




Enterprise Software Development

Join the Conversation #AngularHackDay #Angular @william_dotnet

A photograph of a person's hand hovering over a cup of coffee on a wooden table. The hand is positioned as if about to pick up the cup. The coffee has a heart-shaped latte art. A small potted plant is visible on the left side of the table. The background is slightly blurred, showing a person's legs and a window.

Services & Dependency Injection Angular 8

William Liebenberg [SSW]

Join the Conversation #AngularHackDay #Angular @william_dotnet



William Liebenberg

SSW Solution Architect



@William_DotNet



azuregems.io



/william-liebenberg

Join the Conversation #AngularHackDay #Angular @william_dotnet

Agenda

Services

Dependency Injection

Demo

Services



Encapsulates functionality

Usually concerned with communication

- such as HTTP, JSON, Text, Hardcoded

Removes complexity from components





Make a Service

Use the Angular CLI

```
ng generate service <path>
```

```
ng g s <path>
```

E.g.

```
ng generate service services/joke
```

```
ng g s services/joke
```

Make a Service using the CLI



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

2: powershell ▼



Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

```
PS C:\Code\Hackday\gabe-norris-app> ng generate service services/joke
```

Make a Service using the CLI



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

2: powershell



Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

```
PS C:\Code\Hackday\gabe-norris-app> ng generate service services/joke
```

```
CREATE src/app/services/joke.service.spec.ts (323 bytes)
```

```
CREATE src/app/services/joke.service.ts (133 bytes)
```

```
PS C:\Code\Hackday\gabe-norris-app> 
```


Is this a Joke?



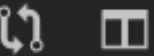
TS *joke.service.ts* x

```
1  import { Injectable } from '@angular/core';
2
3  @Injectable({
4    providedIn: 'root'
5  })
6  export class JokeService {
7
8    constructor() { }
9  }
10 |
```

Implementing a service



TS joke.service.ts ●



```
1  import { Injectable } from '@angular/core';
2
3  @Injectable({
4    providedIn: 'root'
5  })
6  export class JokeService {
7
8    constructor() { }
9
10   public fetchJoke(firstName: string, lastName: string, category: string): Observable<Joke> {
11     // Magic 🧙 and Unicorns 🦄
12
13     // Yes you can code with emoji's!!! 💪⚡️🦄
14   }
15 }
16 |
```



I have a Service... now what?



Dependency Injection (DI)

Components Depend on Services

Services are Injected into Components

Constructor Injection

TS app.component.ts ✕

```
1  import { Component } from '@angular/core';
2  import { JokesService } from '../services/jokes.service';
3  import { Joke } from '../models/Jokes';
4
5  @Component({
6    selector: 'app-root',
7    templateUrl: '../app.component.html',
8    styleUrls: ['../app.component.scss']
9  })
10 export class AppComponent {
11
12     title = 'Gabe Norris\'s Jokes Generator';
13
14     firstName: string;
15     lastName: string;
16     category = 'nerdy';
17
18     latestJoke: Joke;
19
20     constructor(private jokesService: JokesService) {
21     }
22 }
```



Consuming a Service



```
20 constructor(private jokesService: JokesService) {
21 }
22
23 generateJoke() {
24     this.jokesService.fetchJoke(this.firstName, this.lastName, this.category)
25         .subscribe(
26             data => {
27                 if (data !== null) {
28                     this.latestJoke = data;
29                 }
30             },
31             error => {
32                 alert(error);
33             });
34 }
35 }
36
```


Dependency Injection (DI)



Angular built-in DI Framework

- Modularity
- Reusability
- Testability

DEMO

Join the Conversation #AngularHackDay #Angular @william_dotnet



Thank you!

williamliebenberg@ssw.com.au

www.ssw.com.au

Sydney | Melbourne | Brisbane

Questions?

Join the Conversation #AngularHackDay #Angular @william_dotnet