Angular :

<https://angular.io/guide/cheatsheet>

<http://www.dotnetcurry.com/angular/1385/angular-4-cheat-sheet> bvn

https://www.tutorialspoint.com/angular4/angular4\_quick\_guide.htm

<https://programmingwithmosh.com/angular/angular-4-tutorial/>

Angular 4 best quick guide:

<http://www.dotnetcurry.com/angular/1385/angular-4-cheat-sheet> god bvn

<https://github.com/junaid1460/angular-4-cheatsheet> bvn

<https://angular.io/guide/cheatsheet>

How to install node package management: **install npm**

How to install angular CLI tool: **npm install -g @angular/cli**

How to install bootstrap module: **npm install –save bootstrap**

How to run angularjs application: **ng serve**

How to create a new component in AngularJs application through CLI: **ng create component myComp** OR **ng g c myComp**

How to bind external value with a variable in .ts file: **@Input() myVar: string;** and in template we will have **[myVar]=”actualValue”**

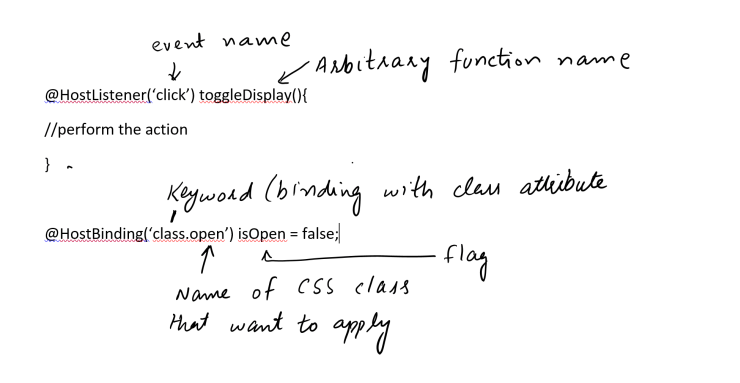
How to emit and event: **@Output() myEventEmitter = new EventEmitter<type>();** and then in a function we can emit: **myEventEmitter.emit()** or **myEventEmitter.emit(data)**

@HostListener(‘click’) toggleDisplay(){

//perform the action

}

@HostBinding and @HostListener



Services

Services are helpful to write clean/lean code. They can hold the data which is available to all components using that service.

How to use a service:

1. Create a server
   1. **export class UserService {}** //this could go a in a file called users.service.ts
2. How to use a Service
   1. Import the service: **import { UserService } from ‘..path..to..file/user.service.ts’;**
   2. You need to provide the service class name in “**Providers: [UserService]**” array of the module which wants to use this service.
   3. Inject the service in constructor of the component **constructor(private userService: UserService) {}**
3. Scope of a service
   1. **Global:** put a service in **Providers: []** of app.module.ts and same instance of the service will be available for all components (Important: to use the same instance the component should not have the service in its Providers array)
   2. **Local:** to use a service just in component put the service in Providers array of component and component will get its own instance of that service.
   3. **Only Components:** if you put service name in Providers of app.component.ts then same instance will be available for all components unless a components mentions the service in its Providers array.