

# NextJS 15 metadata usage

Explain this concept of NextJS 15 and its uses:  
`import { Metadata } from "next"`

```
export const metadata: Metadata = {
  title: {
    default: "SYM-Blog-App",
    template: "%s | SYM",
  },
  description: "Save-Your-Memories",
}
```

In **Next.js 15**, the code you've shown is an example of how to define **metadata** for a page using the new **App Router** architecture. This metadata is used to manage `<head>` elements like `<title>`, `<meta name="description">`, and others, in a **type-safe and declarative** way.

## Code Breakdown:

ts

```
import { Metadata } from "next";

export const metadata: Metadata = {
  title: {
    default: "SYM-Blog-App",
    template: "%s | SYM",
  },
  description: "Save-Your-Memories",
};
```

✓ `import { Metadata } from "next"`

- This imports the **TypeScript type** `Metadata` from Next.js, which ensures your metadata object is correctly typed and supports all the available options.



`metadata` **Object**

This object is exported from a **layout.tsx** or **page.tsx** file in the **App Router** structure.

### Properties:

- `title`
    - `default` : Used when no dynamic title is provided.
    - `template` : Used to format dynamic titles. If a page sets `title: "Blog"`, the final title becomes `"Blog | SYM"`.
  - `description`
    - Sets the default meta description for the page.
- 

### Where It's Used

In **Next.js 15** (using the **App Router**):

- This `metadata` export is used to **automatically manage document head elements**.
  - No need to manually import and use `<Head>` from `next/head`.
  - It supports:
    - SEO meta tags (title, description)
    - Social sharing metadata (Open Graph, Twitter cards)
    - Icons, theme color, etc.
- 

### When and Why to Use It

- Use in any file under `app/` folder (e.g., `app/page.tsx`, `app/blog/layout.tsx`, etc.).
  - Helps with:
    - **Consistent SEO** management across your app
    - **Dynamic titles** and templates
    - **Cleaner code**, with less boilerplate
- 

### Example Usage Context

Suppose you're building a blog app:

- You place this in `app/layout.tsx` to define default metadata for the whole site.

- Then in a specific blog post page ( `app/blog/[slug]/page.tsx` ), you can export additional `metadata` to override the title/description.
- 

## Summary

Concept	Purpose
<code>metadata</code>	Define page-specific <code>&lt;head&gt;</code> info declaratively
<code>Metadata</code> type	Ensures proper structure & autocomplete
<code>title.template</code>	Dynamic title generation
Next.js 15	Fully supports this via App Router only

Let me know if you want an example with dynamic blog post titles!