Angular 2 dependency injection in pipes

Asked 3 years, 5 months ago   Active 3 months ago   Viewed 17k times

▲

20

▼

★

1

How can i inject dependencies like a service into angular2 pipes?

```typescript
import {Pipe, PipeTransform} from 'angular2/core';
import {MyService} from './service';

//How i am injecting MyService to the pipe?

@Pipe({name: 'exponentialStrength'})
export class ExponentialStrengthPipe implements PipeTransform {
  transform(value:number, args:string[]) : any {
    return Math.pow(value, parseInt(args[0] || '1', 10));
  }
}
```
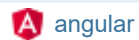
typescript   Ⓐ angular

edited Feb 8 '16 at 5:01   asked Feb 7 '16 at 18:47

SnareChops
9,706   6   57   84

user233232
2,182   6   17   30

1 Answer

▲

44

▼

◢

You can inject the dependency in the constructor like this:

```typescript
export class ExponentialStrengthPipe implements PipeTransform {

  constructor(public testService: TestService) {

  }
```

```
        return Math.pow(value, parseInt(args[0] || '1', 10));
    }
  }
```

Don't forget to make sure you add this dependency to the app module:

```
@NgModule({
    declarations: [...],
    imports: [...],
    providers: [..., TestService],
    bootstrap: [AppComponent],
})
```

edited Apr 17 at 11:29   answered Feb 7 '16 at 19:01

Liam   JS   Bazinga
17.1k   16   80   133   6,774   3   26   49

Thanks! Do you know if there is a way to inject `TestService` for only the Pipe instead of the whole application? e.g. With a `providers` array. – maximedupre Apr 3 '16 at 21:39 ✏️

I don't think it's possible. The pipe decorator only except name and pure properties. – Bazinga Apr 4 '16 at 12:49

---

1   @maximedupre you could use an `NgModule` that only declares your pipe. The `TestService` would be provided in that module only, thus the `TestService` is only available to the pipe and not the whole application. – Philip Bulley Sep 29 '16 at 16:03

I am having an issue with this because I have a service that consumes the pipe as well (the service injected into the pipe is the route to decorate with the url). So In this case Typescript throws an error if I don't inject the ActivatedRoute into my service and pass it through `private prefPipe: PagePreferencesPipe = new PagePreferencesPipe(routerFromService);` Is there a way I can avoid this by injecting into the service or something? – Jackie Feb 23 '17 at 14:43

---

how do you unit test this? – danday74 Oct 24 '18 at 14:01

Got a question that you can't ask on public Stack Overflow? Learn more about sharing private information with Stack Overflow for Teams.   ✕