Next.js Cheat Sheet

---

1. Project Setup

```
npx create-next-app@latest my-app
cd my-app
npm run dev
```

- Runs at `http://localhost:3000` - `pages` or `app` directory depending on version

---

2. Pages & Routing

- **File-based routing:** `pages/about.js` → `/about`
- **Dynamic routes:** `pages/posts/[id].js` → `/posts/1`
- **Catch-all routes:** `pages/posts/[...slug].js` → `/posts/2023/nextjs`

Navigation

```
import Link from 'next/link';
<Link href='/about'>About</Link>
```

Programmatic Navigation

```
import { useRouter } from 'next/router';
const router = useRouter();
router.push('/about');
```

---

3. Data Fetching

getStaticProps (SSG)

```
export async function getStaticProps() {
  const res = await fetch('https://api.example.com/posts');
  const posts = await res.json();
  return { props: { posts } };
}
```

getServerSideProps (SSR)

```
export async function getServerSideProps(context) {
  const res = await fetch('https://api.example.com/data');
  const data = await res.json();
  return { props: { data } };
}
```

getStaticPaths (Dynamic SSG)

```
export async function getStaticPaths() {
  return { paths: [{ params: { id: '1' } }], fallback: false };
}
```

---

4. API Routes

- `pages/api/hello.js`

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

- Accessible at `/api/hello`

---

5. Middleware

```
import { NextResponse } from 'next/server';
export function middleware(req) {
  if (!req.cookies.auth_token) return NextResponse.redirect('/login');
  return NextResponse.next();
}
export const config = { matcher: ['/dashboard/:path*'] };
```

- Runs before page rendering - Protects routes and handles redirects

---

6. Environment Variables

- `.env.local`

  ```
  NEXT_PUBLIC_API_URL=https://api.example.com
  JWT_SECRET=supersecret
  ```

- Access: `process.env.NEXT_PUBLIC_API_URL`
- Client-side must start with `NEXT_PUBLIC_`

---

7. Static Assets

- `/public` folder
- Example: `<img src='/logo.png' alt='Logo' />`

---

8. Styling

- CSS Modules: `styles/Home.module.css`
- Global CSS: `styles/globals.css`
- Tailwind CSS setup via `tailwind.config.js`
- Styled Components or Emotion supported

---

9. Image Optimization

```
import Image from 'next/image';
<Image src='/logo.png' width={100} height={100} alt='Logo' />
```

- Automatic resizing, lazy-loading, and optimization

---

10. API Calls (Client-side)

```
import axios from 'axios';
const api = axios.create({ baseURL: '/', withCredentials: true });
const data = await api.get('/api/data');
```

- `withCredentials: true` sends cookies

---

11. Deployment

- Vercel: `vercel --prod`
- Netlify, AWS, DigitalOcean supported
- Environment variables configured per environment

---

12. Useful Tips

1. `router.query` → get dynamic route params
2. `useEffect` → client-side only effects
3. `next/head` → set page `<title>` and meta tags
4. `next/script` → load third-party scripts efficiently
5. `getServerSideProps` + middleware → SSR auth
6. API routes → backend logic inside Next.js

This cheat sheet covers **everything a Next.js developer needs** for routing, API, data fetching, auth, styling, SSR/SSG, and deployment.