



# Creating and publishing unscoped public packages

## > Table of contents

As an npm user, you can create unscoped packages to use in your own projects and publish them to the npm public registry for others to use in theirs. Unscoped packages are always public and are referred to by the package name only:

```
package-name
```



For more information on package scope, access, and visibility, see "[Package scope, access level, and visibility](#)".

**Note:** Before you can publish public unscoped npm packages, you must [sign up](#) for an npm user account.

## Creating an unscoped public package

1. On the command line, create a directory for your package:

```
mkdir my-test-package
```



2. Navigate to the root directory of your package:

```
cd my-test-package
```



3. If you are using git to manage your package code, in the package root directory, run the following commands, replacing `git-remote-url` with the git remote URL for your package:

```
git init
```



```
git remote add origin git://git-remote-url
```

4. In the package root directory, run the `npm init` command.
5. Respond to the prompts to generate a `package.json` file. For help naming your package, see "[Package name guidelines](#)".
6. Create a [README file](#) that explains what your package code is and how to use it.
7. In your preferred text editor, write the code for your package.

## Reviewing package contents for sensitive or unnecessary information

Publishing sensitive information to the registry can harm your users, compromise your development infrastructure, be expensive to fix, and put you at risk of legal action. **We strongly recommend removing sensitive information, such as private keys, passwords, personally identifiable information (PII), and credit card data before publishing your package to the registry.**

For less sensitive information, such as testing data, use a `.npmignore` or `.gitignore` file to prevent publishing to the registry. For more information, see [this article](#).

## Testing your package

To reduce the chances of publishing bugs, we recommend testing your package before publishing it to the npm registry. To test your package, run `npm install` with the full path to your package directory:

```
npm install my-package
```



## Publishing unscoped public packages

1. On the command line, navigate to the root directory of your package.

```
cd /path/to/package
```



2. To publish your public package to the npm registry, run:

```
npm publish
```



3. To see your public package page, visit [https://npmjs.com/package/\\*package-name](https://npmjs.com/package/*package-name), *replacing* package-name\* with the name of your package. Public packages will say `public` below the package name on the npm website.

For more information on the `publish` command, see the [CLI documentation](#).