Angular Services

When developing an Angular app, you will most likely run into a scenario in which you need to use the same code across multiple components. You may even need to share data between components, or you may need to fetch data from a database of some sort.

It's these times when creating an Angular service makes sense. An angular service is simply a function that allows you to access its' defined properties and methods. It also helps keep your coding organized.

Components shouldn't fetch or save data directly and they certainly shouldn't knowingly present fake data. They should focus on presenting data and delegate data access to a service.

Services are a great way to share information among classes that *don't know each other*

Angular Dependency Injection

Dependency Injection (DI) is a way to create objects that depend upon other objects. A Dependency Injection system supplies the dependent objects (called the *dependencies*) when it creates an instance of an object.

An Angular injector is responsible for creating service instances and injecting them into classes

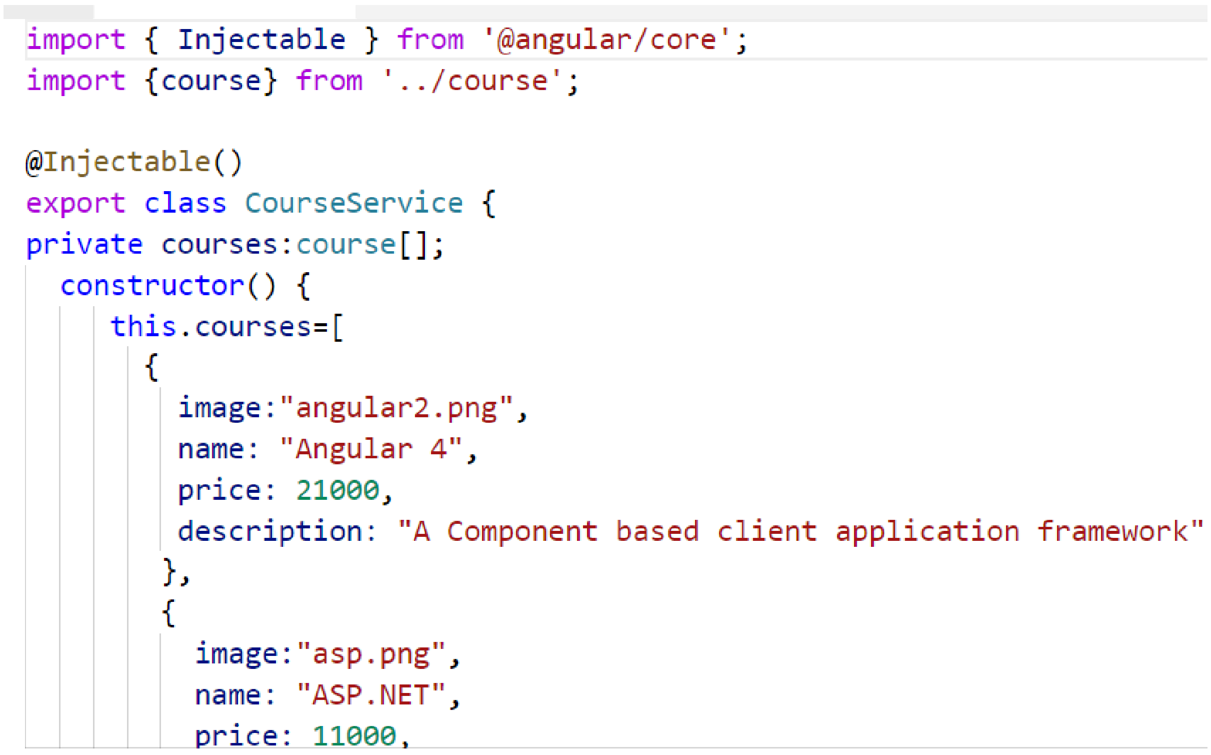
Angular doesn't automatically know how you want to create instances of your services or the injector to create your service. You must configure it by specifying providers for every service.

Providers tell the injector *how to create the service*. Without a provider, the injector would not know that it is responsible for injecting the service nor be able to create the service.

Angular doesn't automatically know how you want to create instances of your services or the injector to create your service. You must configure it by specifying providers for every service.

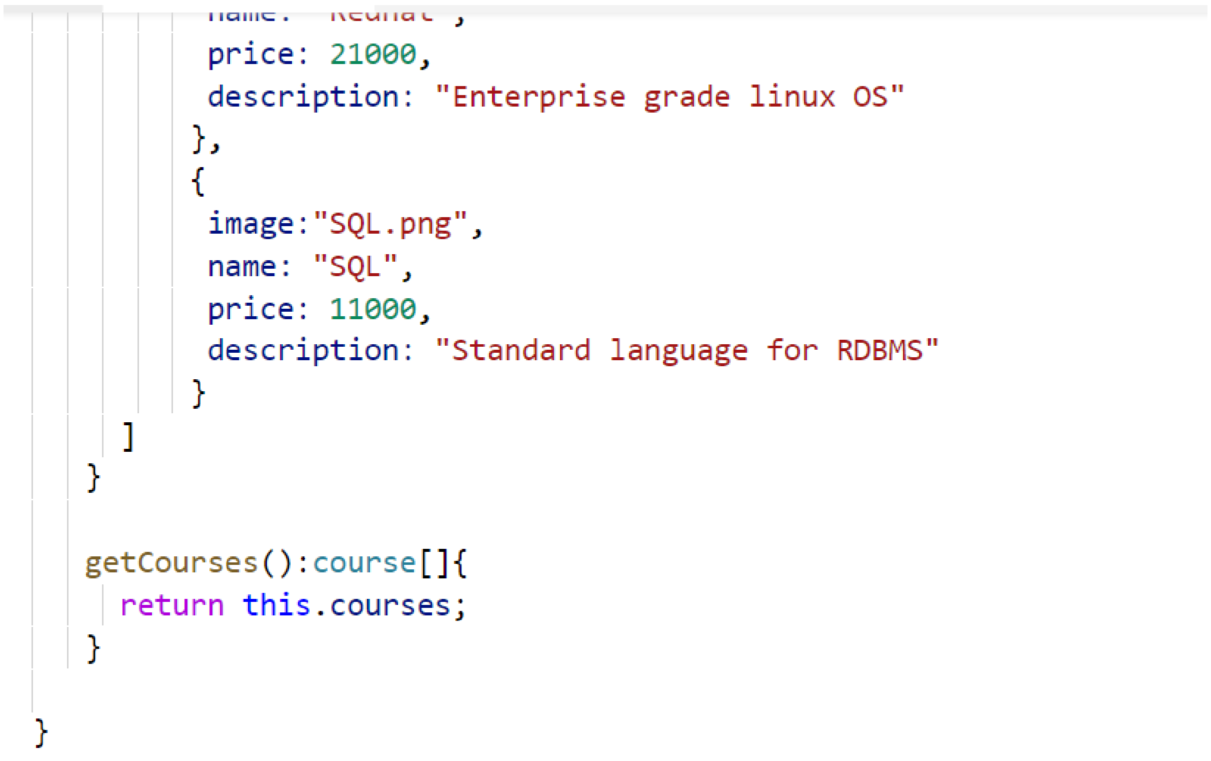
Providers tell the injector *how to create the service*. Without a provider, the injector would not know that it is responsible for injecting the service nor be able to create the service.

In Angular every service you create has a @Injector decorator, When you configure services at the module level it is singleton in nature.



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

@Injectable()
export class CourseService {
private courses:course[];
  constructor() {
     this.courses=[
       {
        image:"angular2.png",
        name: "Angular 4",
        price: 21000,
        description: "A Component based client application framework"
       },
       {
         image:"asp.png",
         name: "ASP.NET",
         price: 11000,
```



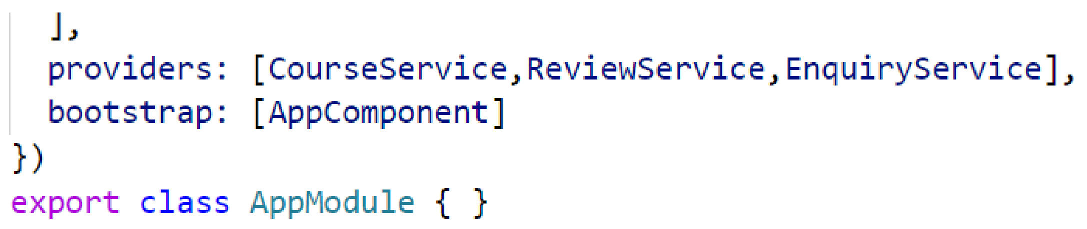
```
          name: "Redhat",
          price: 21000,
          description: "Enterprise grade linux OS"
        },
        {
          image:"SQL.png",
          name: "SQL",
          price: 11000,
          description: "Standard language for RDBMS"
        }
    ]
  }

  getCourses():course[]{
    return this.courses;
  }

}
```

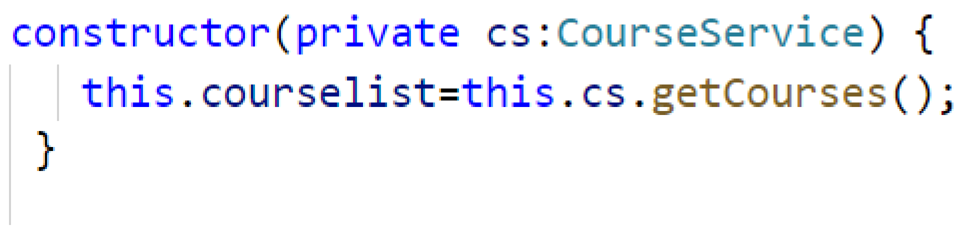
In the above service, we have a @Injector decorator which lets the object of the service to be injected and the class has a method called getCourses which returns course array.

Now we provide the service to all the components by configuring it in app.module .ts as shown in the following picture.



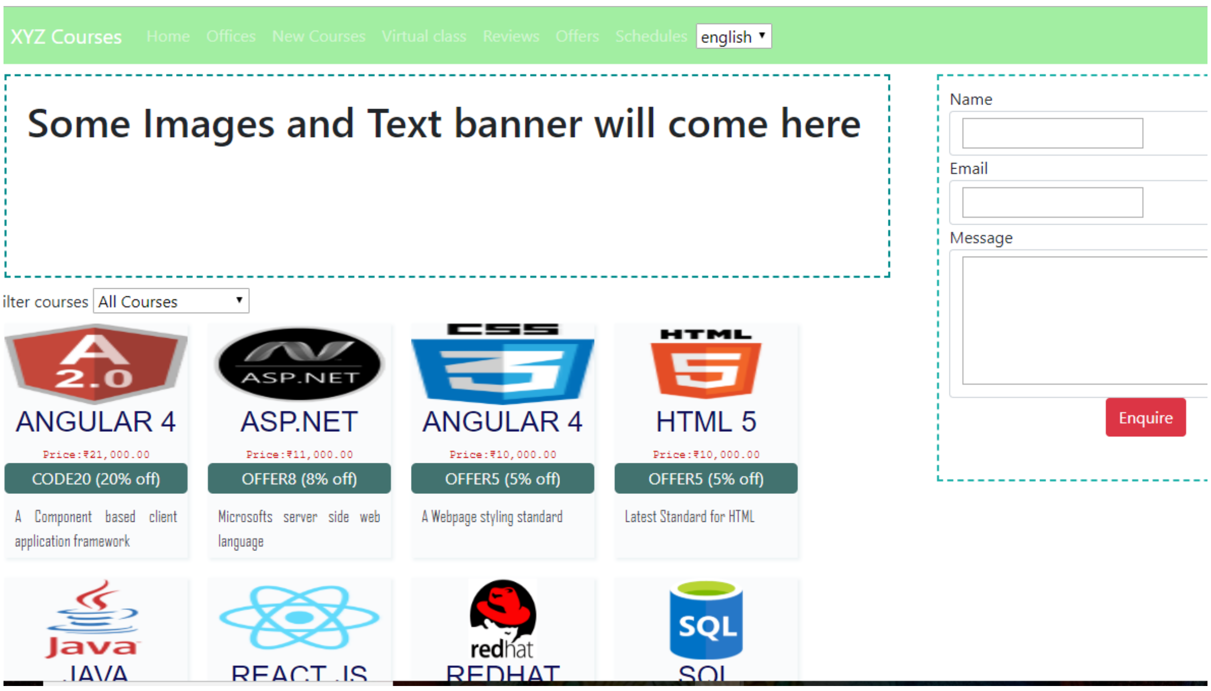
```
  ],
  providers: [CourseService,ReviewService,EnquiryService],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Now in any component, where we want the service , you can get it injected by using the below syntax



```
constructor(private cs:CourseService) {
    this.courselist=this.cs.getCourses();
 }
```

Object of course service gets injected and we can see the data provided by the service is rendered in the component



XYZ Courses Home Offices New Courses Virtual class Reviews Offers Schedules english

Some Images and Text banner will come here

Name

Email

Message

Enquire

ilter courses All Courses

ANGULAR 4
Price:₹21,000.00
CODE20 (20% off)
A Component based client application framework

ASP.NET
Price:₹11,000.00
OFFER8 (8% off)
Microsofts server side web language

ANGULAR 4
Price:₹10,000.00
OFFER5 (5% off)
A Webpage styling standard

HTML 5
Price:₹10,000.00
OFFER5 (5% off)
Latest Standard for HTML

JAVA

REACT JS

REDHAT

SQL