WEB APP DEVELOPMENT WITH REACTJS (INT252)

Lecture 14: URL Parameters & useParams Hook

Unit IV – Routing & Advanced Concepts

Syllabus Mapping:

- Dynamic routing
- URL parameters
- useParams hook

Target Learner: Beginner (HTML, CSS, JavaScript knowledge)

Why URL Parameters Are Needed

Static routes are limited:

```
/user
/user
/user
```

Real apps need:

```
/user/1
/user/2
/user/3
```

What Are URL Parameters?

URL parameters are:

- Dynamic parts of URL
- Used to identify data

Example:

```
/product/101
```

Dynamic Route Syntax

```
<Route path="/user/:id" element={<User />} />
```

`:id` → dynamic parameter

Visual Flow of URL Params

```
URL: /user/5
   ↓
:id = 5
   ↓
Component renders data for user 5
```

useParams Hook

`useParams` allows us to:

- Read URL parameters

Importing useParams

```
import { useParams } from 'react-router-dom';
```

Basic useParams Example

```
function User() {
  const { id } = useParams();

  return <h1>User ID: {id}</h1>;
}
```

Complete Routing Example

```
<Routes>
  <Route path="/user/:id" element={<User />} />
</Routes>
```

Navigating to Dynamic Routes

```
<Link to="/user/10">User 10</Link>
```

Real-World Use Cases

- User profiles
- Product details
- Blog posts
- Order pages

URL Params vs Props

URL Params	Props
From URL	From parent
Used in routing	Used in components

Common Beginner Mistakes

❌ Forgetting `:` in route

❌ Wrong param name

❌ Not importing useParams

Practice Exercises

1. Create `/product/:id` route
2. Display product id
3. Navigate using Link

✅ Answers – Practice Exercises

```
<Route path="/product/:id" element={<Product />} />
```

```
function Product() {
  const { id } = useParams();
  return <h2>Product {id}</h2>;
}
```

```
<Link to="/product/25">View Product</Link>
```

Key Takeaways

- URL params enable dynamic routing
- useParams reads URL data
- Used in real-world apps

Next Lecture

Lecture 15: Nested Routes & Layout Components

Unit IV – Routing & Advanced Concepts