NEXT.js

The React Framework for Production

Next.js Course By Matin Sangabi

The difference between React and Next.js

React :

- React library for building UI
- Needs to third-party libraries : RRD , authentication , styling , …

Next.js :

- A package that uses React for building user interface
- Enable you to building ready to production apps by features like routing , styling , authentication , bundle optimization etc.
- No need to install additional package

Why use Next.js ?

- SEO : fix SEO issues of javaScript Apps
- Better performance of Application

Why Learn Next.js ?

Next.js simplifies the process of building react application for production

1. File based Routing
2. Pre-rendering
3. Api Routes (internal Backend)
4. Support css modules
5. Authentication
6. Optimized build system

Install Next.js

```
npx create-next-app@latest
#or
yarn create next-app
#or
pnpm create next-app
```

If you want use typeScripts

```
npx create-next-app@latest –typescript
#or
yarn create next-app –typescript
#or
pnpm create-next-app --typescript
```

Routing

In React : React router dom => create components => export components   => <Routes element = {<…
/>}/>

In Next.js : in pages folder => files => create new file and finish .
- Routes with pages
- Nested pages : domain.com/about or domain.ir/login
- Dynamic pages : post/1 or post/2 , …
- Catch all Routes

So : in pages folder create index.js is main page or home page that's it and Run npm run dev
Se on localhost:3000 see this page

```
// index.js
Const Home = () =>{
    return <h1>Hello Next.js</h1>
}
export default Home;
```

If we want about page in page folder create about.js file and special components so in url =>
localhost:3000/about see this page
And other pages like this is created

Dynamics Routes

Create dynamic page like posts and post/1 or post/2 to post/n

Create folder as name you like in url for example posts :
Create folder posts and in posts directory create index.js so index is main page of posts
localhost:3000/posts see index and if we want see post/1 or post/2
create new file in posts directory like this [postId].js

```
#[postId].js

Import {useRouter} from "next/router";

const Post () => {
      const router = useRouter();
      const {postId} = router.query;
      return <h1>post page {postId}</h1> // out => ex : post/1 => post page -1 => ex : post/matin => out : post page – matin
}
Export default Post;
```

Nested dynamic Routing

Like localhost:3000/post/2/10

So : create [postId] folder in posts folder and create index.js so search localhost:3000/posts/1 see post if we want review localhost:3000/posts/review/1 => create review folder in [postId] folder and create index.js or [reviewId].js

Notice : if we want query set we need same like name of folder

Localhost:3000/posts/2?sort=last&page=1

In router.query = > {postId : 2 , sort : last , page : 1}

Catch all Routes

Create new folder as name blogs => blogs/2020/1/23 or blogs/a/b/c

Blog : if we want blog between year 2000 – 2020 and in any year 10 blogs so 2000 page of blogs => handle this => create year folder => [dayId.js] => 1*10
If we want new as month or any thing so we must create [dayId] folder and suppose many of this sorted or any thing we must create 1000 folder and this is not working and not good .

So : for fix this in next => [dayId] * [year] => [a] * [b] * [c] = 1 this is catch all routes
For better performance in blogs folder create [...params].js

```
Import {useRouter} from "next/router";
// blogs/2020/10/24
const Blog () => {
        const router = useRouter();
        console.log(router.query) // out => is array [2020 , 10  , 24] and we easiest send to backend
        return <h1>catch all routes</h1>}
Export default Blog;
```

If we search localhost:3000/blogs => error 404 not found for fix this use optional catch all routes => we must convert [...params].js to [[...params]].js => so this is fix

Link components

How to displaced in pages ? In next we use link componetns as work same like <a> tag

```
Import Link from `next/link`;

Cosnt Home = () =>{
      Return <Link href=`posts`>post page</Link>
}
Export default Home;

If we want button what we do ?

Import {useRouter} from 'next/router';
Cosnt Home = () =>{
     Const router = useRouter();
     Const clickHandler = () =>{
          router.push("/");
     }
     Return <button onClick={clickHandler}>post page</button>
} // but this one in js so absolutely we must run js on browser if js not load or any thing this not work
```

Not found page 404

In next.js as default build 404 page if we want style to 404 or personal 404 page we must in root of pages folder create 404.js and special components => now we can edit or any thing …