**TS** TypeScript
TUTORIAL

# TypeScript "Hello, World!"

If this site saves you hours of work, please **whitelist it in your ad blocker 😭 to support us 💖** in creating more helpful and free content in the future.

**Summary**: in this tutorial, you'll learn how to develop the Hello World program in TypeScript.

## TypeScript Hello World program in Node.js

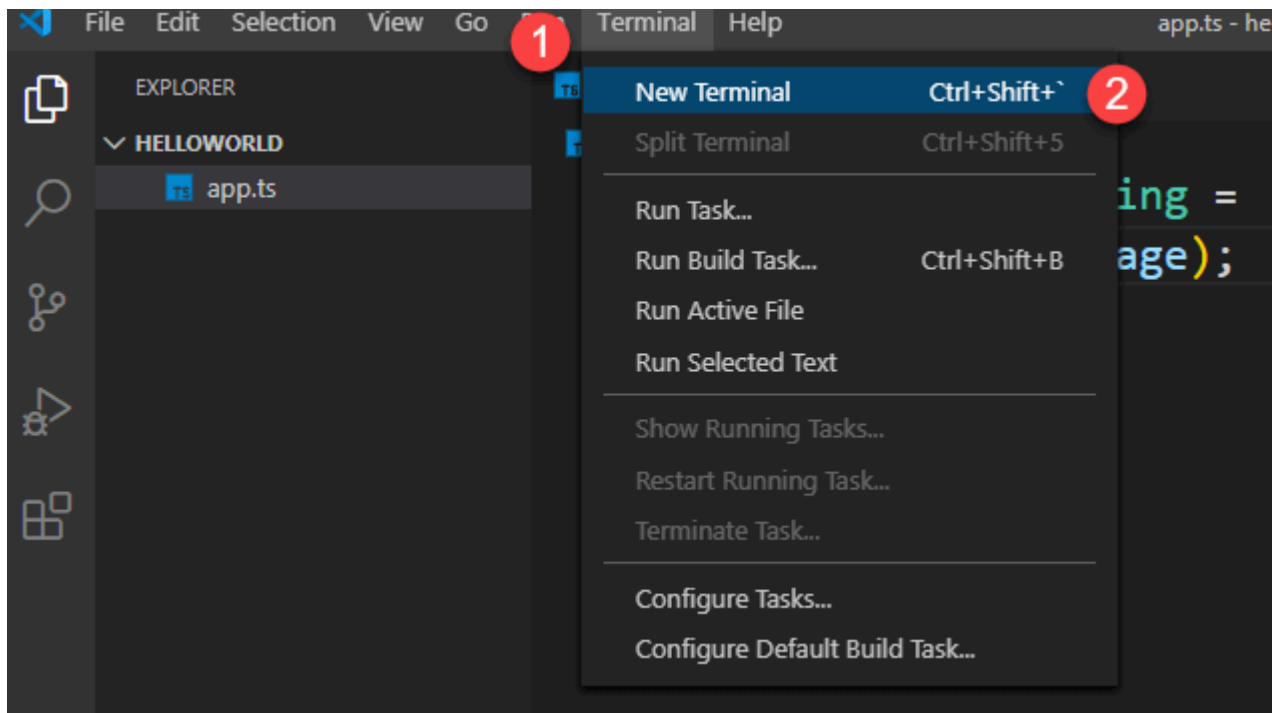First, create a new directory to store the code, e.g., `helloworld`.

Second, launch VS Code and open that directory.

Third, create a new TypeScript file called `app.ts`. The extension of a TypeScript file is `.ts`.

Fourth, type the following source code in the `app.ts` file:

```typescript
let message: string = 'Hello, World!';
console.log(message);
```

Fifth, launch a new Terminal within the VS Code by using the keyboard shortcut `Ctrl+` or follow the menu `Terminal > New Terminal`

Sixth, type the following command on the Terminal to compile the `app.ts` file:

```
tsc app.ts
```

If everything is fine, you'll see a new file called `app.js` is generated by the TypeScript compiler:

To run the `app.js` file in `Node.js` , you use the following command:

```
node app.js
```

If you installed the `ts-node` module mentioned in the [setting up TypeScript development environment](#), you can use just one command to run the TypeScript file directly on Node.js without precompiling it to JavaScript:

```
ts-node app.ts
```

# TypeScript Hello World program in Web Browsers

The following steps show you how to create a webpage that shows the `Hello, World!` message on web browsers.

First, create a new file called `index.html` and include the `app.js` as follows:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>TypeScript: Hello, World!</title>
</head>
<body>
    <script src="app.js"></script>
</body>
</html>
```

Notice that `ts-node` will compile the `app.ts` file and create the `app.js` file.

Second, change the `app.ts` code to the following:

```typescript
let message: string = 'Hello, World!';

// create a new heading 1 element
let heading = document.createElement('h1');
heading.textContent = message;
```

```
// add the heading the document
document.body.appendChild(heading);
```

Third, compile the `app.ts` file into the `app.js` file:

```
tsc app.ts
```

Fourth, open the Live Server from the VS code by right-mouse-clicking the `index.html` and select the Open with Live Server option:

The Live Server will open the `index.html` with the following message:

To make changes, you need to edit the `app.ts` file. For example:

```typescript
let message: string = 'Hello, TypeScript!';

let heading = document.createElement('h1');
heading.textContent = message;

document.body.appendChild(heading);
```

And compile the `app.ts` file:

```
tsc app.ts
```

The TypeScript compiler will generate a new `app.js` file and the Live Server will automatically reload it on the web browser.

Note that the `app.js` is the output file of the `app.ts` file, therefore, you should never directly change its contents, or you'll lose the changes once you recompile the `app.ts` file.

In this tutorial, you have learned how to create the first program in TypeScript called `Hello, World!` that works on Node.js and web browsers.