Let's take a high level view of what we'll be doing. The steps we'll take are as follows:

1. Setup our base project
2. Setup lite-server
3. Setup TypeScript
4. Grab packages and setup System.js
5. Build with Angular 2!

The directory structure for our app will look like this:

|- app/

|- app.component.ts // main app component

|- app.module.ts // main app module

|- main.ts // bootstrap our app

|- index.html

|- package.json

|- tsconfig.json

|- systemjs.config.json

**TypeScript**

* TypeScript হচ্ছে Angular2 এর একটি প্রাথমিক ভাষা ।
* TypeScript development এর সময় টাইপ-নিরাপত্তা ও OOP সুবিধা debugging সহায়তা, ide Intellisense প্রদান করে যা JavaScript এ পাওয়া যায় না ।
* এ ছাড়া JavaScript এর functional code বুঝা কঠিন তবে TypeScript এর code C# /Java এর মতো যা বুঝা সহজ।
* Browsers TypeScript সরাসরি execute করতে পারে না এইজন্য অনুবাদক দরকার যে TypeScript কে JavaScript এ রূপান্তরিত করবে।

**Installing TypeScript**

There are two main ways to get the TypeScript tools:

* Via npm (the Node.js package manager)
* By installing TypeScript’s Visual Studio plugins. Visual Studio2017, 2015 and Visual Studio 2013 Update 2 include TypeScript by default.

**Setting Up Type script**

* Create an empty folder
* Open it with visual studio code.
* Open integrated terminal
* Installing TypeScript and configure tsconfig.json

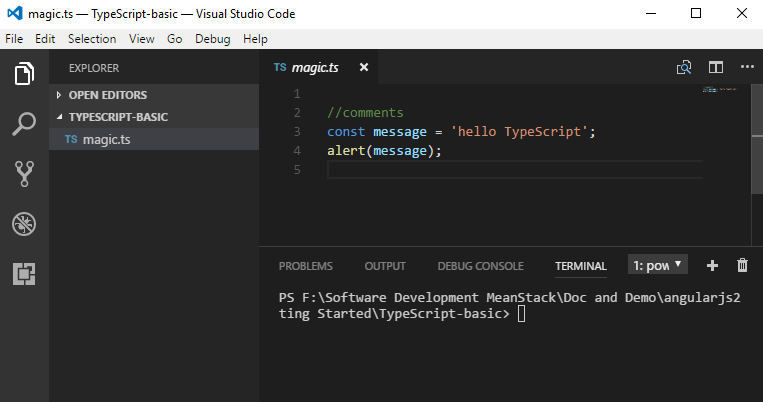
Terminal Commands

npm install -g typescript

Check typescript version

tsc –v

Create typescript file :



//comments

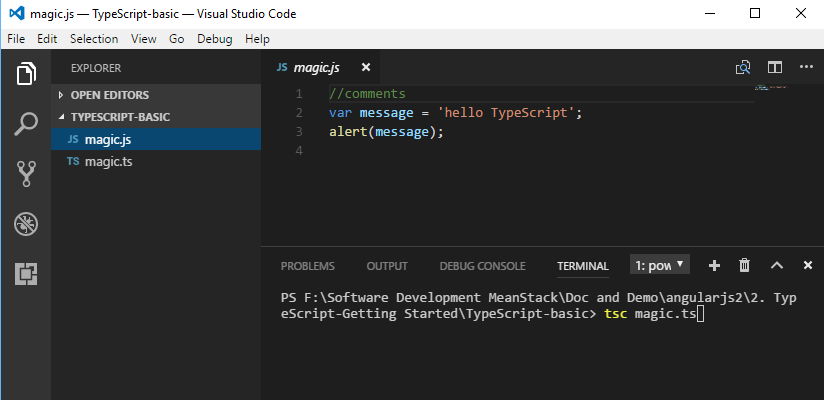
const message = 'hello TypeScript';

alert(message);

COMPILEtypescript file:

tsc magic.ts

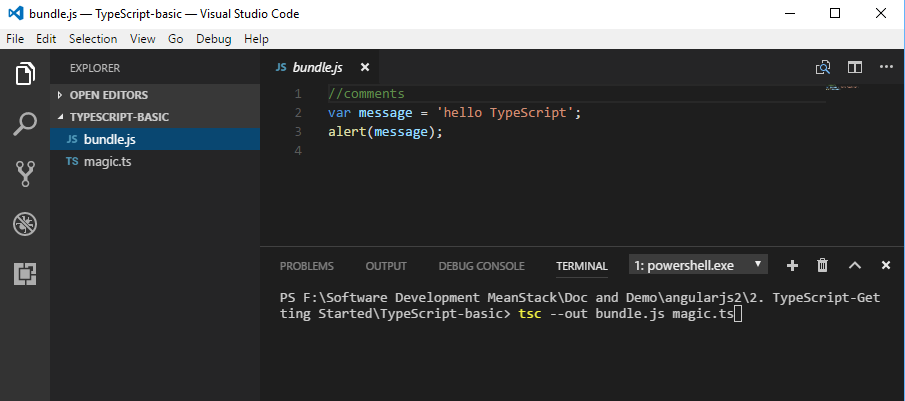
* It will create magic.js file.



যদি ভিন্ন নাম যেমন bundile.js নামে file তৈরী করতে চান তখন নিচের command টি লিখুন

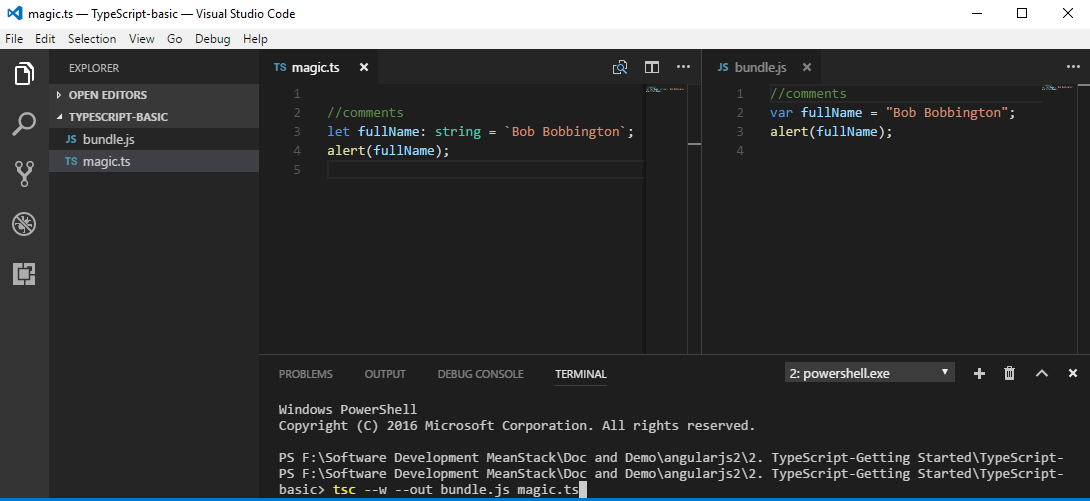
tsc --out bundle.js magic.js

* It will create bundle.js file.



যদি magic.ts file এর কোনো পরিবর্তন হলে আবার Compileকরতে হবে। যাতে বার বার Compile না করা লাগে এবং তৎক্ষণাৎ যেন ts থেকে js এ রূপান্তরিত হয়ে যায় এজন্য নিচের command টি লিখুন

tsc –w --out bundle.js magic.js



যদি একসাথে একাধিক ts file থেকে js file এ compile করতে চান

tsc –-init

* tsconfig.json নামক file তৈরী হবে
* সাধারণত tsconfig.json নামক file হচ্ছে TypeScript কে JavaScript এ রূপান্তরিত করার default file.
* একটি directory তে tsconfig.json ফাইলের উপস্থিতি নির্দেশ করে যে এটি TypeScript project এর root file এবং এই directory র সব ts file কে js file এ রূপান্তর করতে হবে
* Browsers TypeScript সরাসরি execute করতে পারে না এইজন্য অনুবাদক দরকার যে TypeScript কে JavaScript এ রূপান্তরিত করবে।
* আরো ভালো জানার জন্য নিচের লিংক গুলো পড়ুন :

<https://www.typescriptlang.org/docs/handbook/tsconfig-json.html>

<https://www.typescriptlang.org/docs/handbook/compiler-options.html>

<https://angular.io/guide/typescript-configuration>

{

"compilerOptions": {

/\* Basic Options \*/

"target": "es5",

"module": "commonjs",

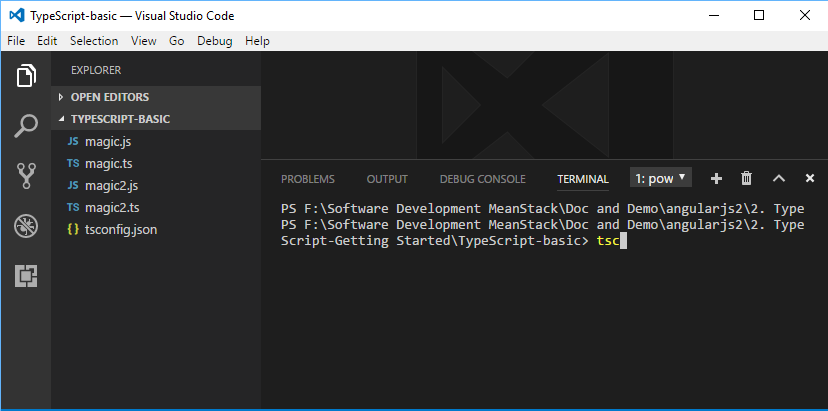
"strict": true

}

}

সব typescript file একসাথে COMPILE করুন :

tsc



ts file গুলো কে আদালা directory এবং js file গুলো কে আলাদা directory করা

{

"compilerOptions": {

"target": "es5",

"module": "commonjs",

"outDir": "./dist",

"rootDir": "./src",

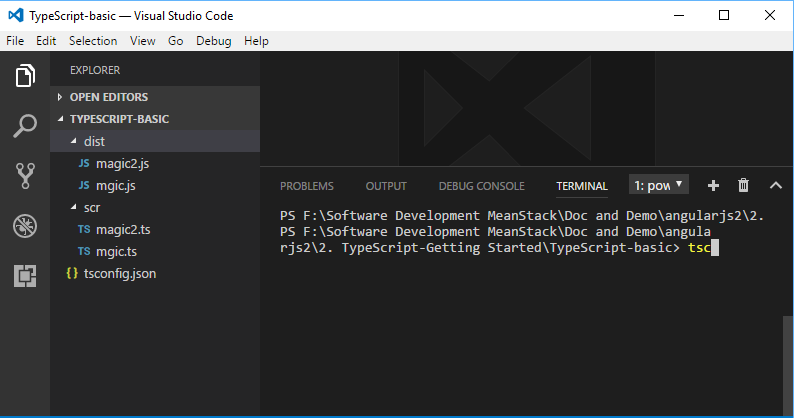
"strict": true

}

}

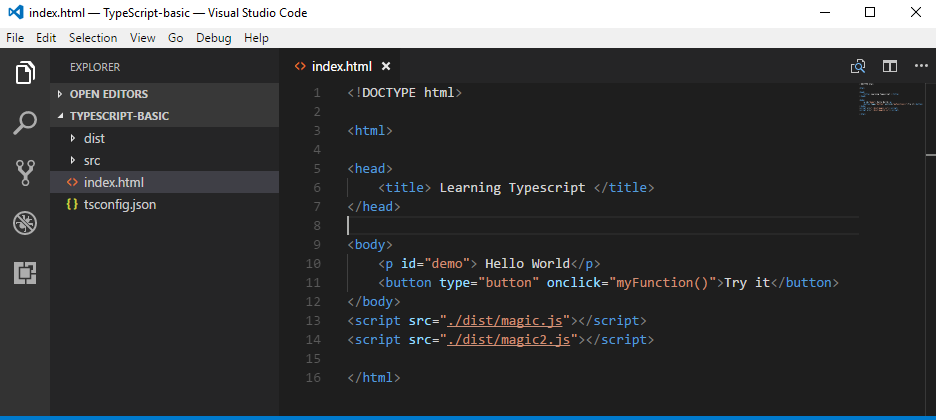
COMPILE করুন :

Tsc --w



Html এর সাথে ts file থেকে রূপান্তরিত js file এর configuration

* একটি html তৈরী করুন
* ts file থেকে রূপান্তরিত js file এর link দিন
* html file টি browser এ run করুন
* লক্ষ্য করুন ts file change করে save করলে সাথে সাথে js file গুলো update হয়ে যাচ্ছে তবে html file টি update হচ্ছে না reload করলে update হচ্ছে
* আরেকটা জিনিস typescript globally install করার কারণে typings লাগছে না তবে locally install করলে লাগবে যা আমরা পরবর্তী এ করবো



Html file :

<!DOCTYPE html>

<html>

<head>

<title> Learning Typescript </title>

</head>

<body>

<p id="demo"> Hello World</p>

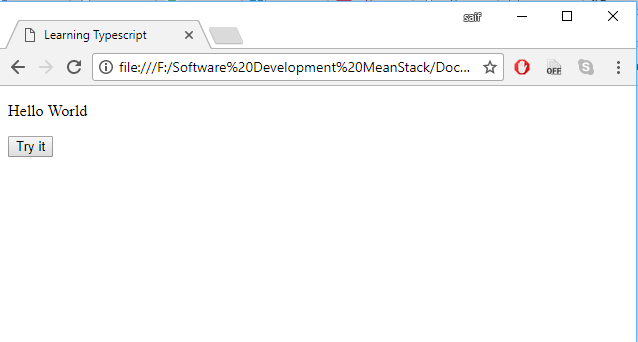
<button type="button" onclick="myFunction()">Try it</button>

</body>

<script src="./dist/magic.js"></script>

<script src="./dist/magic2.js"></script>

</html>



**Configuring a New TypeScript Project**

* Create an empty folder
* Open it with visual studio code.
* Open integrated terminal
* TypeScript install and tscofig.json configuration
* Create package.json

Terminal Commands

tsc –-init

* tsconfig.json file টি তৈরী হবে
* tsconfig.json file নিচের মতো করুন

{

"compilerOptions": {

"target": "es5",

"module": "commonjs",

"moduleResolution": "node",

"sourceMap": false,

"emitDecoratorMetadata": true,

"experimentalDecorators": true,

"removeComments": false,

"lib": [

"es2015",

"dom"

],

"noImplicitAny": true,

"suppressImplicitAnyIndexErrors": true,

"outDir": "./dist",

"rootDir": "./src"

},

"exclude": [

"node\_modules",

"dist"

]

}

Terminal Commands

npm init -y

* package.json file টি তৈরী হবে

{

"name": "TypeScript-config",

"version": "1.0.0",

"description": "",

"main": "index.js",

"scripts": {

"test": "echo \"Error: no test specified\" && exit 1"

},

"keywords": [],

"author": "",

"license": "ISC"

}

Terminal Commands for install **typescript and types**

npm install --save-dev typescript

npm install --save-dev @types/node

npm install --save-dev @types/jasmine

npm install --save-dev @types/core-js

অথবা একসাথে

npm install --save-dev typescript @types/node @types/jasmine @types/core-js

Terminal Commands for install **lite-server**

npm install --save-dev lite-server

Terminal Commands for install **concurrently**

npm install --save-dev lite-server

* lite-server এবং TypeScript watcher একই সময়ে run করার জন্য **concurrently** install করতে হবে।
* উপরুক্ত package গুলা install করার ফলে node\_modules folder create হবে
* এবং package.json file টি update হবে

{

"name": "TypeScript-config",

"version": "1.0.0",

"description": "",

"main": "index.js",

"scripts": {

"test": "echo \"Error: no test specified\" && exit 1"

},

"keywords": [],

"author": "",

"license": "ISC",

"devDependencies": {

"@types/core-js": "^0.9.42",

"@types/jasmine": "^2.5.53",

"@types/node": "^8.0.6",

"concurrently": "^3.5.0",

"lite-server": "^2.3.0",

"typescript": "^2.4.1"

}

}

এখন package.json এর script লিখুন নিন্মোক্ত ভাবে

{

"name": "tpescript-config",

"version": "1.0.0",

"description": "",

"main": "index.js",

"scripts": {

"start": "tsc && concurrently \"npm run tsc:w\" \"npm run lite\"",

"lite": "lite-server",

"tsc": "tsc",

"tsc:w": "tsc -w"

},

"keywords": [],

"author": "",

"license": "ISC",

"devDependencies": {

"@types/core-js": "^0.9.37",

"@types/jasmine": "^2.5.45",

"@types/node": "^7.0.8",

"typescript": "^2.2.1",

"concurrently": "^3.4.0",

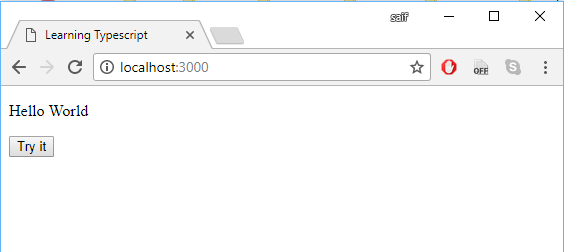
"lite-server": "^2.3.0"

}

}

Terminal Commands

npm start



এখন যেকোনো ts file এ change করে save করুন tsc –-w এর কারণে যখন ts এ js রূপান্তরিত হচ্ছে দেখুন একই সময়ে যেহেতু js পরিবর্তন হচ্ছে তাই html file টি reload হচ্ছে

Resources :

<https://scotch.io/courses/getting-started-with-angular-2/typescript-setup>

<https://scotch.io/courses/getting-started-with-angular-2>

<https://egghead.io/lessons/typescript-configuring-a-new-typescript-project>

<https://egghead.io/lessons/typescript-installing-typescript-and-running-the-typescript-compiler-tsc>