

1. Install Node.js and npm/yarn/pnpm

Ensure you have **Node.js** installed, which includes **npm** (Node Package Manager).

Check if Node.js is installed:

```
node -v
```

-

Check npm version:

```
npm -v
```

- - If not installed, download and install from [Node.js official site](https://nodejs.org/en/).
-

2. Initialize a New TypeScript Project

Create a project directory and initialize it:

```
mkdir typescript-project && cd typescript-project  
npm init -y
```

This creates a `package.json` file with default values.

3. Install TypeScript and Required Dependencies

Based on your given `package.json`, install TypeScript and other dev dependencies:

```
pnpm install --save-dev typescript vitest nodemon prettier tsd  
vite-tsconfig-paths @total-typescript/exercise-cli  
@total-typescript/helpers cross-fetch npm-run-all
```

4. Generate a `tsconfig.json` Configuration File

Run the following command to generate a default `tsconfig.json`:

```
npx tsc --init
```

Then, modify the `tsconfig.json` to enable strict type-checking and improve development experience:

```
{
  "compilerOptions": {
    "target": "ESNext",
    "module": "NodeNext",
    "moduleResolution": "NodeNext",
    "outDir": "dist",
    "rootDir": "src",
    "strict": true,
    "esModuleInterop": true,
    "forceConsistentCasingInFileNames": true,
    "skipLibCheck": true
  },
  "include": ["src"],
  "exclude": ["node_modules"]
}
```

5. Set Up the Project Structure

Create the following folders and files:

```
mkdir src
touch src/index.ts
```

Edit `src/index.ts`:

```
console.log("Hello, TypeScript!");
```

6. Add Scripts to `package.json`

Modify your `package.json` scripts section:

```
"scripts": {  
  "dev": "tsc --watch",  
  "dev:start": "nodemon src/index.ts",  
  "start": "node dist/index.js",  
  "build": "tsc",  
  "test": "vitest"  
}
```

- **dev** → Runs TypeScript compiler in watch mode.
- **start** → Runs the compiled JavaScript file.
- **build** → Compiles TypeScript to JavaScript.
- **test** → Runs tests using Vitest.

7. Run the TypeScript Compiler

Compile your TypeScript code:

```
npm run build
```

This generates the compiled `dist/index.js` file.

Run it with:

```
node dist/index.js
```

Or use **watch mode** for auto-compilation:

```
npm run dev
```

8. Setup Prettier for Formatting

Create `.prettierrc`:

```
echo '{
  "semi": false,
  "singleQuote": true,
  "tabWidth": 2
}' > .prettierrc
```

Format the code:

```
npx prettier --write .
```

9. Use Nodemon for Auto-Restart

For automatic restarts on file changes, modify `package.json`:

```
"scripts": {
  "dev": "nodemon --ext ts --exec ts-node src/index.ts"
}
```

Run:

```
npm run dev
```

10. Run Tests with Vitest

Inside `src/index.test.ts`, add:

```
import { expect, test } from "vitest";

test("Basic test", () => {
  expect(2 + 2).toBe(4);
});
```

Run tests:

```
npm run test
```

