**TYPESCRIPT:**

Typescript is used for type checking. It is used to check all errors related to typescript before runtime.

It is a transpiler or transcompiler based language. It is based on javascript. When we write something in typescript. Typescript compiler turns it into js which then runs into the browser. Js runs in browser, not typescript. In typescript, we write js syntax in the form typed language. By this, all typescript identify error in compiling time. Typescript is the super set of js. It decreases development time by 1/2.

File extension -> filename.ts

**Installation:**

The below will install typescript globally in whole computer.

npm install -g typescript

**Starting:**

Create index.ts file

Then write tsc .\above file name

**Example:**

Tsc .\index.ts

This will create js file

**Create complete project:**

Npm init

Then name of the project

We can change the description entry point etc later

After that, it will create package.json

**For tsconfig.json:**

tsc –init

Like in react, we create components folder and place all of our files in that, in typescript, the best practice is to create src folder and put all src files their. So compiler compiles all the ts files in it.

**Some changes in the ts.config:**

     "outDir": "./dist",                        /\* Redirect output structure to the directory. \*/

     "rootDir": "./src",

After that create src folder. Move index.ts in it. Now we just need to write

tsc

In the vs terminal, that will compile all file in the src folder and generate dist folder containing .js files like index.js

If we want to check output through node, then

Node .\dist\index.js

We can simply create index.html file and use index.js or any .js file that was created from .ts file

**Sumarizing the starting steps:**

1. Npm init
2. tsc –init
3. Some changes in the ts.config:

     "outDir": "./dist",                        /\* Redirect output structure to the directory. \*/

     "rootDir": "./src",

1. create src folder.
2. Index.ts in it.

**To check errors:**

Write tsc in vs terminal

**To check output of .js:**

node .\dist\index.js or file.js

**any Type:**

If we want a type in typescript that can take any value, we use type “any”. Type any is used when want dynamic variable.

let b : number = 10;

// we cannot do this b = "string"

let c : any = 100;

c = "type change due to any";

//as due to any type, we can assign any type to variable c

console.log(c);

Now we can also assign constant type like this. This is like something when we want some serious strict action

let d : true = true;

// d = false; -> gives an error as we assign true type, so it will only accept true

let e : 20 = 20;

// e = 30 or any number; -> gives an error as we assign 20 type, so it will only accept 20

Symbol data type can be used to create symbols

We can also specify certain properties or key value in object data type

let user : {firstName: string, lastName: string} = {

    firstName: "Muhammad Muneeb Waseem",

    lastName: "Waseem"

}

// we can not add more key value pair or more properties in object user as we specify properties initially

console.log(user);