

Introduction

to

Next.js



What is Next.js?

What is Next.js?

- ❑ React framework that gives you building blocks to create fast web applications

What is Next.js?

- ❑ React framework that gives you building blocks to create fast web applications
- ❑ Handles the tooling and configuration needed for React, and provides additional structure, features and optimizations for your application

What is Next.js?

- ❑ React framework that gives you building blocks to create fast web applications
- ❑ Handles the tooling and configuration needed for React, and provides additional structure, features and optimizations for your application
- ❑ Can be used to solve common web application requirements such as
 - Routing
 - Data fetching
 - Integrations

What is rendering?

What is rendering?

- Work done to convert React code to HTML representation of UI

What is rendering?

- Work done to convert React code to HTML representation of UI

Server

Client



What is rendering?

- Work done to convert React code to HTML representation of UI

Server

Client



- Can happen at build-time and at runtime

Rendering types

Rendering types

Pre-rendering

- ☐ Server-Side Rendering
- ☐ Static Site Generation

Rendering types

Pre-rendering

- ☐ Server-Side Rendering
- ☐ Static Site Generation

Client Side Rendering

- ☐ Rendering work done by client

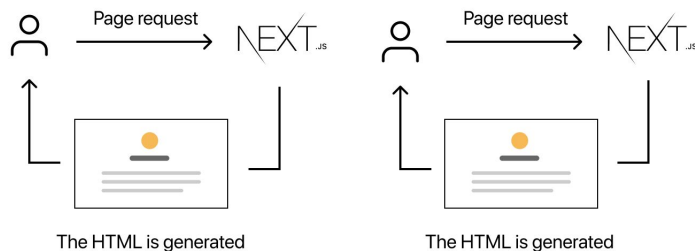
Rendering types

Pre-rendering

- ❑ Server-Side Rendering
- ❑ Static Site Generation

Server-side Rendering

The HTML is generated on **each request**.

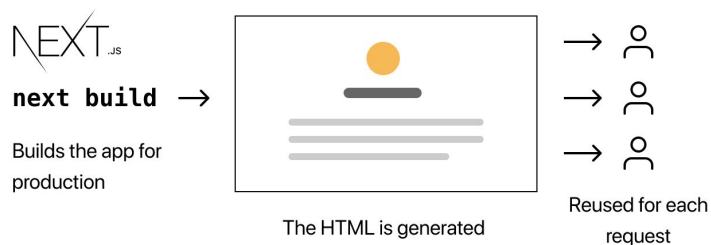


Client Side Rendering

- ❑ Rendering work done by client

Static Generation

The HTML is generated at **build-time** and is reused for each request.



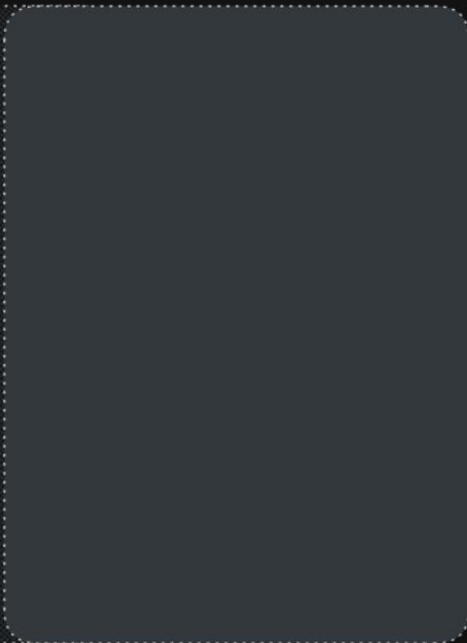
Client-side Rendering (CSR)

Browser receives empty HTML shell from the server and JS instructions to construct the UI

Initial HTML

```
<html>
  <body>
    <div></div>
  </body>
</html>
```

Initial UI



Client-side Render

Rendered UI

Team

- A. Lovelace
- G. Hopper
- M. Hamilton

Like (0)

Pre-rendering

Next.js Pre-renders
every page by default

HTML is generated
in advance on a server

Initial HTML

```
<html>
  <body>
    <div>
      <h1>Team</h1>
      <ul>
        <li>A. Lovelace
        <li>G. Hopper
        <li>M. Hamilton
      </ul>
      <button>Like
    </div>
  </body>
</html>
```

Initial UI

Team

- A. Lovelace
- G. Hopper
- M. Hamilton

Like (0)

Hydration

Rendered UI

Team

- A. Lovelace
- G. Hopper
- M. Hamilton

Like (0)

Using Server-side rendering

Using Server-side rendering

```
import React from 'react'

const BlogsPage = ({ blogs }) => {
  // Do what you want with blogs.
  return (
    <div></div>
  )
}

export const getServerSideProps = async () => {
  const res = await fetch('...')
  const blogs = await res.json()

  return { props: { blogs: blogs } }
}

export default BlogsPage
```

Using Server-side rendering

```
export const getServerSideProps = async () => {  
  
  // Mongoose way  
  await dbConnect()  
  const result = await Blogs.find({})  
  const blogs = result.map((doc) => {  
    const blog = doc.toObject()  
    blog._id = blog._id.toString()  
    return blog  
  })  
  
  return { props: { blogs: blogs } }  
}
```

```
export const getServerSideProps = async () => {  
  
  // Prisma way  
  const prisma = new PrismaClient()  
  const blogs = await prisma.blogs.findMany()  
  
  return { props: { blogs: blogs } }  
}
```

```
export const getServerSideProps = async () => {  
  const res = await fetch('...')  
  const blogs = await res.json()  
  
  return { props: { blogs: blogs } }  
}
```

Using pages

Using pages

/pages

blog.tsx → /blog

Using pages

/pages

blog.tsx → /blog

/pages/blog

index.tsx → /blog

about.tsx → /blog/about

Using pages

/pages

`blog.tsx` → `/blog`

/pages/posts

`[id].tsx` → `/posts/1`
`/posts/2`
`/posts/...`

/pages/blog

`index.tsx` → `/blog`

`about.tsx` → `/blog/about`

Project structure

- ▼ NEXTJS-INTRODUCTION
 - > .next
 - > node_modules
 - ▼ pages
 - ▼ api
 - TS hello.ts
 - TS _app.tsx
 - TS _document.tsx
 - TS index.tsx
 - ▼ public
 - ★ favicon.ico
 - next.svg
 - thirteen.svg
 - vercel.svg
 - ▼ styles
 - # globals.css
 - # Home.module.css
 - ◆ .gitignore
 - TS next-env.d.ts
 - JS next.config.js
 - { } package-lock.json
 - { } package.json
 - ① README.md
 - TS tsconfig.json

Project structure

□ /pages → file based routing for pages

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      |   TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    🗂 next.svg
    🗂 thirteen.svg
    🗂 vercel.svg
  ▼ styles
    |   # globals.css
    |   # Home.module.css
  📄 .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```


Project structure

- /pages → file based routing for pages
- /public → assets

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      |   TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    📁 next.svg
    📁 thirteen.svg
    📁 vercel.svg
  ▼ styles
    # globals.css
    # Home.module.css
  💎 .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```

Project structure

- /pages → file based routing for pages
- /public → assets
- /styles → css styles

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    next.svg
    thirteen.svg
    vercel.svg
  ▼ styles
    # globals.css
    # Home.module.css
  .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```

Project structure

- /pages → file based routing for pages
- /public → assets
- /styles → css styles
- root level:
npm package files,
configuration files

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    next.svg
    thirteen.svg
    vercel.svg
  ▼ styles
    # globals.css
    # Home.module.css
  .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```

Project structure

Question

If you wanted to create a new route
/authors/me, what would the file name be
(including the directory it has to be in?)

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      |   TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    📁 next.svg
    📁 thirteen.svg
    📁 vercel.svg
  ▼ styles
    # globals.css
    # Home.module.css
  .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```

Project structure

Question

If you wanted to create a new route /authors/me, what would the file name be (including the directory it has to be in?)

- ☐ authors/me.tsx
- ☐ pages/authors/me.tsx
- ☐ routes/authors/me.tsx

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    next.svg
    thirteen.svg
    vercel.svg
  ▼ styles
    # globals.css
    # Home.module.css
  .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```

Project structure

Question

If you wanted to create a new route /authors/me, what would the file name be (including the directory it has to be in?)

- ☐ authors/me.tsx
- ☒ pages/authors/me.tsx
- ☐ routes/authors/me.tsx

```
▼ NEXTJS-INTRODUCTION
  > .next
  > node_modules
  ▼ pages
    ▼ api
      |   TS hello.ts
    TS _app.tsx
    TS _document.tsx
    TS index.tsx
  ▼ public
    ★ favicon.ico
    🖼 next.svg
    🖼 thirteen.svg
    🖼 vercel.svg
  ▼ styles
    # globals.css
    # Home.module.css
  💎 .gitignore
  TS next-env.d.ts
  JS next.config.js
  {} package-lock.json
  {} package.json
  ⓘ README.md
  TS tsconfig.json
```

Other features

MDX

AMP Support

Turbopack

Dynamic Imports

Image/Font optimizations

Static File Serving

TypeScript

API Routes

Custom Server

Next.js 13

- ❑ app directory (beta)
 - ❑ layouts
 - ❑ React Server Components
 - ❑ Streaming (progressively render)

Turbopack

improved next/image

new font system

improved next/link