**✅ .editorconfig**

**📂 What is .editorconfig?**

.editorconfig helps maintain consistent coding styles **between different editors and IDEs**. It doesn’t replace Prettier or ESLint but works in harmony with them by enforcing **base-level rules** in code editors (like VSCode, WebStorm, etc.).

🔧 IDEs like VSCode support .editorconfig natively or via plugins.

**📄 Sample .editorconfig (Production-Ready for NestJS Projects)**

# top-most EditorConfig file

root = true

# Applies to all files

[\*]

charset = utf-8 # Default encoding

end\_of\_line = lf # Use Unix-style line endings

insert\_final\_newline = true # Newline at end of every file

indent\_style = space # Use spaces instead of tabs

indent\_size = 2 # Two spaces for indentation

trim\_trailing\_whitespace = true # Remove spaces at end of lines

# Markdown and JSON files (don’t trim trailing spaces)

[\*.md]

trim\_trailing\_whitespace = false

[\*.json]

indent\_style = space

indent\_size = 2

[\*.yml]

indent\_style = space

indent\_size = 2

**✅ Why Use .editorconfig?**

| **Feature** | **Benefit** |
| --- | --- |
| indent\_style | Ensures consistent use of tabs or spaces |
| end\_of\_line | Avoids cross-platform issues (CRLF vs LF) |
| insert\_final\_newline | Required by many POSIX systems |
| trim\_trailing\_whitespace | Prevents noisy diffs in VCS |

**💡 Example Use Case**

Let’s say two developers use different editors:

* **Dev A** uses VSCode with tabs
* **Dev B** uses IntelliJ with 4-space indentation

.editorconfig ensures both write **code formatted with 2 spaces**, not tabs, and remove trailing whitespace.

**🛠 Editor Support**

| **Editor** | **Support** |
| --- | --- |
| VSCode | ✅ (built-in support) |
| WebStorm | ✅ (needs plugin or set auto-detect) |
| Sublime | ✅ |
| Atom | ✅ |
| Notepad++ | ⚠️ Partial via plugin |