Spring - Feb 23

Bean:

- scope: singleton(default)
    - prototype

@Scope(”prototype”) // change scope

Annotation:

~~In Spring Framework annotation-based configuration instead of using XML for describing the bean wiring~~, it is done by using annotations on the relevant class, method or the field declaration.

@Component

When the spring container initializes, it searches the root and sub folders that are defined at base-package, instantiating the bean which has the @Component annotation.

@Autowired

The @Autowired annotation can be used with fields or methods for injecting a bean by type.

@Configuration

When spring boots scanning the classes and it will see this configuration annotation it will treat this class it will treat this class as a configuration part of the configuration is no longer part of the runtime run objects.

XML is too old.

AOP (**Aspect-Oriented Programming**) (proxy design pattern)

the code that automatically run we call it aspect

- as long as the code execution meeting particular criteria, then this aspect will be triggered. That aspect could be a security check.

e.g

I → buy fast food → MC

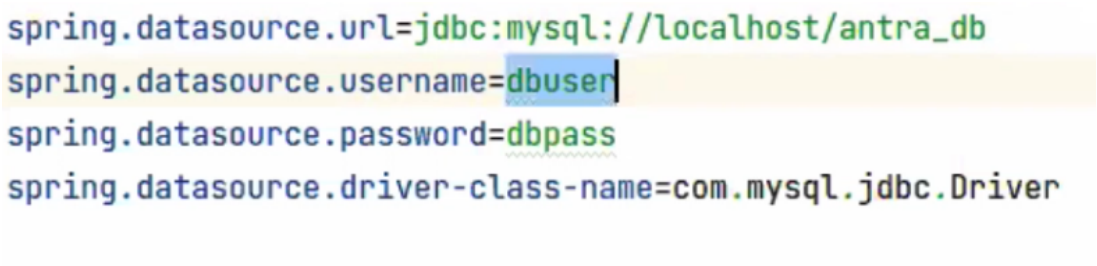
1. I → call my mom → approved
2. MC → call my mom → approved (AOP)

Datasource:

Spring boot will automatically config datasource, when initiates.

if not, will have an error ‘url’ attribute is not specified

1. set source in application.properties



2. set exclude

```
@SpringBootApplication(exclude = {DataSourceAutoConfiguration.class})
public class AssignmentApplication {}
```

Spring Boot

Material:

Why do we choose Spring?

- **Versatile**: We can integrate any technologies with spring, such as hibernate, structs, etc.
- **Modularity:** We've options to use the entire Spring framework or just the modules necessary. Moreover, we can optionally include one or more Spring projects depending upon the need.
- **End to End Development :** Spring supports all aspects of application development, Web aspects, Business aspects, Persistence aspects, etc, so we can develop a complete application using spring.

How do you provide configuration metadata to the Spring Container?

XML based configuration file <bean>

Annotation-based configuration: @Component, @Autowired

Java-based configuration: @Configuration and @Bean