**Lesson30 Angular Injector**

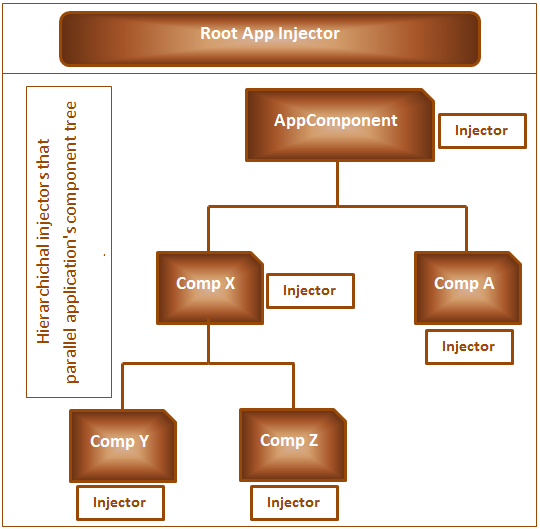
**Notes:-**

**1-Dependency Injector can be applied at different levels**

**A-Component Level**

**B-App. module Level**

**C-Root App Level (when we apply multiple modules at the same app.module)**



@Component({

providers:[TestServiceService]

})

@NgModule({

imports: [

providers: [EmployeeService]])

**1-In the Root App Level the instance is shared across all the components**

**2-In the Component App. Module level the instance is shared across all the components except the lazy loading component**

**3-In the Component level the instance is shared on the same children and the child components**

**4-The priority is begin from the**

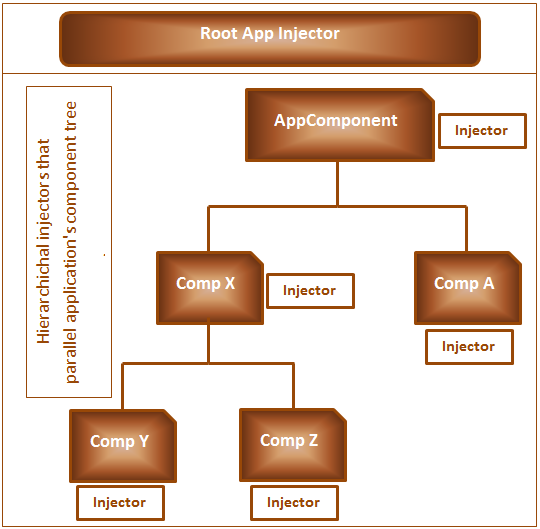
**Component level 🡺 app.component level 🡺 Root App Level**

**(This means that the component ask if the providers contain the service providers, if not exist it will going to the parent component and do the same thing, if not exist it will go to the Root App Injector)**

**5-when we apply the providers at the component level, we see that the instance from Comp A is different from the comp X, but the instance is the same in the Comp X, Comp Y, Comp Z.**

**Apply Multiple Component inside the app.module**

**1-you can inject multiple Modules inside the app.module , such as we create the Employee Module and the department Module and insert it to the app.module.**



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

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

import { TestServiceService } from './Service/test-service.service';

@NgModule({

})

export class EmpModule { }

**In the app.module.ts we will type the following :-**

import {EmpModule} from './EmpModule';

import { TestServiceService } from './Service/test-service.service';

@NgModule({

declarations: [

Home2Component],

imports: [

RouterModule.forRoot(appRoutes),

EmpModule

],

providers:[]

, bootstrap: [AppComponent]

})

export class AppModule { }