TRAINING TOPICS

Links For Front End

* <https://www.freecodecamp.org/news/html-css-and-javascript-explained-for-beginners/> - Day 1
* HTML . Start from <https://www.w3schools.com/html/> - Day 2 to Day 5
* CSS, Start from <https://www.w3schools.com/css/> - Day 6 to Day 8
* JA , Start from <https://www.w3schools.com/js/> - Day 9 – Day 14
* Angular, Start from <https://angular.io/guide/understanding-angular-overview> Day 15 – Day 25.

Links For Back End

* To Be provided

Important Topic of **Angular**:

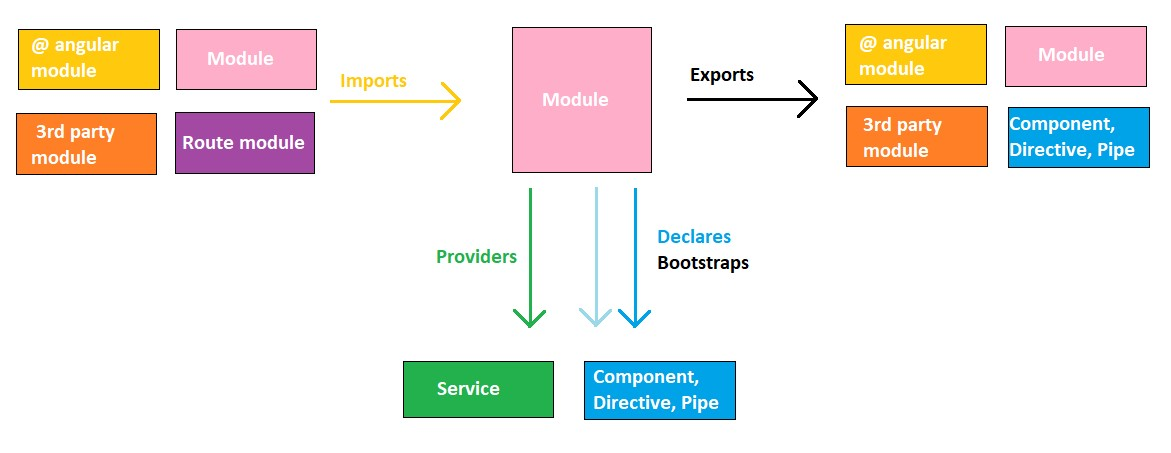
**1 Component**

Ng new c componentName

(Main building block)

**2 Module**

a way to organize an app and extend its capabilities with external libraries.



**3 Data binding**

(custom binding)

String Interpolation {{}} view/html←—------components

Property Binding[] view/html←—------components

Event Binding() view/html—------>components

Two Way Binding [(ngModel)] view/html←—------>components

**4. Directives and Life cycle**

NgClass-css

NgStyle-html

NgModel-binding to an html elements/two way bind

NgIf-conditionally create.

NgFor - repeat a node for each item in the list.

NgSwitch - switch among alternate views/or conditions

<div \*ngFor=let item of items>

Component -use with template

Structural-change the dom layout

Attribute- change the appearance of behavior

**5.Routing**

App.routing.model.ts

Added{path}

Like-

Const route:Route=[

{path: ‘ ‘,component: component Name}

];

We can pass direct RoutingLink=’ ’ with tag <a routing link=” ”/>

Dynamic Routing- shows the data directly coming from db / dynamically that time we can use dynamic routing.

**6. Parent to child communication-**

Input/OutPut/EVentEmitter

$event- the content data user types/shows into input in child template UI.

**7.Observable-**(with the help of eventemitter class publishing values from components through @output decorator.)

Asynchronous operations. (Subject / Behaviour)

Request

view====> server<—->DB/API

<====

Response

Promise

Server gathers all data once, gets then sent to cleint.

Observables

methods- Add

Closed

Error

Next

Remove

Subscribe – takes 3 arguments- next/ error/ complete.call back function(optional parameter)

Sending data into packages/Stream the data

import{observable) from ‘rxjs’;

Myobservale = new observable(observer) => { condition});

Rxjs-reactive extension library-

Observable ←—--------------------------------------------- observer

Stream data Subscribe

How to handle error→ operator-observable

Emit data

Emit error

Emit signal

Of- iterate directly all values from array

From-iterate one by one

**8.Local storage-**

Let data ={id:’’,name:’’}

locatstorage.setItem(‘session’,JSON.Stringify(data));

**9.Service-**

Ng new s service name

(provide reusability of code)

One service have multiple functions like component1,comp2,compo3

Module.ts ⇒ when provide service in app.module.ts its available for through out the app

(all components, all derivatives and all service)

**10. DI(Dependency Injection-**

Review dependency from external sources rather than itself.

Constructor(Private service :ServiceName){}

@componet

Providers:[service name]

Access with the help of this.

IMP-

**11.Lazy loading- on demand-**

Load children instead of components- which load on demand.

{path : "admin",loadChildren :()=> import('./admin/admin.module')

.then(mod=>mod.AdminModule){,

**12. Shared data between components**

@Input.

@Output.

@ViewChildren. @ViewChild(ChildComponent) child: any;

**13.Angular Life cycle:**

**-ngOnChanges()**

**-ngOnInit()**

**-ngDoCheck()**

**-ngAfterContentInit()**

**-ngAfterContentChecked()**

**-ngAfterViewInit()**

**-ngAfterViewChecked()**

**-ngOnDestroy()**

**Angular Front end-**

**Services (All)**

**main.module.tc**

**Main.routing.module.ts**

**Service.Proxy.Module**

**Backend-**

**RoleAppService.cs->IRoleAppService(role and management page)**

**->IApplicationService(identify and conversion)**

**ITransientDependency.cs(inject transient object)**

**Simplifyi3AppServiceBase.cs->base class**

**AbpServiceBase**

# Important Topics:-

Syncfusion

form io

Using Pipes to Transform Output

Making HTTP Requests

How to Authentication & Route Protection in Angular Apps

How Optimizing Apps using modules