Most Asked Next.js Interview Questions and Answers

Following is a list of most frequently asked Next.js interview questions and their answers.

1) What is Next.js?

Next.js is an open-source, lightweight React.js framework that facilitates developers to build static and server-side rendering web applications. It was created by Zeit. Next.js framework is based on React, Webpack, and Babel and allows us to write server-rendered React apps easily. It doesn't require any webpack configuration and only needs npm run dev start building your next feature-filled web application.

2) Why is Next.js used for? / Why do world's leading companies prefer Next.js?

If you want to build a complete web application with React from scratch, you have to fulfill the following points:

- Your code has to be bundled using a bundler like webpack and transformed using a compiler like Babel.
- You have to do production optimizations such as code splitting.
- You have to pre-render some pages for performance and SEO statically. You might also want to use server-side rendering or client-side rendering.
- You might have to write some server-side code to connect your React app to your data store.

Next.js fulfills the above all requirements.

[redacted]

[redacted]

Reasons w

Zero Setup hot code reloading, and universal rendering; that's why the world's leading companies prefer it.

Fully Extensible: Next.js is fully extensible and has complete control over Babel and Webpack. It provides a customizable server, routing, and next plugins.

Ready for Production: Next.js is optimized for smaller build sizes, faster dev compilation, and many other improvements, making it a popular choice.

Next.js can Deploy Anywhere: Next.js is an open-source, lightweight React.js framework that facilitates developers to build static and server-side rendering web applications.

3) What is the process to install Next.js? / How to install Next.js?

Before installing Next.js, you must have installed Node.js on your system. Learn how to install Node.js on your system: https://www.javatpoint.com/install-nodejs

It requires NPM to start installing Next.js with all its dependencies.

Follow the steps given below to install Next.js:

- First, create a directory to keep the Next.js project and go into it:

```
mkdir my-portfolio-site
cd my-portfolio-site
```

- Now, initialize this with a package.json file.
- Use the y flag by npm init -y
- Use the following syntax to install Next.js

```
npm install react react-dom next
```

- Update package.json with run script languages to start the initialization of the Next.js application.
- Now, find the package.json file on the root folder and add the below mentioned script

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

- After that, the filesystem is the main API. Every ".js" file becomes a route that gets automatically processed and rendered.
- Now, the process is completed, and Next.js is installed on your system.

4) What are the most prominent features of Next.js?

Following is a list of the most prominent features of Next.js that excite the developers most:

- js provides the by default and easy server rendering.
- js supports static exporting.
- It provides a Webpack-based dev environment which supports Hot Module Replacement (HMR)
- It seaports automatic code-splitting for faster page loads.
- It supports simple client-side routing (page-based) or file system-based routing.
- It provides complete Webpack and Babel control.
- It provides a faster and optimized development compilation.
- It can be implemented with Express or any other Node.js HTTP server.
- You can easily customize it with your own Babel and Webpack configurations.
- It supports hot code reloading.

Besides this, Next.js also has some awesome features such as:

- Dynamic styles and themes support
- Built-in CSS vendor prefixing
- CSS Preprocessing via Plugins
- Full CSS support, no tradeoffs in power
- Its runtime size is very small. It is just 3kb (zipped from 12kb)
- It provides source maps support

5) Which types of websites most popularly use Next.js?

Next.js is a popular framework of React.js that is most popularly used for building the following types of apps and websites:

- Static Websites
- Desktop Websites
- SEO Friendly Websites
- Server Rendered Apps
- Progressive Web Apps (PWA) etc.

6) Is it possible to use Next.js with Redux?

Yes. You can easily use Next.js with Redux.

7) What is the recommended method to fetch data in Next.js?

There are multiple ways to fetch data in Next.js, but Next.js itself recommends **getInitialProps,** an async function to retrieve data from anywhere. When we use **getInitialProps** to retrieve data, it receives a context object which has the following properties:

- **pathname-** It specifies the path section of the URL.
- **query-** It is used to specify the query string section of URL parsed as an object.
- **asPath-** It specifies the string of the actual path (including the query) shows in the browser.
- **req-** It is used to specify the HTTP request object (server only).
- **res-** It is used to specify the HTTP response object (server only).
- **err-** It is used to specify the error object if any error is encountered during the rendering.

8) Give an example to demonstrate how do you set up CDN in Next.js?

To setup CDN in Next.js, the developers have to follow the steps given below:

To start, we have to first set up the "assetPrefix" setting and configure our CDN origin to support and resolve the domain that our Next.js is hosted on.

```
const isProd = process.env.NODE_ENV === 'production';
 module.exports = {
// You may only need to add assetPrefix in the production.
 assetPrefix: isProd ? 'https://cdn.mydomain.com' : ''
};
```

If the CDN is present on a separate domain, we have to set a configuration option as following:

```
// next.config.js
odule.exports = {
crossOrigin: 'anonymous'
;
```

9) Are Create-React-App and Next.js used for the same thing?

The Create-React-App is basically React with an integrated build system. It acts like a good boilerplate, so we don't need to set up Webpack, Babel, and other dependent packages to run React. Other than that, if you require extra functionalities such as routing, server-side rendering, and so on, you just need to add packages on top of Create-React-App. On the other hand, The Next.js is an open-source, lightweight full-stack React framework that comes bundled with an efficient build system, server-side rendering, routing, API routing, and many other awesome features that make the production environment easy.

10) How can you install and use Next.js?

There are mainly two ways to install and run Next.js on your system. If you're new to Next.js, we recommend that you make sure that your development environment is ready. Next.js is a React framework, and it requires Node.js to be installed on your system. If you don't have Node.js installed, you can install it from here: https://www.javatpoint.com/install-nodejs. Next.js requires Node.js version 10.13 or later. You should also have your text editor and terminal.

System Requirement for Next.js

- You must have installed Node.js version 12.0 or later.
- js supports Windows, MacOS, and Linux also.

The simplest way to install Next.js

- The simplest way to install and run Next.js on your system is by creating a new Next.js app using create-next-app. It sets up everything automatically for you. To create a project, run the following command on the Node.js console.

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

- If you want to start with a TypeScript project, you can use the following command:

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

- After completing the installation, follow the instructions to start the development server. You can also try to edit the pages/index.js and see the result on your browser, which by default runs on http://localhost:3000

Manual Installation and Setup of Next.js

- First, install the next react and react-dom in your project by running the following command.

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

- Now, open the package.json file and add the following scripts:

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

The above script specifies the different stages of developing an application:

- **dev -** It runs the next dev, which starts Next.js in development mode.
- **build -** It runs "next build", which builds the application for production usage.
- **start -** It runs "next start", which starts a Next.js production server.
- **lint -** It runs "next lint", which sets up Next.js' built-in ESLint configuration.

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 filename. For example, pages/about.js is mapped to /about. You can even add dynamic route parameters with the filename.

11) How can you disable the etag generation in Next.js?

Generally, we use the app.disable('etag') syntax to disable the etag generation in Next.js. But, this may not work for all static contents. So, we should use the following syntax to disable the etag for all static contents.

Syntax:

```
app.use(express.static(path.join(__dirname, 'public'), {
etag: false
}));
```

12) How can you configure the build-id in Next.js?

To configure the build-id in Next.js, we must configure a static ID between our builds. So, we have to provide the "generateBuildId" function with the following configuration.

Syntax:

```
// next.config.js
module.exports = {
  generateBuildId: async () => {
 // For example get the latest git commit hash here
 return 'my-build-id';
 }
};
```

13) How can you create a page directory inside your project?

To create a page directory inside our project we have to populate the ./pages/index.js with the following contents:

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

To start developing our application, we have to run the npm run dev or yarn dev command. This will start the development server on http://localhost:3000. Now we can visit the localhost: http://localhost:3000 to view our application.

14) Give an example to demonstrate how to create a custom error page in Next.js?

We can create our custom error page by defining a _error.js in the pages folder. See the following example:

```
import React from "react";
class Error extends React.Component {
  static getInitialProps({ res, err }) {
    const statusCode = res ? res.statusCode : err ? err.statusCode : null;
    return { statusCode };
  }
  render() {
    return (

        {this.props.statusCode
          ? `An error ${this.props.statusCode} has occurred on the server`
          : "An error occurred on client-side"}

    );
  }
}
export default Error;
```

15) What do you understand by code splitting in Next.js?

Generally, code splitting is one of the most compelling features of webpack. This feature facilitates us to split our code into various bundles, which can be loaded only on-demand or in parallel. This is mainly used to achieve the smaller bundles and facilitates us to control resource load prioritization which finally has a great impact on the load time.

There are mainly three approaches to code splitting:

- **Entry Points:** It is used to split code using entry configuration manually.
- **Prevent Duplication:** It uses Entry dependencies or SplitChunksPlugin to dedupe and split chunks.
- **Dynamic Imports:** It splits the code via inline function calls within modules

It is mainly used to enable pages that can never load unnecessary code.

16) How can you enable AMP in Next.js?

This is an important question and is asked in many Next.js interview questions. There are two ways to enable AMP in Next.js.

- AMP-First Pages
- Hybrid AMP Pages

AMP-First Pages: The AMP-First Pages are served to the primary traffic of the website as well as traffic generated from the search engine. Use the following syntax to implement AMP-first pages.

Example:

```
// pages/index.js
import { withAmp } from 'next/amp'
function HomePage() {
return <p> Welcome to AMP + Next.js.</p>
}
export default withAmp(HomePage)
```

Hybrid AMP Pages: The Hybrid AMP pages allow the users to have coexisted AMP version of a traditional page so that the search engines can display the AMP version or the page in different mobile search results.

See the following example to understand how to implement the Hybrid AMP to pages:

Example:

```
// pages/index.js
import { withAmp } from 'next/amp'
function HomePage() {
return <p> Welcome to AMP + Next.js.</p>
}
export default withAmp(HomePage)
```

17) Is it possible to host Next.js in a web server like Nginx?

Next.js is not as simple as static html files. It requires an application server that runs Node.js to deploy and run a Next.js application. Here, we get requests that have to be processed on the server.

You may also like:

- Java Interview Questions
- SQL Interview Questions
- Python Interview Questions
- JavaScript Interview Questions
- Angular Interview Questions
- Selenium Interview Questions
- Spring Boot Interview Questions
- HR Interview Questions
- C Programming Interview Questions
- C++ Interview Questions
- Data Structure Interview Questions
- DBMS Interview Questions
- HTML Interview Questions
- IAS Interview Questions