Java config

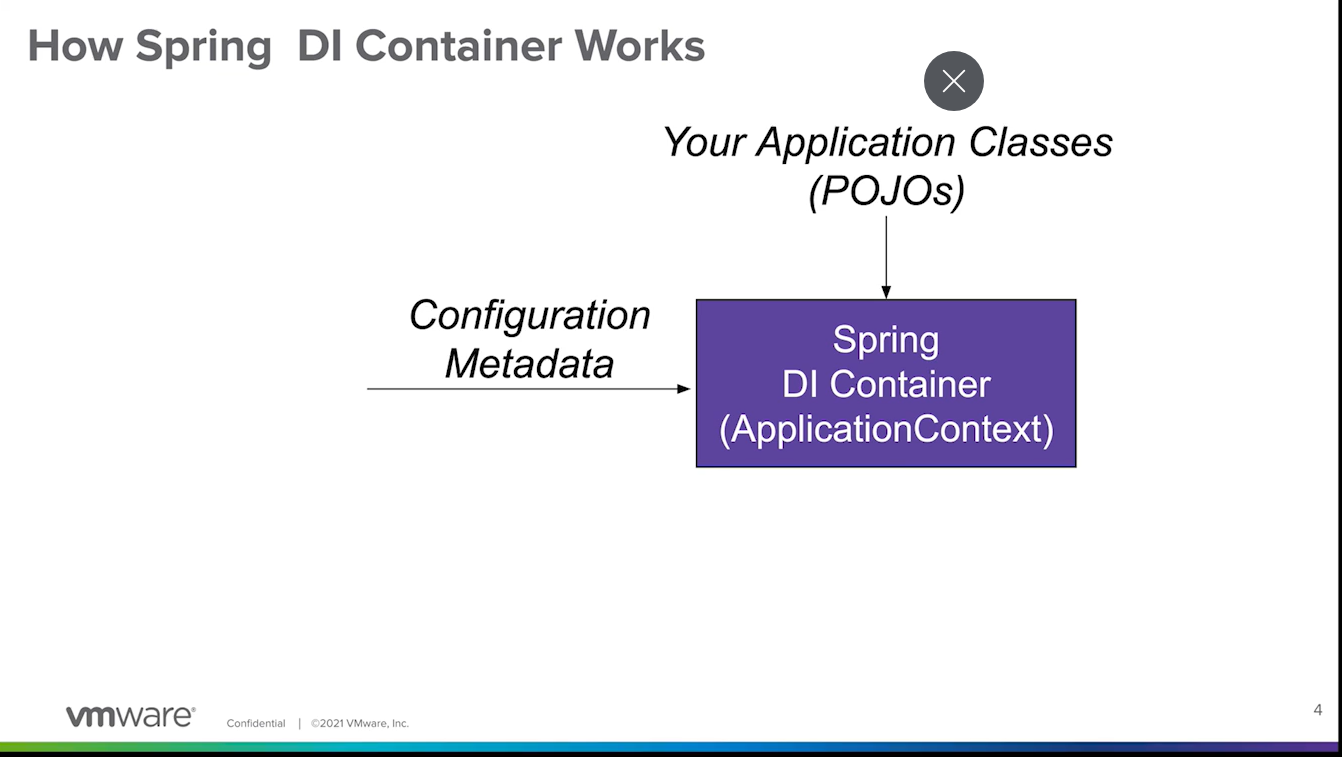
Define spring beans using java code

Access beans in the app context

Handle multiple config files

Handle dependencies between beans

Explain and define bean scopes



A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

First one needs casting!

The bean that doesn’t have any dependencies will be the first one to be constructed – in the above it is datasource.!

A diagram of a application context

Description automatically generated

Spring app context can be created in

Standalone app

Web application

Test class

A screenshot of a computer

Description automatically generated

Handle multiple configs

A screenshot of a computer

Description automatically generated

A close-up of a yellow card

Description automatically generated

A close-up of a yellow card

Description automatically generated



Another way is

A close-up of a computer code

Description automatically generated

Bean Scope

@Bean

@Scope(“singleton”) 🡪 default bean scope

Singleton has a prob with multiple threads

* Immutable
* Synchronized
* Use a different scope

@Bean

@Scope(“prototype”)

New instance created every time bean is requested

Session –

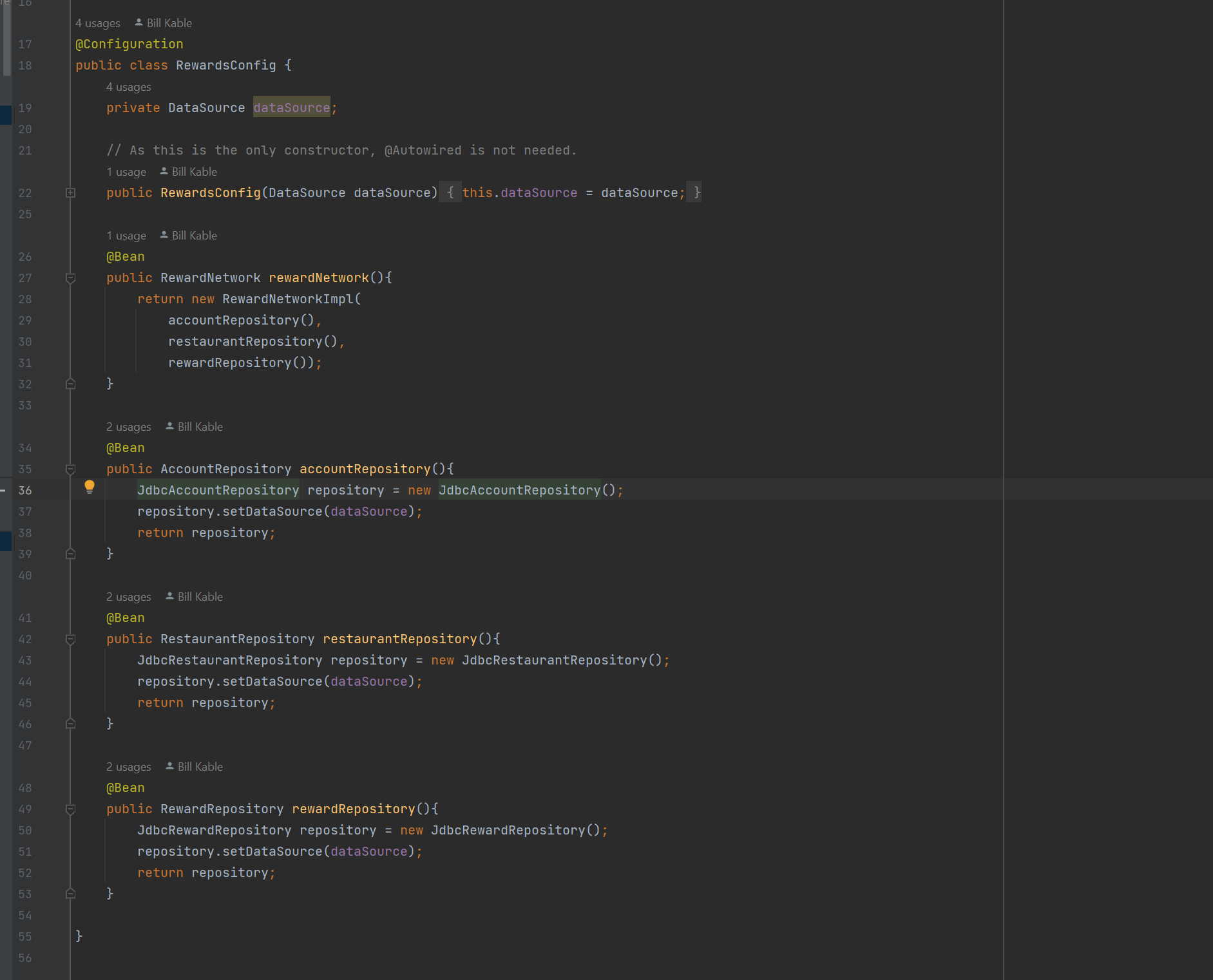
Request –

Got other scopes

Got this project

12-javaconfig-dependency-injection-solution

1. There is a configuration class



1. The DataSource bean is coming from somewhere. Where?
2. From here

A screen shot of a computer

Description automatically generated

This datasource refers to a db and tables

As per below

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Embedded database builder builds an embedded database this is good for testing

Now the TestConfiguration class

Has an import for RewardsConfig

Which means, this TestInfrastructureConfig class can be used as a upper level config class when defining an application context to be created

This is done in the RewardNetworkTests class

A screenshot of a computer program

Description automatically generated

The context will contain

* A datasource
* A rewardNetwork
* An accountRepository
* A restaurantRepository
* A RewardRepository

You can retrieve all of these

A screenshot of a computer

Description automatically generated

Now the test can be executed

A screenshot of a computer program

Description automatically generated

The project is in



Play around with it.