**TypeScript**

* Better JavaScript
* Implementation of ES6
* Superset of JavaScript
  + Using TypeScript compiler JavaScript code can be compiled
* Developed by Microsoft
* Features
  + Strongly typed (strictly types)
    - A number variable can not have any value other number
  + Object Oriented Programming Language
* To check if typescript is installed on your machine
  + > tsc
* To install locally
  + > sudo npm install -g typescript
* JavaScript -> loosely typed
* TypeScript -> strictly typed



**To run TS code**

* Transpile the TS code into JS
* Transpile = translate + compile
* Syntax:

tsc <file name>

* E.g.

tsc page1.ts

* Run the JS file using node
* Syntax:

node <file name>

* E.g.

node page1.js



**Variable declaration**

* Variables can be declared
  + Using type inference (implicitly)
    - TS will automatically decide the data type
    - E.g.

var num = 100;

// num will be declared as number

// num = “100”; // error: string can be stored in number

* + Using a specific data type (explicitly)
    - Syntax:

let <var name> : <data type>;

* + - E.g.

let num: number;

let firstName: string;

let canVote: boolean;

let value: any;

**Data Types**

* Number
  + Can store only number

var num = 100;

var num:number = 100;

//num = “test”; // error

* string

can store only string values

var firstName = “test”;

var firstName:string = “test”;

//firstName = true; // error

* boolean

can store only true or false

var canVote = true;

var canVote:Boolean = true;

// canVote = “test”; // error

* Object
* undefined
* any
  + can store **any** value

let num: any;

num = 100;

num = “test”;

num = true;

**Function**

* Syntax:

function <function name> (<param name>:<data type>) : <return type> {

// function body here

}

* E.g.

function add(p1:number, p2:number) : void {

console.log(“add: “ + (p1 + p2));

}

function divide(p1:number, p2:number) : number {

return p1 / p2;

// return “test”; //error

// return true; // error

}