Angular

- A UI framework built by Google
- A component based framework for building scalable app
- Provided collection of libraries to cover features like routing, forms etc.

Frameworks:

- Frameworks generally contains set of libraries
- You can extend frameworks.
- eg- Angular, .Net, Ionic, ExpressJS

Libraries:

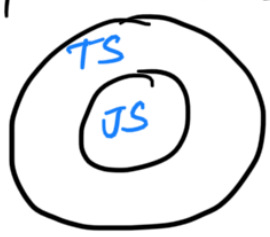
- perform speeific operation
- you can combine multiple libs to make an app.
- eg- ReactJS, MomentJS, Lodash.

Features of Angular

- Template, Data binding, forms, Routing, Observables, PWA

Typescript (TS)

- A strongly typed programming language by Microsoft
  eg. java. where u declare the type of var beforehand.
- Superset of Js.



Why Typescript?

- Compiles to Javascript.
- Keeps code evergreen
- Supports all major libraries and frameworks
- React, Vue, Angular, express etc. all can be written in TS.

Type Safety:

- Keeps app free from Type Errors. eg - C#, Java
- Keeps your js code free from undefined and null values
- In TS types are stripped when your code is converted to js.

Run a ts program:

1. Install node (Use latest version)
2. 'npm init' to create a package.json
3. Install typescript 'npm i typescript' (npm install -g typescript).
4. Initialize ts `tsc --init'

tsconfig.json
- where you actually change version and compile your js to es 2015, es 2010 etc

- 'target': js version, all modern browsers support 'es 2015'

- 'module': where you want to run your code - browser, os, cmd, etc. In Angular it is 'esnext'

tsc - Typescript comes with its own compiler 'tsc'

SPA - Single Page Application
- In older days when we used to create websites we used to create many pages. html code which was not reusable
- To switch b/w urls there used to be page refresh. whenever you clicked on a url the entire compilation used to happen on server side

- With SPA, it does not make request to server for every url request
- The entire resources are sent to client browser, so the browser has it downloaded and upon request particular pages are rendered. So a page is not reloaded in this case, just a new view is rendered

- Angular has routing functionality to create SPA.
- Angular also offers SSR (Server side rendering) which support SPA (good for SEO)

* Data Types:
- string, number, boolean, array, enum, tuple, any, void, never

Typescript can determine the data type from value assigned
```
let fname = 'Harry'

fname = 10      // gives error as it assign string to fname.
```

Note - Define datatype of var as much as you can as it may sometimes assign incorrect datatype.

1. String
```
let fname : string;      OR      let lname = "Harry"
fname = "Harry";
```

2. Number

let page : number OR let page = 20
page = 20
page = 20.5 (you can do decimal)

let age = parseInt("25")

3. boolean
let flag : boolean
If you try to print flag you will get error., because a value is not assigned and when it converts to js it finds no value hence throws error. This is because of 'strict' type in tsconfig.json.

4. Array
let list = []
list = [1, 2, 3, 4, 5]

let emplist : string[]
OR
let emplist : Array<string>

list.filter((num) => num > 2)
emplist.find((emp) => emp == "b")
list.reduce((num, sum) => num + sum)

5. enum
const enum Color { green, blue, red }

let c : Color = Color.blue;

//when you compile it, some lines of code are generated for enum. But you dont need these values in production. So you can declare enum as const. It will stream down everything, remove those lines of code generated just keep a index eg- 0 for green.

6. tuple
let swapNums : [number, number];

function swapNumbers(num1, num2) : [number, number] {
    return [num2, num1]
}

swapNumbs = swapNumbers(10, 20);
swapNumbs[0]

—u——— LI]

(used to return tuple where you have say 2 values to return)

7. any (DO NOT USE IT)

- kind of equivalent

```
let dept : any
dept = "IT"
dept = 10
```

8. void