

Next.js Interview Notes & Q&A;

1. What is Next.js?

Next.js is a React-based framework that enables hybrid static & server rendering, file-system based routing, API routes, and built-in optimizations like image optimization.

2. Difference between Next.js and React?

React is a library for building UI components, while Next.js is a framework built on React that adds server-side rendering (SSR), static site generation (SSG), API routes, routing, and more.

3. What are the rendering modes in Next.js?

1. SSR (Server-Side Rendering) 2. SSG (Static Site Generation) 3. ISR (Incremental Static Regeneration) 4. CSR (Client-Side Rendering).

4. Explain `getStaticProps`.

`getStaticProps` is used for Static Site Generation. It runs at build time and pre-renders pages with fetched data.

5. Explain `getServerSideProps`.

`getServerSideProps` is used for Server-Side Rendering. It runs on every request and fetches fresh data before rendering.

6. Explain `getStaticPaths`.

Used with `getStaticProps` for dynamic routes. Defines a list of paths that should be statically generated.

7. What is ISR (Incremental Static Regeneration)?

ISR allows you to update static pages after the site has been built. You can revalidate pages at a set interval.

8. What are API Routes in Next.js?

API Routes let you create RESTful endpoints inside the ``pages/api`` folder. Example: ``pages/api/hello.js``.

9. Example of an API route?

```
export default function handler(req, res) { res.status(200).json({ message: 'Hello Next.js!' }) }
```

10. What is Middleware in Next.js?

Middleware allows you to run code before a request is completed. Example use cases: authentication, redirects, logging.

11. Explain Next.js Routing.

Next.js uses file-based routing. Any file in the `pages` directory becomes a route automatically.

12. How to create dynamic routes?

Use brackets []. Example: `pages/blog/[id].js` handles routes like `/blog/1`, `/blog/2` etc.

13. How does Next.js handle images?

Next.js provides the `next/image` component which optimizes images automatically for better performance.

14. What are Server Components in Next.js 13?

Server Components run on the server, reduce bundle size, and improve performance by avoiding client-side JS.

15. Difference between pages router and app router?

Pages router (Next.js <=12) uses `pages/` directory. App router (Next.js 13+) uses `app/` directory with server components and layouts.

16. How does Next.js improve SEO?

By supporting SSR & SSG, Next.js allows search engines to crawl fully rendered HTML pages, improving SEO.

17. What is getInitialProps?

Legacy data fetching method. Runs on server & client. Replaced by `getStaticProps` & `getServerSideProps` in newer versions.

18. Example of getStaticProps?

```
export async function getStaticProps() { return { props: { data: 'Hello' } } }
```

19. How to add global CSS in Next.js?

You can import global styles in `pages/_app.js` using `import '../styles/globals.css'`.

20. How to add component-level CSS?

Use CSS modules: `component.module.css` and import them in your component.

21. What is next.config.js used for?

Configuration file for customizing webpack, environment variables, redirects, rewrites, etc.

22. How to deploy a Next.js app?

You can deploy to Vercel (official), Netlify, AWS, or any Node.js server using `next build` & `next start`.

23. How does Next.js handle authentication?

You can integrate with libraries like NextAuth.js, JWT, or custom middleware to protect routes.

24. Explain Static Export in Next.js.

`next export` generates a static HTML export of your site, without needing a Node.js server.

25. How to use environment variables?

Create a `.env.local` file and access using `process.env.VAR_NAME`.