



BTI425

Web Programming for Apps and Services

[Notes](#)[Weekly](#)[Resources](#)[Graded work](#)[Policies](#)[Standards](#)[Professors & addendum](#)[Code examples](#)

A brief introduction to an Angular service

This document will enable you to get started with a simple Angular service. Later in the course, we will return to this topic and cover more detail.

Angular “service”

In Angular, a *service* is a class that provides value and functionality to your app’s components. You can think of this as a technique for writing code that can be used in MANY of your app’s components. Often, a service is a *data provider*, which can manage interaction with a web API.

Read the Angular documentation topic about an [introduction to services](#).

Generate a service

The Angular CLI has a *service generator*. For example:

```
ng g s DataManager --flat -S
```

The generator will create code like this:

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

@Injectable({
  providedIn: 'root'
})
export class DataManagerService {

  constructor() { }
}
```

Angular uses a software design pattern named *dependency injection* to enable a service to be used in a component. That explains the `@Injectable` decorator.

More info about dependency injection is [here](#).

What code is added to a service? We add:

- data-storing properties if needed, and
- methods that do things.

Use a service

In any component class, a service can be used. We must always do two coding tasks to make this happen.

First, import the service:

```
import { DataManagerService } from './data-manager.service';
```

Second, update the constructor signature, by adding an argument:

```
constructor(private m: DataManagerService) {
```

Then, in the component's class code, `m` (for “manager”) is a reference to the service. When Angular creates your component's instance in memory, it also creates the service instance in memory (if it's not there already), and “injects” the service reference into the component.

Can I see this in action?

Yes, complete the work in the Week 7 non-graded “[in-class](#)” exercise.

In a later class, we will cover services in more depth.

© 2020 - Seneca School of ICT