

Documentation

- **Getting Started**

- ▼ Basic Features

- Pages
- Data Fetching
- Built-in CSS Support
- Image Optimization
- Static File Serving
- Fast Refresh
- TypeScript
- Environment Variables
- Supported Browsers and Features

- › Routing

- › API Routes

- Deployment

- Authentication

Getting Started

Welcome to the Next.js documentation!

If you're new to Next.js we recommend that you start with the [learn course](#).

The interactive course with quizzes will guide you through everything you need to know to use Next.js.

If you have questions about anything related to Next.js, you're always welcome to ask our community on [GitHub Discussions](#).

System Requirements

- [Node.js 10.13](#) or later
- MacOS, Windows (including WSL), and Linux are supported

Setup

We recommend creating a new Next.js app using `create-next-app`, which sets up everything automatically for you. To create a project, run:

```
npx create-next-app
# or
yarn create next-app
```

After the installation is complete, follow the instructions to start the development server. Try editing `pages/index.js` and see the result on your browser.

For more information on how to use `create-next-app`, you can review the [create-next-app](#) documentation

Manual Setup

Install `next`, `react` and `react-dom` in your project:

```
npm install next react react-dom
# or
yarn add next react react-dom
```

Open `package.json` and add the following `scripts`:

```
"scripts": {
  "dev": "next dev",
  "build": "next build",
  "start": "next start"
}
```

These scripts refer to the different stages of developing an application:

- `dev` - Runs `next dev` which starts Next.js in development mode
- `build` - Runs `next build` which builds the application for production usage
- `start` - Runs `next start` which starts a Next.js production server

Next.js is built around the concept of pages. A page is a [React Component](#) exported from a `.js`, `.jsx`, `.ts`, or `.tsx` file in the `pages` directory.

Pages are associated with a route based on their file name. For example ``pages/about.js`` is mapped to ``/about``. You can even add dynamic route parameters with the filename.

Create a ``pages`` directory inside your project.

Populate ``./pages/index.js`` with the following contents:

```
function HomePage() {  
  return <div>Welcome to Next.js!</div>  
}  
  
export default HomePage
```

To start developing your application run ``npm run dev`` or ``yarn dev``. This starts the development server on ``http://localhost:3000``.

Visit ``http://localhost:3000`` to view your application.

So far, we get:

- Automatic compilation and bundling (with webpack and babel)
- React Fast Refresh
- Static generation and server-side rendering of ``./pages/``
- Static file serving. ``./public/`` is mapped to ``/``

In addition, any Next.js application is ready for production from the start, read more in our Deployment documentation.

Related

For more information on what to do next, we recommend the following sections:



Pages:

Learn more about what pages are in Next.js.

CSS Support:

Use the built-in CSS support to add custom styles to your app.

CLI:

Learn more about the Next.js CLI.

[Basic Features: Pages](#) >

Was this helpful?



[Edit this page on GitHub](#)