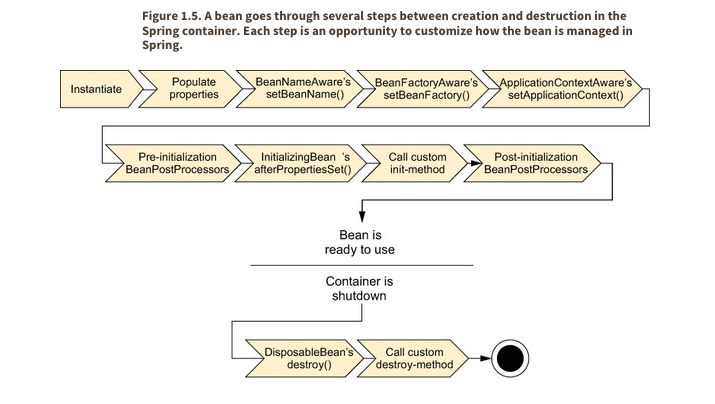
# Spring Concepts

## Spring Bean Life Cycle

* Instantiates
* Populates properties: injects values and bean reference into bean properties
* BeanNameAware 🡪 setBeanName
* BeanFactoryAware🡪setBeanFactory
* ApplicationContextAware🡪setApplicationContext
* Pre-initialization of BeanPostProcessors
  + Annotated methods of afterPropertySet
  + Call custom init methods
* Post-initialization of BeanPostProcessors
* READY TO USE
* DisposableBean 🡪destroy
* Custom destroy methods



**Q) What is use of @import annotation?**

Used to import other configuration information into main Configuration

**Q) How do you import xml based configuration into Java based Configuration?**

@ImportResource annotation is used to import xml file into java configuration

**Q) How to configurare embeddde databaes in spring?**

@Bean(destroyMethod="shutdown")

public DataSource dataSource() {

return new EmbeddedDatabaseBuilder() .addScript("classpath:schema.sql") .addScript("classpath:test-data.sql") .build();

}

**Q) How do you activate profiles in Spring Framework?**

Spring comes with two different properies to activate profiles

1. First one is **spring.prifles.active=dev**
2. Second one is **spring.profiles.default=dev**

**Q) How many ways we can activate profiles?**

1) Intialization properies to the DispatchServlet

2) Context paramter of web application

3) Environment properties/ JNDI entries

4) JVM Argument

4) @ActiveProfiles annotation on integration test class

**Q) How do we activate profile in web.xml?**

<context-param>

<param-name>spring.profiles.default</param-name>

</param-value>dev</param-value>

</context-param>

**Q) How to achieve Conditional Configuration in spring 4?**

Using @Conditional annotation and Condition interface we can achive this

**Q) What is the Advantage of @Primary annotation?**

This annotation used to choose one among multiple annotations. This can be applied along side of @Component annotation.

**Q) What is the advantage of @Qualifier annotation?**

This is works along with @Autowired or @Inject annotation to select from multuple bean implementations

**Q) How do you create Session Scope bean?**

@Component

@Scope (value=WebApplicationContext.SCOPE\_SESSION, proxyMode=ScopedProxyMode.INTERFACES) // create proxy and implement interface

public ShoppingCart cart() { ... }

**Q) When you inject SessionScope bean into Singleton bean how does it works?**

Using attribute proxyMode it injects proxy instead of actual Bean. In the runtime it will delegates calls to actual sessionScoped bean. This is main motton of Proxy Design Pattern.

The **proxyMode**is set to **ScopedProxyMode**.**INTERFACES**, indicating that the proxy should implement the **ShoppingCart**interface and delegate to the implementation bean.

NOTE: this works only if you are injected Interface to singleton bean

Create a JDK dynamic proxy implementing *all* interfaces exposed by the class of the target object.

**Q) How do you inject Concrete Implemenation of SessionScoped bean class into Singleton Bean?**

If the bean type is a concrete class, you must set **proxyMode**to **ScopedProxyMode.TARGET\_CLASS** to indicate that the proxy should be generated as an extension of the target class.

**NOTE: It Uses CGLib to generate extension class**

Create a class-based proxy (uses CGLIB).

**Q) How do you inject properties dynamically using SpEL?**

Using spring expression language we can inject properties dynamically.

@Value(#{“T(System).getCurrenttimemillis()”})

Using this we can call Static methods of any class

**Q) What are cross cutting concerns?**

Functions that are span into multiple points of an application are called cross cutting concerns. These are conceptually separate from the actual business logic implementations.

Ex: Logging, security, caching and transaction management

Cross cutting concerns modularizd into special classes called **aspects**.

**Q) Define AOP Terminology?**

Advice: Jo b of an Aspect is called Advice. It is fall into five main flavours. Person who measures electicity is an example of Advice.

1. Before Advice:
2. After Advice:
3. Around Advice:
4. After returning Advice:
5. After throwing Advice:

**Q) Define Joinpoint?**

Join Point is a point in the application where aspect can be executed. Ectricity measurement meters on each house an example of joinpoint. In an application we can say all the methods are joinpoints.

**Q) Define pointcut?**

Pointcuts are the points where one or more jointpoints invoked.

Ex: Person can visit few of the houses to measure electricity. For this person those houses are pointcuts.