this api https://jsonplaceholder.typicode.com/users

A1.) Nextjs


Module4 > Exercise5 > todo-app > app > users > page.jsx > Users

```jsx
import Link from "next/link";
import './page.css'
export default async function Users(){

    const data=await fetch("https://jsonplaceholder.typicode.com/users");
    const users=await data.json();
    return(
        <div className="users">
            <h2>Users List</h2>

            <ul>
                {users.map(({id,name,username,email},i)=>(
                    <li key={i}>
                        <h3>User Id : {id}</h3>
                        <span>Name: {name}</span>
                        <span>UserName: {username}</span>
                        <span>Email: {email}</span>
                        <Link href={`/users/${id}`}>User Details</Link>

                    </li>
                ))}
            </ul>
        </div>

    );
}
```

Output



Users List

User Id : 1

Name: Leanne Graham

UserName: Bret

Email: Sincere@april.biz

User Id : 2

Name: Ervin Howell

UserName: Antonette

Email: Shanna@melissa.tv

User Id : 3

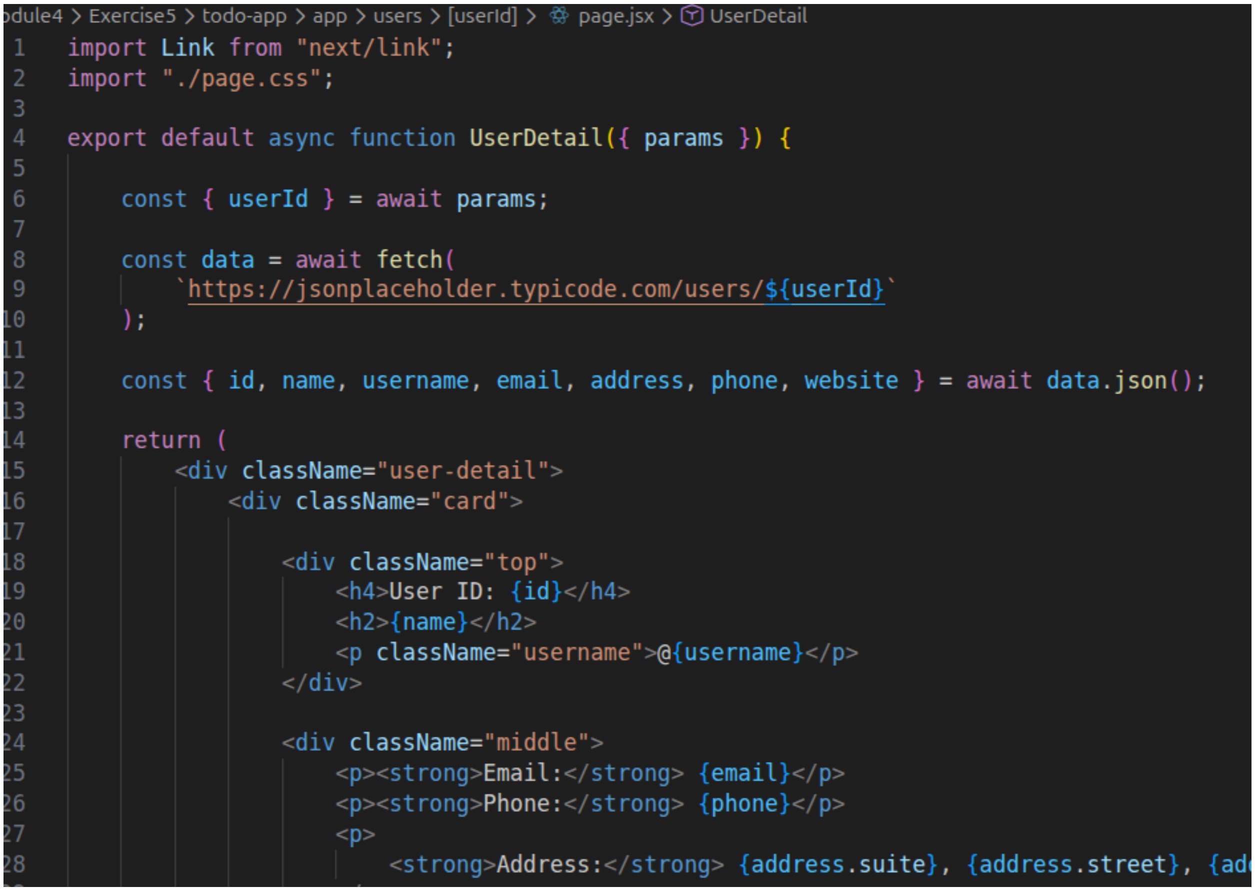
Name: Clementine Bauch

UserName: Samantha

Email: Nathan@yesenia.net

Q2.) Create a page which show individual user details using this api https://jsonplaceholder.typicode.com/users/[id]

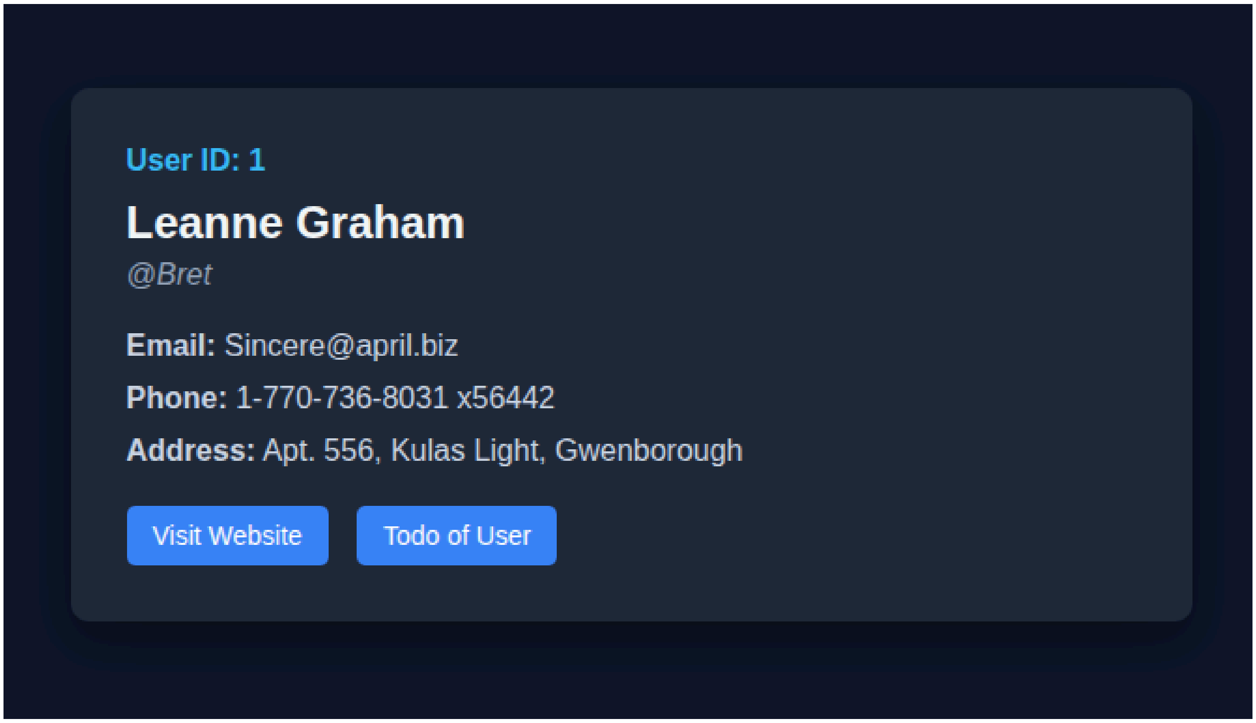
A2.) NextJs



odule4 > Exercise5 > todo-app > app > users > [userId] > page.jsx > UserDetail

```jsx
import Link from "next/link";
import "./page.css";

export default async function UserDetail({ params }) {

    const { userId } = await params;

    const data = await fetch(
        `https://jsonplaceholder.typicode.com/users/${userId}`
    );

    const { id, name, username, email, address, phone, website } = await data.json();

    return (
        <div className="user-detail">
            <div className="card">

                <div className="top">
                    <h4>User ID: {id}</h4>
                    <h2>{name}</h2>
                    <p className="username">@{username}</p>
                </div>

                <div className="middle">
                    <p><strong>Email:</strong> {email}</p>
                    <p><strong>Phone:</strong> {phone}</p>
                    <p>
                        <strong>Address:</strong> {address.suite}, {address.street}, {ad
```

Output



User ID: 1

Leanne Graham

@Bret

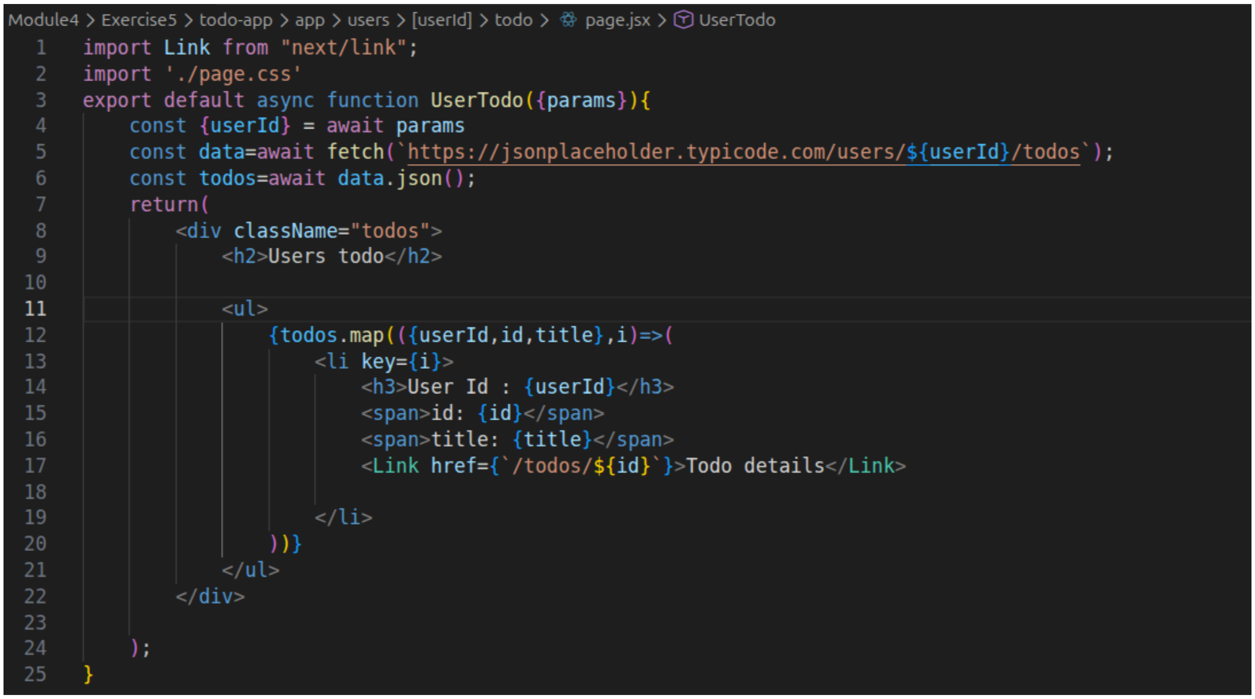
Email: Sincere@april.biz

Phone: 1-770-736-8031 x56442

Address: Apt. 556, Kulas Light, Gwenborough

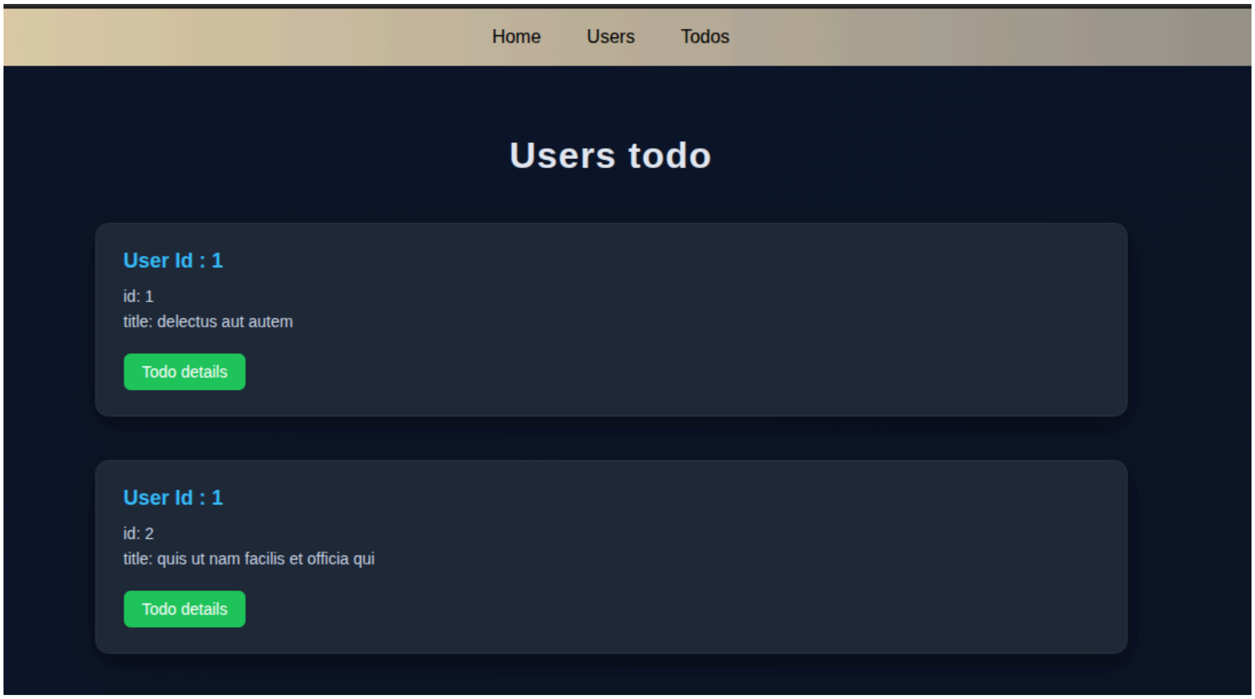
Q3.) On user Details page show todos of the that user using this api https://jsonplaceholder.typicode.com/users/1/todos

A3.) NextJs


Module4 > Exercise5 > todo-app > app > users > [userId] > todo > page.jsx > UserTodo

```
import Link from "next/link";
import './page.css'
export default async function UserTodo({params}){
    const {userId} = await params
    const data=await fetch(`https://jsonplaceholder.typicode.com/users/${userId}/todos`);
    const todos=await data.json();
    return(
        <div className="todos">
            <h2>Users todo</h2>

            <ul>
                {todos.map(({userId,id,title},i)=>(
                    <li key={i}>
                        <h3>User Id : {userId}</h3>
                        <span>id: {id}</span>
                        <span>title: {title}</span>
                        <Link href={`/todos/${id}`}>Todo details</Link>

                    </li>
                ))}
            </ul>
        </div>

    );
}
```

Output



Users todo

User Id : 1

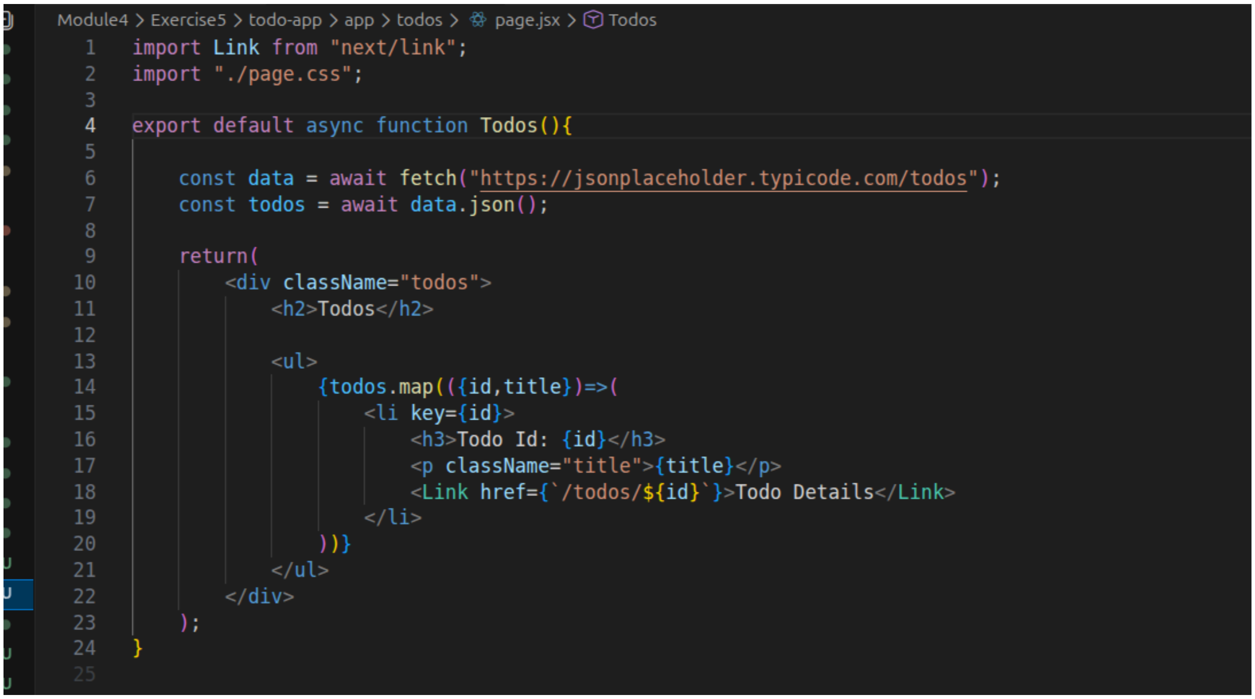
id: 1
title: delectus aut autem

User Id : 1

id: 2
title: quis ut nam facilis et officia qui

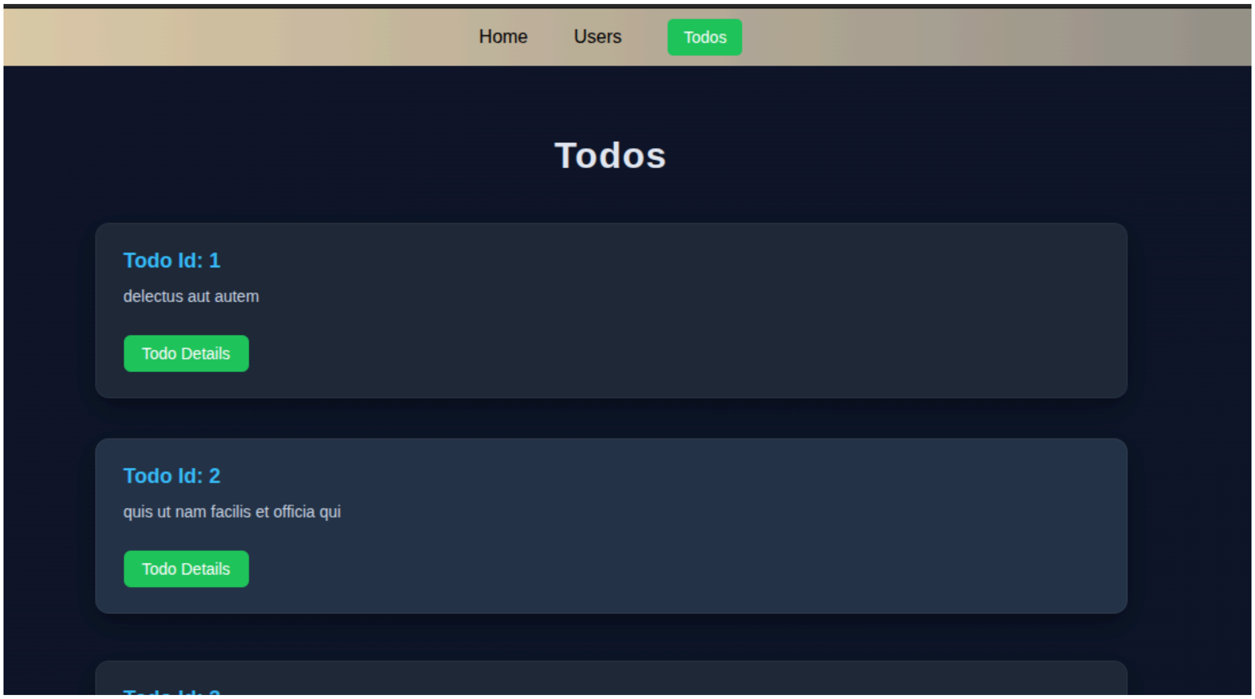
Q4.) Create a todos listing page using fetch data of todos using this api https://jsonplaceholder.typicode.com/todos

A4.) NextJs


Module4 > Exercise5 > todo-app > app > todos > page.jsx > Todos

```jsx
import Link from "next/link";
import "./page.css";

export default async function Todos(){

    const data = await fetch("https://jsonplaceholder.typicode.com/todos");
    const todos = await data.json();

    return(
        <div className="todos">
            <h2>Todos</h2>

            <ul>
                {todos.map(({id,title})=>(
                    <li key={id}>
                        <h3>Todo Id: {id}</h3>
                        <p className="title">{title}</p>
                        <Link href={`/todos/${id}`}>Todo Details</Link>
                    </li>
                ))}
            </ul>
        </div>
    );
}
```

Output



Todos

Todo Id: 1

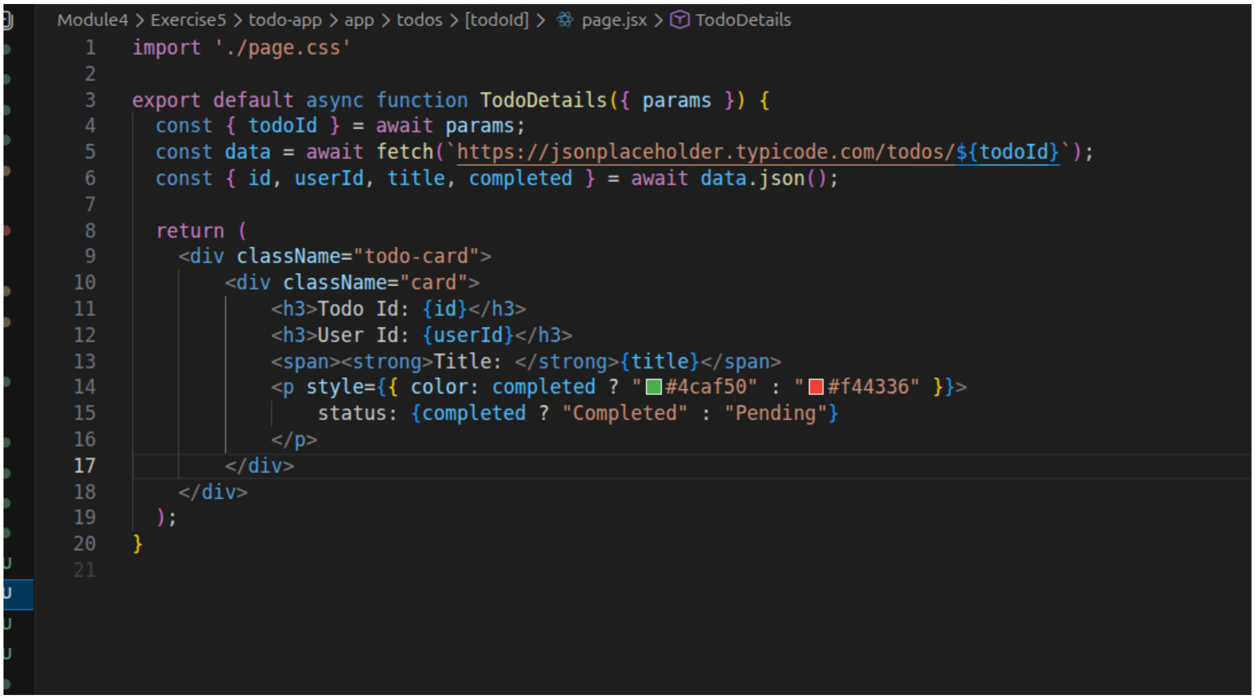
delectus aut autem

Todo Id: 2

quis ut nam facilis et officia qui

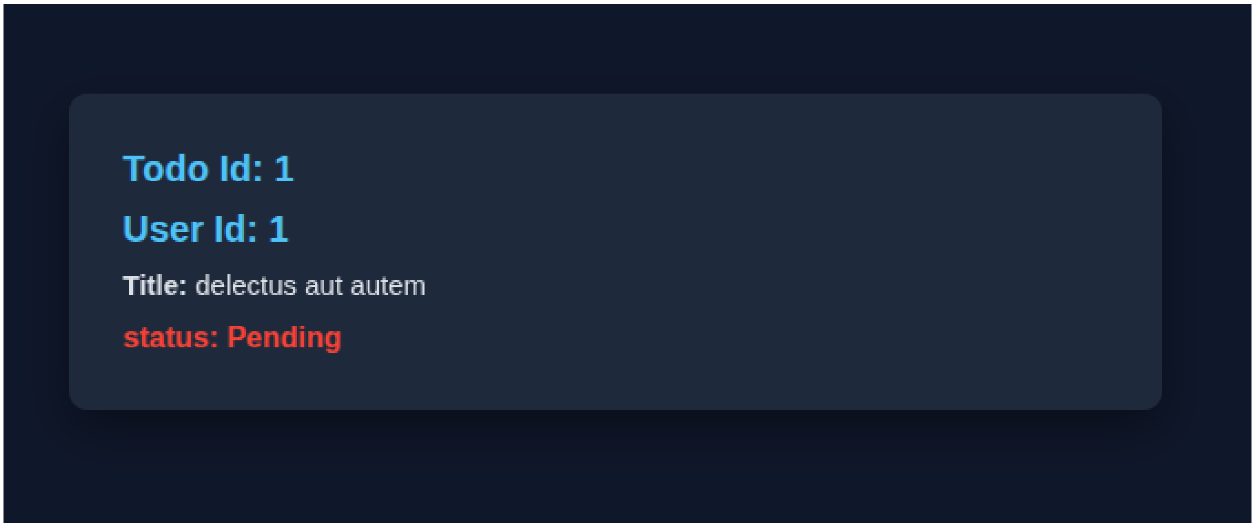
Q5.) Create a page which show individual todos details using this api https://jsonplaceholder.typicode.com/todos

A5.) NextJs


Module4 > Exercise5 > todo-app > app > todos > [todoId] > page.jsx > TodoDetails

```
import './page.css'

export default async function TodoDetails({ params }) {
  const { todoId } = await params;
  const data = await fetch(`https://jsonplaceholder.typicode.com/todos/${todoId}`);
  const { id, userId, title, completed } = await data.json();

  return (
    <div className="todo-card">
        <div className="card">
            <h3>Todo Id: {id}</h3>
            <h3>User Id: {userId}</h3>
            <span><strong>Title: </strong>{title}</span>
            <p style={{ color: completed ? "🟩#4caf50" : "🟥#f44336" }}>
                status: {completed ? "Completed" : "Pending"}
            </p>
        </div>
    </div>
  );
}
```

Output



Todo Id: 1

User Id: 1

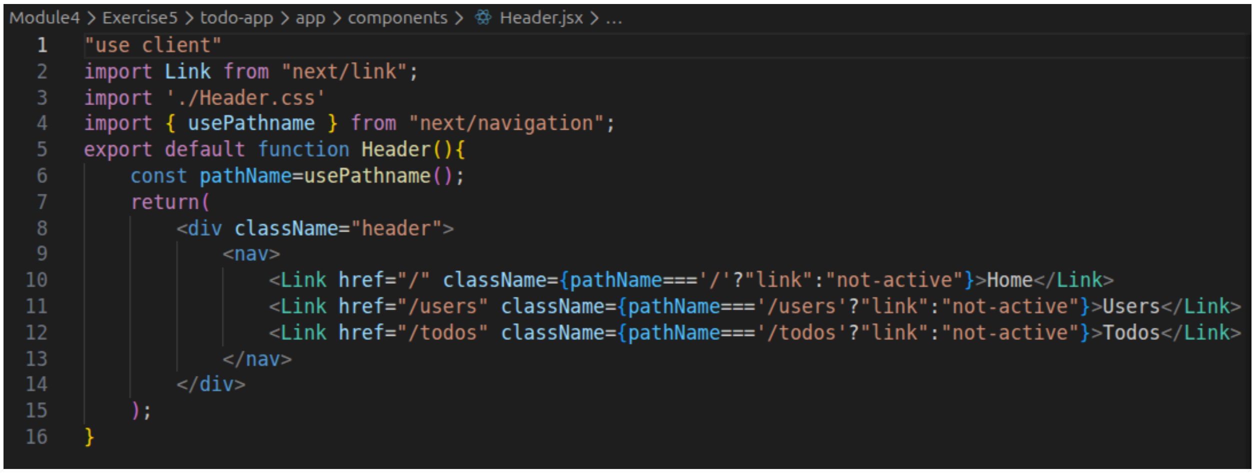
Title: delectus aut autem

status: Pending

Q6.) Create a common Header and Footer components and add these to all the pages.

A6.) NextJs

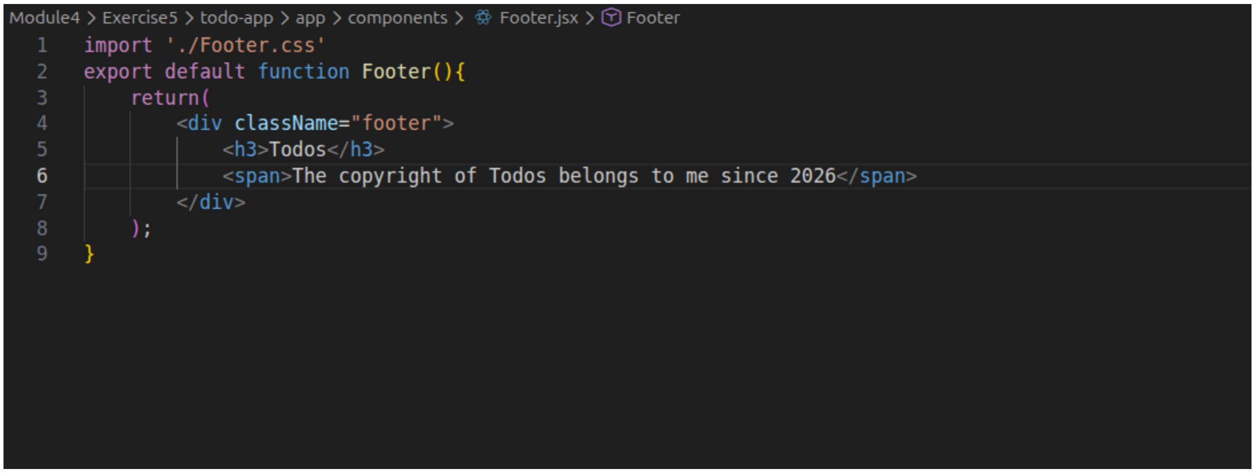
[Header.JS]



Module4 > Exercise5 > todo-app > app > components > Header.jsx > ...

```
"use client"
import Link from "next/link";
import './Header.css'
import { usePathname } from "next/navigation";
export default function Header(){
    const pathName=usePathname();
    return(
        <div className="header">
            <nav>
                <Link href="/" className={pathName==='/'?"link":"not-active"}>Home</Link>
                <Link href="/users" className={pathName==='/users'?"link":"not-active"}>Users</Link>
                <Link href="/todos" className={pathName==='/todos'?"link":"not-active"}>Todos</Link>
            </nav>
        </div>
    );
}
```

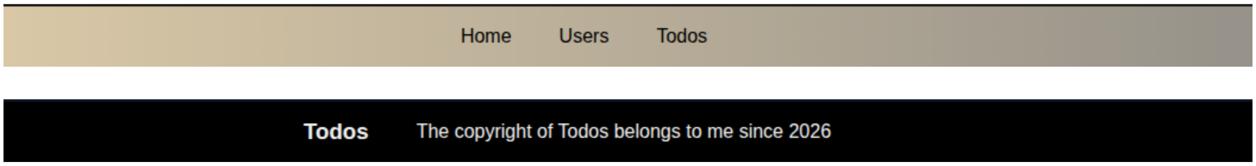
[Footer.JS]



Module4 > Exercise5 > todo-app > app > components > Footer.jsx > Footer

```
import './Footer.css'
export default function Footer(){
    return(
        <div className="footer">
            <h3>Todos</h3>
            <span>The copyright of Todos belongs to me since 2026</span>
        </div>
    );
}
```

Output

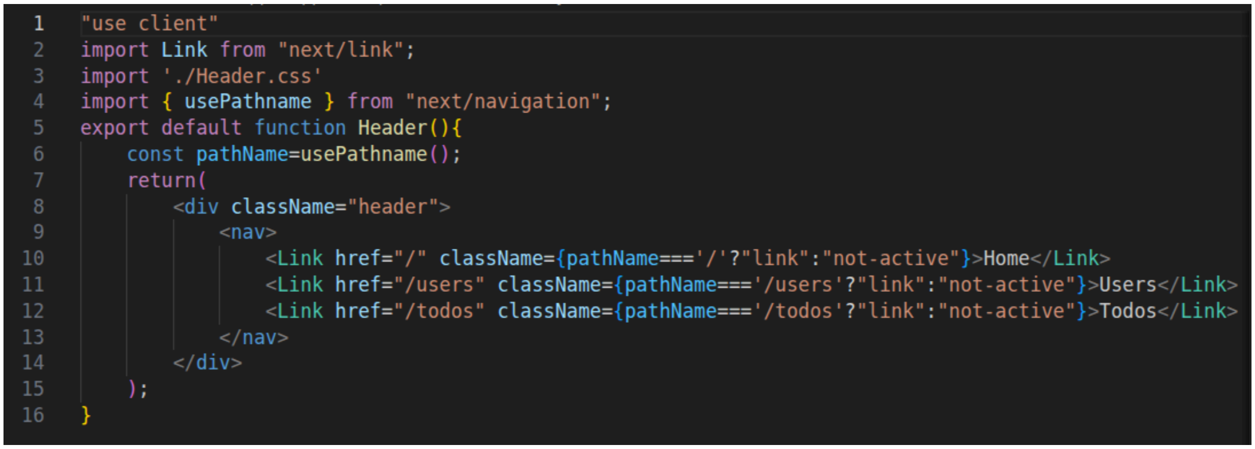


Home Users Todos

Todos The copyright of Todos belongs to me since 2026

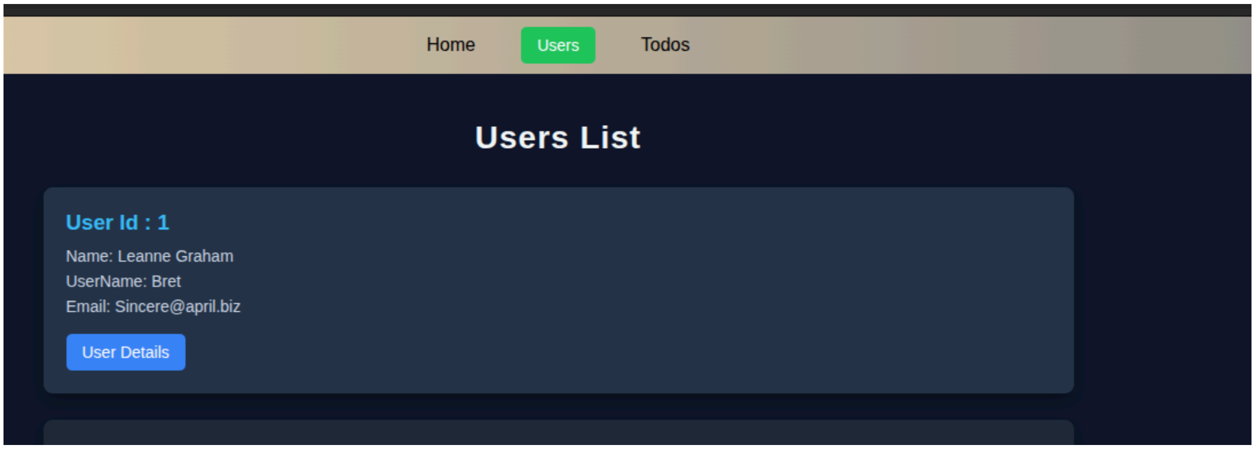
Q7.) usePathname to show active menu items.

A7.) NextJs



```
"use client"
import Link from "next/link";
import './Header.css'
import { usePathname } from "next/navigation";
export default function Header(){
    const pathName=usePathname();
    return(
        <div className="header">
            <nav>
                <Link href="/" className={pathName==='/'?"link":"not-active"}>Home</Link>
                <Link href="/users" className={pathName==='/users'?"link":"not-active"}>Users</Link>
                <Link href="/todos" className={pathName==='/todos'?"link":"not-active"}>Todos</Link>
            </nav>
        </div>
    );
}
```

Output



Users List

User Id : 1

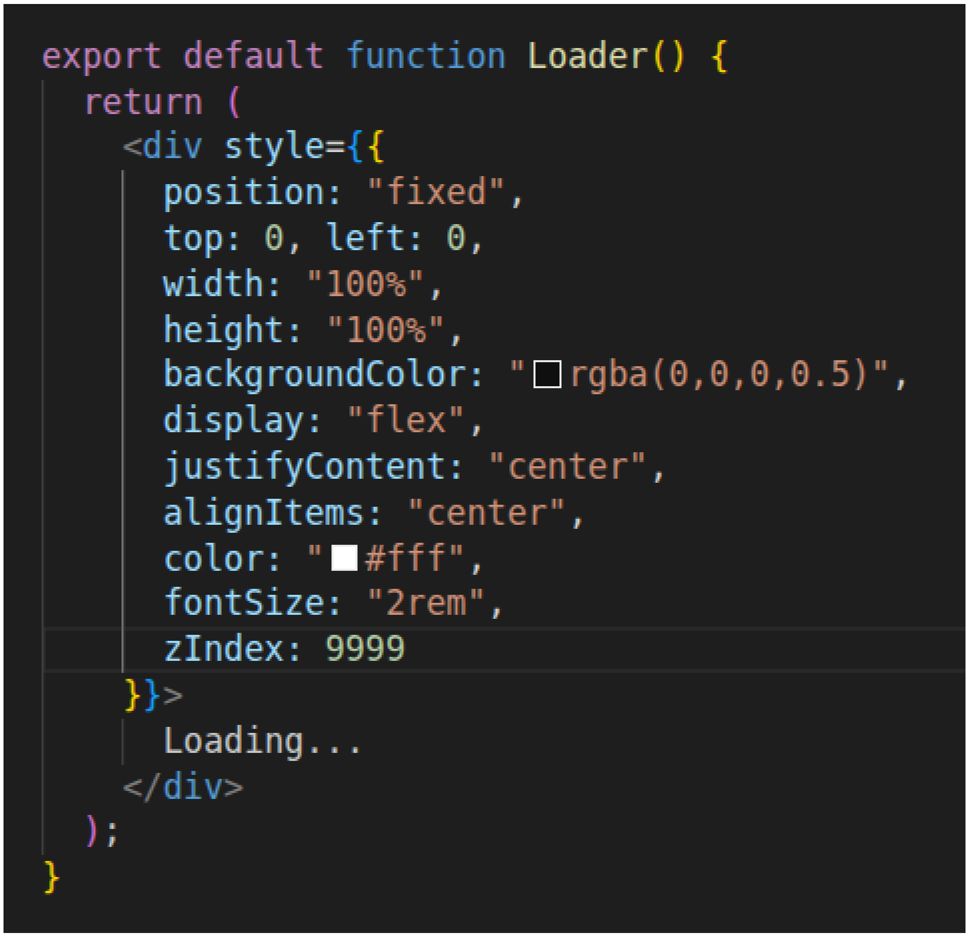
Name: Leanne Graham

UserName: Bret

Email: Sincere@april.biz

Q8.) Add loading UI using loading.js.

A8.) NextJs



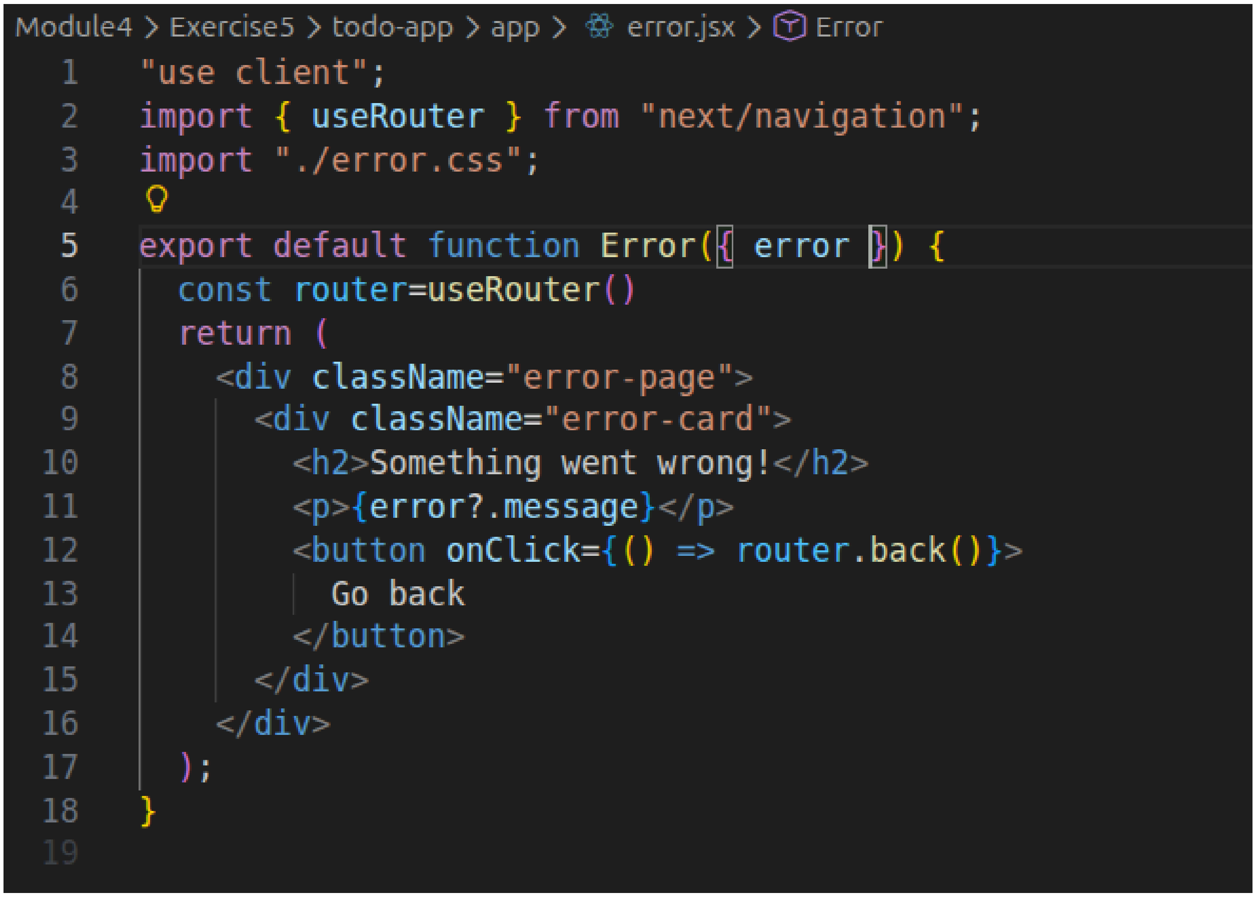
```
export default function Loader() {
  return (
    <div style={{
      position: "fixed",
      top: 0, left: 0,
      width: "100%",
      height: "100%",
      backgroundColor: "rgba(0,0,0,0.5)",
      display: "flex",
      justifyContent: "center",
      alignItems: "center",
      color: "#fff",
      fontSize: "2rem",
      zIndex: 9999
    }}>
      Loading...
    </div>
  );
}
```

Output

Loading...

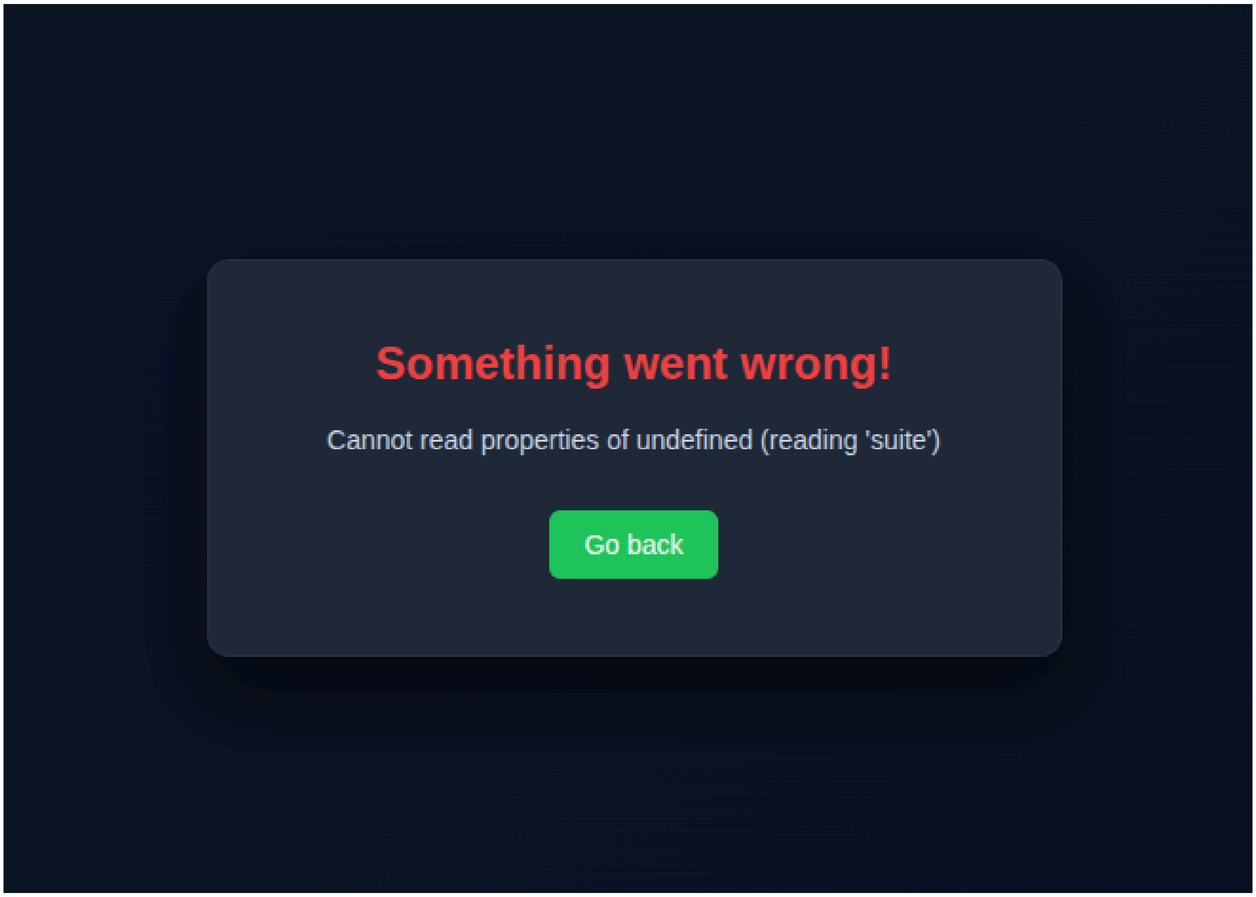
Q9.) Add error page using error.js

A9.) NextJs



```jsx
"use client";
import { useRouter } from "next/navigation";
import "./error.css";

export default function Error({ error }) {
  const router=useRouter()
  return (
    <div className="error-page">
      <div className="error-card">
        <h2>Something went wrong!</h2>
        <p>{error?.message}</p>
        <button onClick={() => router.back()}>
          Go back
        </button>
      </div>
    </div>
  );
}
```

Output

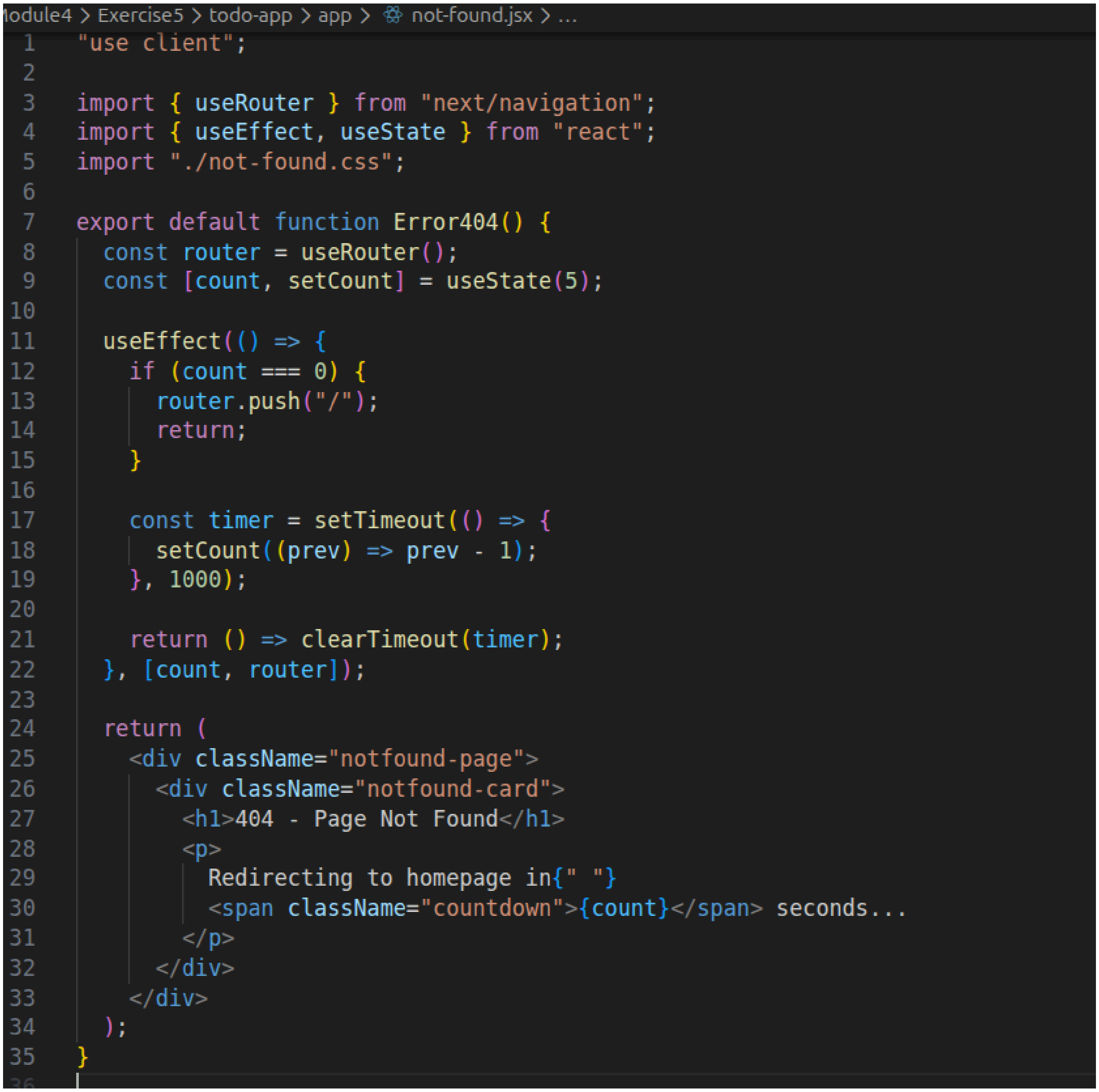


Something went wrong!

Cannot read properties of undefined (reading 'suite')

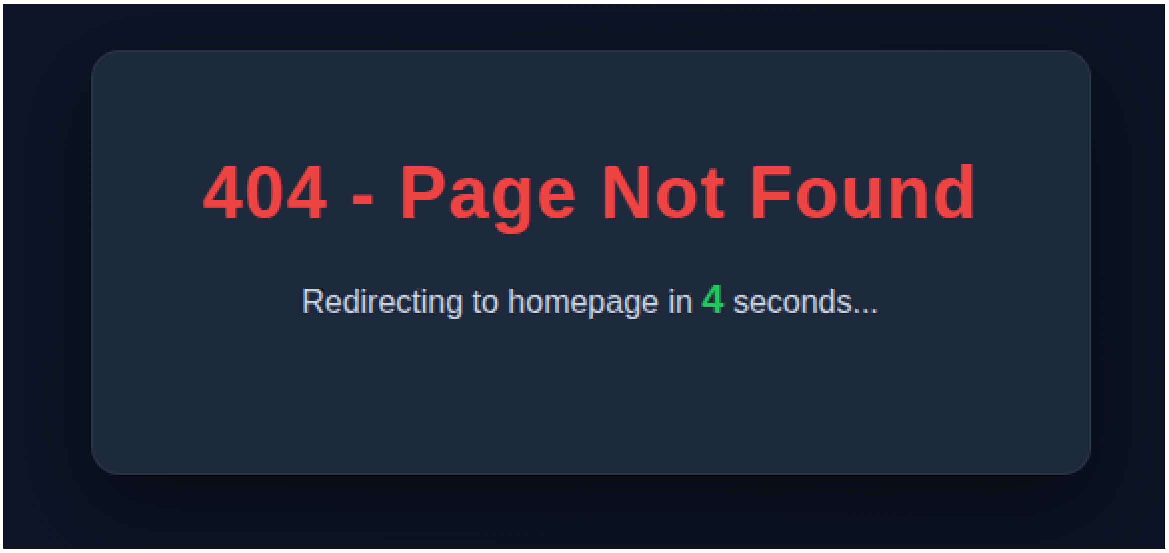
Q10.) Add not-found page if some route not exist inside the project.

A10.) NextJs



```jsx
"use client";

import { useRouter } from "next/navigation";
import { useEffect, useState } from "react";
import "./not-found.css";

export default function Error404() {
  const router = useRouter();
  const [count, setCount] = useState(5);

  useEffect(() => {
    if (count === 0) {
      router.push("/");
      return;
    }

    const timer = setTimeout(() => {
      setCount((prev) => prev - 1);
    }, 1000);

    return () => clearTimeout(timer);
  }, [count, router]);

  return (
    <div className="notfound-page">
      <div className="notfound-card">
        <h1>404 - Page Not Found</h1>
        <p>
          Redirecting to homepage in{" "}
          <span className="countdown">{count}</span> seconds...
        </p>
      </div>
    </div>
  );
}
```

Output



404 - Page Not Found

Redirecting to homepage in 4 seconds...