Assignment -1
NPM Registration and Publishing

1. Install and check version for Node
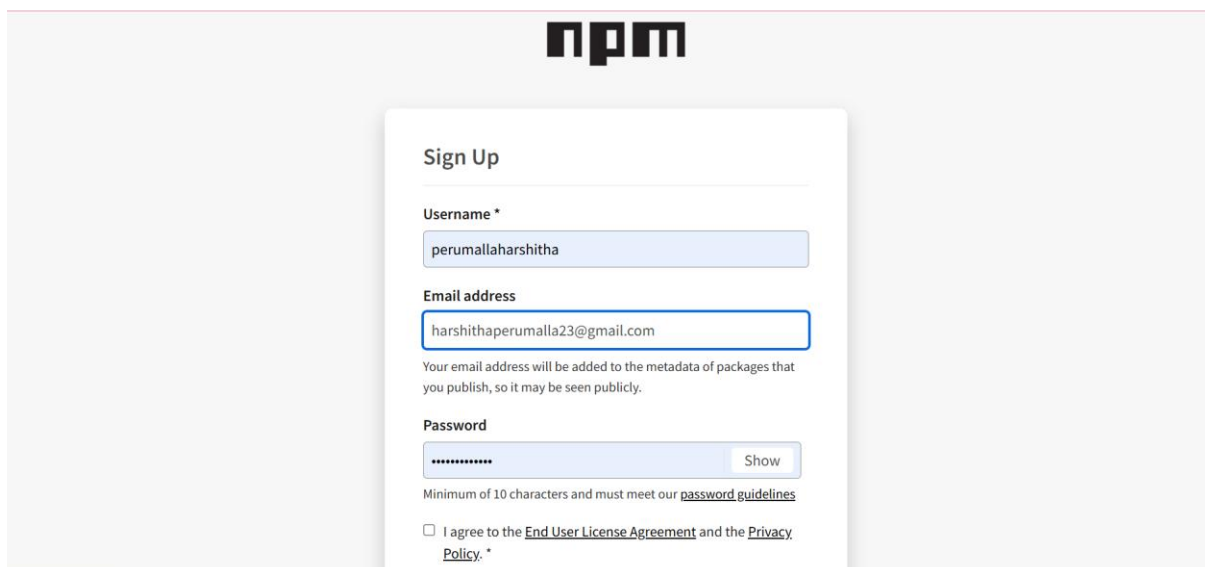
```
C:\Users\harsh>node -v
v22.12.0
```

2. Install and check version for NPM

```
C:\Users\harsh>npm -v
10.9.0
```

3. Register an NPM Account

1. Open a browser and go to npmjs.com.

2. Register for an npm account by providing a username, email, and password.

4. Initialize a Git Repository

1. Create a folder for the package in Git to maintain version control.

2. Open your terminal or command prompt and run the following

```
C:\Users\harsh>git init
Initialized empty Git repository in C:/Users/harsh/.git/

C:\Users\harsh>mkdir package

C:\Users\harsh>cd package

C:\Users\harsh\package>
```

5. Create a New JavaScript File

1. Open Visual Studio and create a new JavaScript file named `index.js` in your `package_example` folder.
2. **Add the following content in** `index.js`:

```
node_modules > 22e5 > JS index.js
   1    console.log("Hii there");
```

3. Save the file

6. Run the Script with Node

1. Open a terminal in the project folder and run:

```
● PS D:\Projects\NPM> cd node_modules
● PS D:\Projects\NPM\node_modules> cd 225e5
● PS D:\Projects\NPM\node_modules\225e5> node index.js
  Hello
✧ PS D:\Projects\NPM\node_modules\225e5>
```

7. Initialize NPM Package

1. In your terminal, run

   npm init

2. You'll be prompted to enter information for your package.json file:
   a. **name**: A unique name for your package (lowercase, no spaces).
   b. **version**: Starting at 1.0.0.
   c. **description**: A brief description of your package.
   d. **entry point**: Set to index.js by default.
   e. **test command**: Leave it blank or set it to a custom test command.
   f. **git repository**: Your GitHub repository URL (optional).
   g. **keywords**: List relevant keywords.
   h. **author**: Your name.
   i. **license**: You can use ISC or any other license.

```
Command Prompt                                                          —  □  ×

C:\Users\harsh\package>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (package) npm_package_225e5
version: (1.0.0)
description: sample package
entry point: (index.js)
test command:
git repository: https://github.com/perumallaharshitha/package.git
keywords:
author: Harshitha
license: (ISC)
About to write to C:\Users\harsh\package\package.json:

{
  "name": "npm_package_225e5",
  "version": "1.0.0",
  "description": "sample package",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "repository": {
    "type": "git",

version: (1.0.0)
description: sample package
entry point: (index.js)
test command:
git repository: https://github.com/perumallaharshitha/package.git
keywords:
author: Harshitha
license: (ISC)
About to write to C:\Users\harsh\package\package.json:

{
  "name": "npm_package_225e5",
  "version": "1.0.0",
  "description": "sample package",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/perumallaharshitha/package.git"
  },
  "author": "Harshitha",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/perumallaharshitha/package/issues"
  },
  "homepage": "https://github.com/perumallaharshitha/package#readme"
}

Is this OK? (yes) yes
```

8. Log in to NPM

```
C:\Users\harsh>npm login
npm notice Log in on https://registry.npmjs.org/
Login at:
https://www.npmjs.com/login?next=/login/cli/7ef72a91-7bf0-4fef-80ce-45d08eebf520
Press ENTER to open in the browser...
```

9. Verify Logged-In User

1. To verify that you're logged in, run : npm whoami

10. Publish the Package to npm

1. Once you're ready to publish, run: npm publish

```
C:\Users\harsh\package>npm whoami
perumallaharshitha

C:\Users\harsh\package>npm publish
npm notice
npm notice package: e5_npm_package@1.0.0
npm notice Tarball Contents
npm notice 495B package.json
npm notice Tarball Details
npm notice name: e5_npm_package
npm notice version: 1.0.0
npm notice filename: e5_npm_package-1.0.0.tgz
npm notice package size: 335 B
npm notice unpacked size: 495 B
npm notice shasum: 8646335529e8a257d95784eb38a95749971fac33
npm notice integrity: sha512-boeb3/CJ9Le0O[...]PZvd4mY1g9tKw==
npm notice total files: 1
npm notice
npm notice Publishing to https://registry.npmjs.org/ with tag latest and default access
+ e5_npm_package@1.0.0
```


Git and Version Control Steps
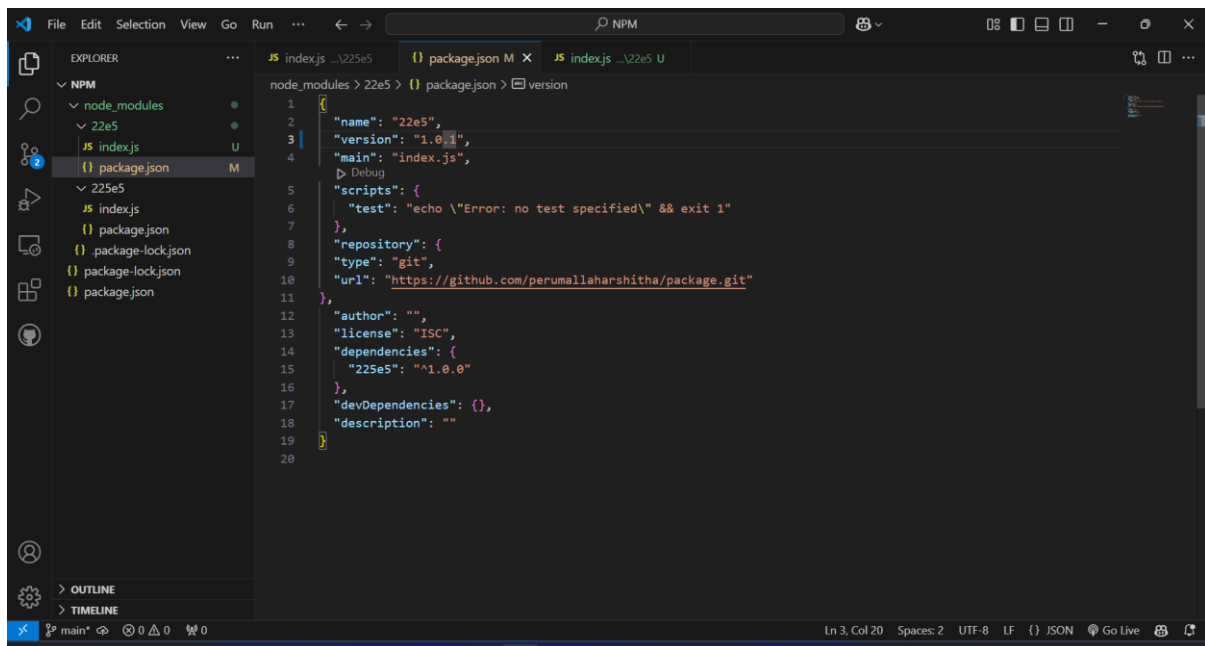
11. Add Your Code to Git Repository:

1. After initializing your Git repo, add all files to Git
2. Link your local repo to a remote GitHub repo (create one on GitHub first)

12. Update Package Version:

1. If you need to update the package (e.g., fix bugs or add new features), you should update the version number following semantic versioning:
   - **Major**: For breaking changes.
   - **Minor**: For new features that are backward compatible.
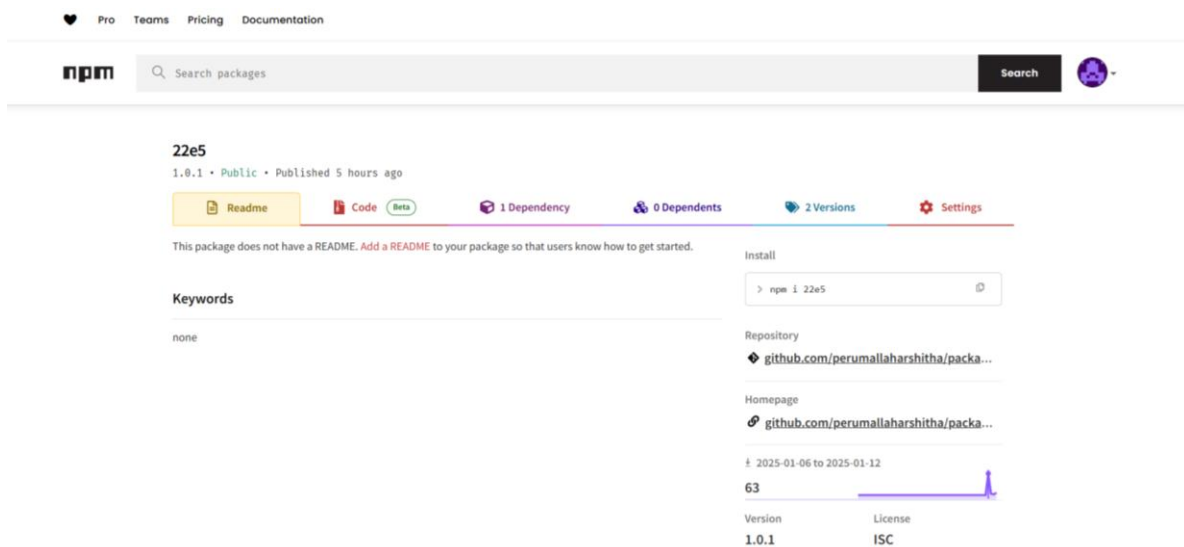   - **Patch**: For backward-compatible bug fixes.

13. Publish Updated Version to npm:

1. Once the version is updated, publish the new version to npm again