**-- New project without much dependency installing**

ng new

**-- New project**

ng new HelloWorld

**-- When angular not installed**

npm install -g @angular/cli

**Once installed go to**

C:/user/HVuser/appdata/roaming/npm/ -- copy the path

**-- Set path variable**

once path variable is set,

--> open terminal

--> ng new <filename if needed>

**Inside the application folder**

../../helloWorld/src/

**appmodule.ts file**

only two import : system import

import { BrowserModule } from '@angular/platform-browser';

import { NgModule } from '@angular/core';

In imports we have browser module to run the programs in browser.

provider is used for services.

**appComponent.ts file**

Similar to java it has a constructor.

We can add data into the constructor.

Used for styling: styleUrls: ['./app.component.css']

Selector name and index.html file name must be the same.

Creating a new component

**ng generate** is a command: if we give **ng generate component name** it will create a component.

Data binding : used to bind the controller values to the template(html).

Diff types of data binding is:

1. Property data binding

[property] =”value”

Eg:

[firstname]= “sruthi”

1. Event data binding :it represtents an event

(ngClick)=”sayHello()”

ngChange=”show()”

1. Two way data binding

Reflects changes automatically in all places as model and view

[(ngModel)]=”Sruthi”

**Routing:**

This method is helpful for users to direct from different pages upon the option they selected

Every program starts with app.modules.ts.

So route import should be made in this file

//import for router

import {RouterModule,Routes}from '@angular/router';

then give RouterModule in imports

imports: [

BrowserModule,RouterModule,FormsModule

],

Here we can set path for the pages for the clicks in two ways.

1.

const appRoutes : Routes=[

{path:'Calc',component: CalcComponent},

{path:'Name',component: NameDropdownComponent},

{path:'Login',component: LoginComponent},

{path:'Stud',component: StudentComponent}

]

Where calc is a component. Then

imports: [

BrowserModule,RouterModule.forRoot(aooRoute),FormsModule

],

**2.**

**To directly** give the path.

imports: [

BrowserModule,FormsModule,

RouterModule.forRoot ([

{path: 'First',component:FirstComponentComponent},

{path: 'Second',component:SecondComponentComponent},

{path: 'Third', component:ThirdComponentComponent},

{path: 'Menu',component:MenuComponent},

{path:'Login',component:LoginComponent},

{path:'',component:ConcatenationComponent}

])

],

@Injectable():

@injectable() Is a decorator is needed when a service needs dependencies injected to its constructor .Its optional when a service does not need dependencies injected into it, but it is a common practice to use @injectable() decorator for all services.

It is defined at class level

@promises:

@observable: similar to promise. But efficient.

Handles data asynchronously and can handle multiple data

In HTTP services, we call the existing services which are implemented in another programming language, and invoke them by using angular.

Http service returns promises which contains 2 values as:

1. resolve: comes when the service is successfully served
2. reject : comes when the service access fails

Drawbacks with Promises are, they will return only one value at a time, but during service requests data to be accessed asynchronously, then we use observables instead of promises

Assume we have to return list of users from the given service,

To store the list of users returned by the service then we need the help of Observable array.

Observable<User[ ]>

To call the http Service initialize the object of http in the constructor first.

Constructor(private http : Http) {

getUsers() : Obsevable<User[]> {

return this.http.grt(<https://jsonplaceholder.typicode.com/users).map(res> ; Response ) => res.json()).catch(“

}