# File-system Routing
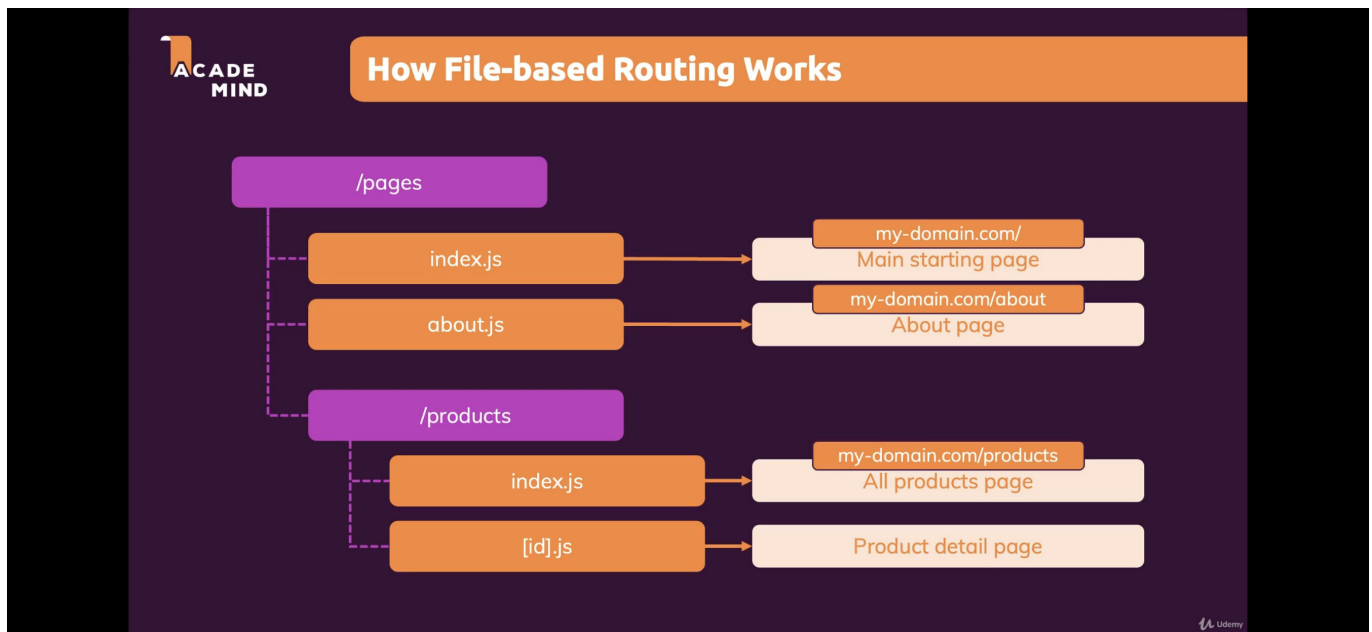
**AGENDA**

1. Overview
2. Dynamic routes
3. Route match order
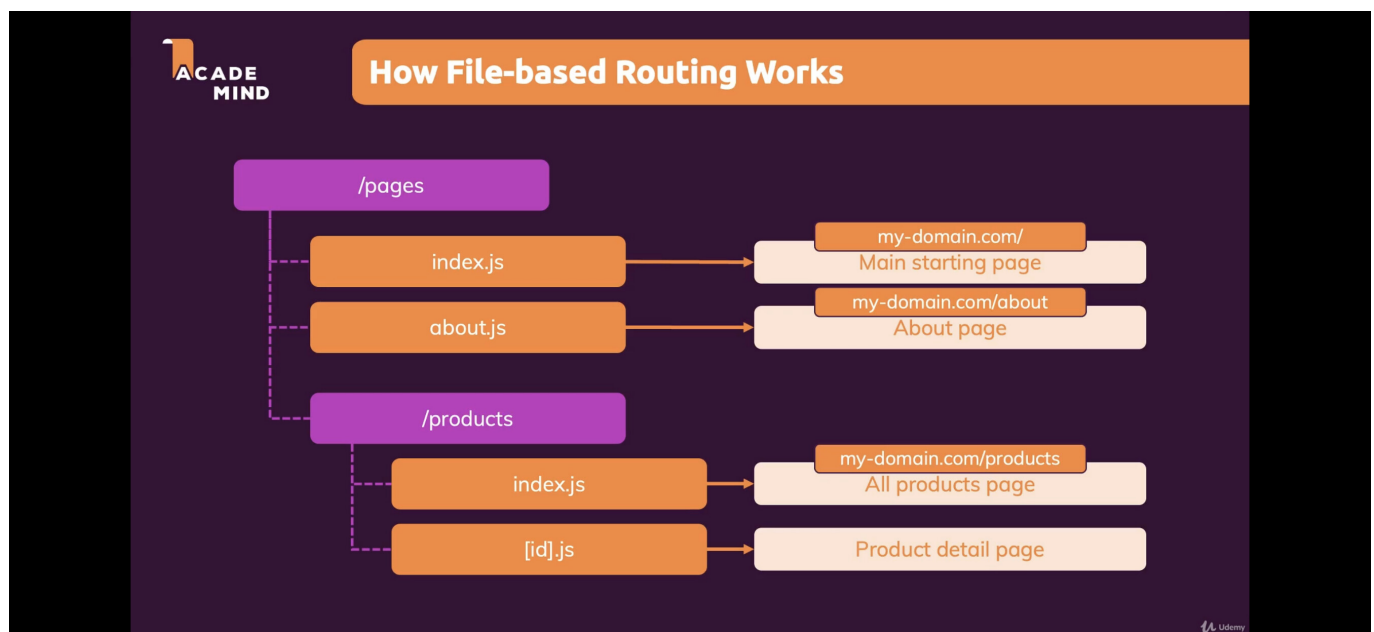
# 1. Overview

Add routings with **react-router-dom** in ReactJS

```
function App() {                    2 / 5
  // ...

  return (
    <Switch>
      <Route exact path="/" component={HomePage} />
      <Route path="/login" component={LoginPage} />
      <Route path="/products" component={ProductFeature} />
      <Route component={NotFound} />
    </Switch>
  )
}
```

But in **NextJS**, we use **File-system Routing**, what does it means?

- When a file is **added to the pages directory** it's automatically available **as a route**.
- There are three kind of routes
    - **Index routes**: file named index
    - **Nested routes**: nested folder in pages
    - **Dynamic routes**: brackets in file name to match parameter dynamically



*Source:* *https://forums.meteor.com/t/next-js-file-based-routing/55517*

--> With pre-defined routes, one page for one URL, but what if you need to match many URLs for only one page? Well get to the next section hehe 😅

## 2. Dynamic routes

We can use dynamic routes (bracket syntax) to support path parameters.

| # | Name | File Path |
|---|------|-----------|
| 1 | Single parameter | pages/posts/**[postId]**.tsx |
| 2 | Multiple parameters | pages/categories/**[categoryId]**/posts/**[postId]**.tsx |
| 3 | Catch all routes | pages/posts/**[...slug]**.tsx |
| 4 | Optional catch all routes | pages/posts/**[[...slug]]**.tsx |

> Note: You can use names other than `slug`, such as: `[...params]`

**Single parameter**: `pages/posts/[postId].tsx` will match:

| # | URL | router.query |
|---|-----|--------------|
| 1 | /posts/123 | `{ postId: '123' }` |
| 2 | /posts/developer-tips | `{ postId: 'developer-tips' }` |
| 3 | /posts/developer-roadmap | `{ postId: 'developer-roadmap' }` |

but it won't match **/posts/developer-roadmap/something-else**

**Multiple parameters**: `pages/categories/[categoryId]/posts/[postId].tsx` will match:

| # | URL | router.query |
|---|-----|--------------|
| 1 | /categories/123/posts/456 | `{ categoryId: '123', postId: '456' }` |
| 2 | /categories/frontend/posts/js | `{ categoryId: 'frontend', postId: 'js' }` |

```
pages
|__ categories
|   |__ [categoryId] (folder)
|       |__ posts
|           |__ [postId].tsx  (file)
```

**Catch all routes**: `pages/posts/[...slug].tsx` will match:

| # | URL | router.query |
|---|-----|--------------|
| 1 | /posts/123 | `{ slug: ['123'] }` |
| 2 | /posts/easy-nextjs | `{ slug: ['easy-nextjs'] }` |
| 3 | /posts/easy/frontend | `{ slug: ['easy', 'frontend'] }` |
| 4 | /posts/easy/frontend/nextjs | `{ slug: ['easy', 'frontend', 'nextjs'] }` |

but it won't match **/posts**

**Optional catch all routes**: `pages/posts/[[...slug]].tsx` will match:

Same to **Catch all routes** but it also match **/posts**

## 3. Route match order

| Priority | Name | File Path | Code |
|----------|------|-----------|------|
| 1 | Pre-defined routes | pages/posts/**create**.tsx | A |
| 2 | Dynamic routes | pages/posts/**[postId]**.tsx | B |
| 3 | Catch all routes | pages/posts/**[...slug]**.tsx | C |

Examples:

| # | if user visit | it will match |
|---|---------------|---------------|
| 1 | /posts/create | A |
| 2 | /posts/12345 | B |
| 3 | /posts/developer-tips | B |
| 4 | /posts/frontend-road-map | B |
| 5 | /posts/12345/comments | C |
| 6 | /posts/12345/comments/123 | C |

# Tham khảo

- https://nextjs.org/docs/routing/introduction
- https://nextjs.org/docs/routing/dynamic-routes
- https://www.smashingmagazine.com/2020/10/getting-started-with-next-js/

---

**Series - NextJS + Typescript** 🎉

- Tác giả: **Hậu Nguyễn**
- Được phát hành trên kênh youtube **Easy Frontend**.
- Tài liệu pdf và videos đều có bản quyền thuộc về Easy Frontend.
- Videos được phát hành cho fan cứng trước, public sau.
- Đăng ký fan cứng để xem series này đầy đủ và sớm nhất nhé.

☎️ Kết nối với mình

- ✅ Follow Facebook: https://www.facebook.com/nvhauesmn/
- ✅ Like Fanpage: https://www.facebook.com/learn.easyfrontend
- ✅ Youtube Channel: https://www.youtube.com/easyfrontend