

USO DE
@POSTCONSTRUCT
Y @PREDESTROY



CICLO DE VIDA DE UN BEAN

- ▶ Ya hemos visto cómo manejar el ciclo de vida vía XML.

```
<beans xmlns="http://www.springframework.org/schema/beans"...>  
  <bean id="personaDao"  
    class="com.openwebinars.lifecycle.PersonaDAOImplMemory"  
    init-method="init" destroy-method="destroy" />  
</beans>
```

```
public class PersonaDAOImplMemory implements PersonaDAO {  
  
    public void init() throws Exception { ... }  
    public void destroy() throws Exception { ... }  
  
}
```

CICLO DE VIDA DE UN BEAN

- Podemos usar las anotaciones
@PostConstruct y **@PreDestroy**

```
<beans xmlns="http://www.springframework.org/schema/beans"...>  
  <bean id="personaDao"  
    class="com.openwebinars.lifecycle.PersonaDAOImplMemory"/>  
</beans>
```

```
public class PersonaDAOImplMemory implements PersonaDAO {  
  
  @PostConstruct  
  public void init() { ... }  
  @PreDestroy  
  public void destroy() { ... }  
  
}
```