

# removing a dependency from node project

To remove a dependency from a Node.js project, the `npm uninstall` command is utilized. This command effectively removes the package from both the `node_modules` directory and the `package.json` file, ensuring a complete removal.

Steps to remove a dependency:

- **Identify the package name:** Locate the exact name of the package you wish to remove within your project's `package.json` file under the `dependencies` or `devDependencies` sections.
- **Execute the uninstall command:** Open your terminal or command prompt and navigate to the root directory of your Node.js project. Then, run the `npm uninstall` command followed by the package name:

Code

```
npm uninstall <package_name>
```

- **For production dependencies:** Use the command as shown above.
- **For development dependencies:** If the package is a development dependency (listed under `devDependencies` in `package.json`), use the `-D` or `--save-dev` flag:

Code

```
npm uninstall -D <package_name>  
# or  
npm uninstall --save-dev <package_name>
```

- **For global packages:** If the package was installed globally, use the `-g` flag:

Code

```
npm uninstall -g <package_name>
```

You can also uninstall multiple packages at once by listing them separated by spaces:

Code

```
npm uninstall <package1> <package2>
```

## Explanation:

- Running `npm uninstall` removes the package's files from your `node_modules` directory.
  - Simultaneously, it updates your `package.json` file by removing the entry for the uninstalled package from either the `dependencies` or `devDependencies` list, depending on how it was originally installed. This prevents the package from being reinstalled during future `npm install` operations
- 

Revision #2

Created 29 October 2025 02:43:36 by AI API

Updated 19 November 2025 05:23:49 by AI Channel