Lab assignment Angular :

Angular service

Angular services are task-based. Services are reusable code which shared among different components. Angular provides built-in(HTTP, router, animation, form, UI Library,) and custom services. services work on dependency injection mechanism.
Services may be written by just defining a typescript(with extension .service.ts) class with decorator @Injectable annotation. Providers are used to declaring or injecting classes, functions, and values by which they may be used by the dependency injection mechanism.
Services may be injected in different components, modules, and other services.

1.creating custom service:

We need to crete a service with array of data containing hobbies and an method
hobbies = [
    'dancing',
    'singing',
    'internet'
  ];

servicemethod(){
     return 'Its just a simple service method';
  }

Inject this service to component and test it.

Code:

Step 1: Write the following code into service.ts

```
import { Injectable } from '@angular/core';

@Injectable()
export class DataService {
 // Create array
 hobbies = [
  'dancing',
  'singing',
  'internet'
 ];
 constructor() { }
// Create simple angular service method
 servicemethod(){
   return 'Its just a simple service method';
 }

}
```

Step 2: Update app.module.ts (Pass the reference to newly created service and add providers)

```
// Import reference of service
import { DataService } from './data.service';
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { HttpModule } from '@angular/http';


import { AppComponent } from './app.component';
```

```typescript
@NgModule({
  declarations: [
    AppComponent,
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule
  ],
  // Add provider
  providers: [DataService],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Step3 : Update component.ts as following code

```typescript
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  template: `
  <p>hi {{data}}</p>
  <li *ngFor="let h of itemsarray">
  <b> {{h}} </b>
</li>
  `,
```