TypeScript:

1. A Programming Language created by Microsoft to address shortcomings of java-Script
2. Typescript is a programming Language build on top of java-script

Statis Typing

Code Completion

Refactoring

Shorthand Notations

1. TypeScript is like Java-script with type checking

Drawbacks:

Transpilation : that compile the typescript into js

Discipline in Coding

Installation:

npm to install typescript

npm i -g typescript

verify the installation

tsc -v

Create new Folder mkdir folderName

To compile the typescript file

tsc filename or tsc

Typescript- Type Annotations  
Where we can specify the type of variable, function parameters and object properties

In TypeScript we can catch a lot of mistakes using compile times

Es5 (Ecma-Script) standard or a specification while java-Script is an implementation of that specification

Implicit: guess the type based on the assigned Values

Explicit: writing the annotating types with the variable

Creating the Configure File for TypeScript Compiler

tsc --init

In tsconfig.json File

1) target (which specifies the version of js that typescript compiler gonna generate)

2)Module(common js) talk later…..

3)rootDir the directory that contains our source file

4)OutDir it specify the directory which contains our javascript file

5)removecomments it remove all the comments in typescript file and compile the comment free js code

6)noemitonerror bydefault when we compile our code if we have error it still generate the js file

Debugging Typescript Application:

7)SourceMap is a file that specific how each line of our typescript code maps to the generated javascript code

And run tsc again

IN debugging mode in launch.json file add this file

"preLaunchTask": "tsc: build - tsconfig.json",

“preLaunchTask”: “tsc: build – tsconfig.json”,

By this setting:

We are telling vs code to use the typescript compiler to build our application using this configure file

Step over F10 :

For Executing one line of the code

Step into F11:

Stepping into the Function

Step Out Shift + F11:

Stepping outside of the Function

Restart

Stop

FUNDAMENTAL Type of TypeScript

Built-in-Types:

JavaScript Have:

Number, string, Boolean, null, undefined, object

TypeScript:

Any, unknown, never, enum, tuple

In Typescript if u have large number we can separate the digits by underscore

To make the Code More Readable

Any:

If we declare anything and didn’t initialize it

Bcz we use typescript for type security

So u should avoid any type as much as possible

Implicity means we haven’t explicity or clearly set the type of this paramets

You can change the tsconfig.json file as well

To avoid the any error

By Changing the noimplicitAny:false

Arrays:

let numbers: number[] =[];

numbers.forEach()

if u didn’t describe type it will take it as a any

Tuples:

Fixed length array where each element has a particular type

We often use them when we have pair of values

If we compile them we see regular js array

*// user.push(1) troublesome {compiler will not give the error }*

**Enums:**

**A list of related constants**

**If we use enum with const it will generate more optimize java-Script code**

**PascalCase Pascal Naming convention it mean the first letter will be the uppercase**

**If we initialize with the string then we have to initialize with all of them**

Functions

\*Void meaning this function doesn’t return any value

The Parameter and return should probably be annotated in Typescript for better user experience

Typejson file

If u have unused parameter

Type Checking Section : NoUnusedParameters:true

If we forget to return the value

Type Checking Section : NoImplicitReturns:true

If you didn’t use any local variable

Type Checking Section : NoUnusedLocals:true

You can add ? if u didn’t pass that parameter value

Objects:

Readyonly if we accidently didn’t change them

Advance Steps

Type Aliases:

With aliases we can define custom types

Union Types:

We can give function parameter more than one type

To use specific method of number and string we user narrowing

Intersection Types: