



**Universidad Nacional Autónoma de
México**

**Dirección General de Cómputo de
Tecnologías de Información y
Comunicación**

Diplomado

Desarrollo de Sistemas con Tecnología Java

Ejercicio Diez

“Alcance de un Bean (scope)”

Mtro. ISC. Miguel Ángel Sánchez Hernández

Tabla de contenido

1. Copiar un proyecto en IntelliJ IDEA	3
2. Archivo de configuración bean-configuration.xml	3
3. Clase Inicio	4
4. Archivo de configuración bean-configuration.xml	5

1. Copiar un proyecto en IntelliJ IDEA

Para copiar un proyecto, hacemos lo siguiente:

1. Señalamos el proyecto spring-core-factory
2. Apretamos Ctrl + C
3. Luego Ctrl + V
4. Project Name: spring-core-scope
5. En el archivo pom.xml cambiar las siguientes líneas
 - `<artifactId>spring-core-factory</artifactId>` por lo siguiente
 - `<artifactId>spring-core-scope</artifactId>`
 - Cambiar las etiquetas `<name>` con
 - `<name> spring-core-scope </name>`
 - Cambiar la etiqueta `<description>` con
 - `<description> Ejemplo de scope con Spring </description>`
6. Actualizar maven

2. Archivo de configuración bean-configuration.xml

Modificamos el archivo bean-configuration.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">
  <!-- bean definitions here -->
  <bean id="persona" class="dgtic.core.modelo.Persona"
        scope="singleton">
    <property name="nombre" value="Pedro" />
    <property name="edad" value="27" />
  </bean>
</beans>
```

3. Clase Inicio

Modificamos la clase Inicio como sigue:

```
package dgtic.core.inicio;
import dgtic.core.modelo.Persona;
import dgtic.core.modelo.TiposCarro;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Inicio {
    public static void main(String[] args) throws IllegalAccessException {
        ApplicationContext contexto = new ClassPathXmlApplicationContext(
            new String[] { "/src/main/java/dgtic/core/xml/bean-configuration.xml",
                "/src/main/java/dgtic/core/xml/bean-services.xml" });
        Persona persona = (Persona) contexto.getBean("persona");
        persona.getCoche().getModeloCoche(TiposCarro.DEPORTIVO).crear();
        System.out.println(persona.toString());
        System.out.println("-----");
        Persona personaDos = (Persona) contexto.getBean("persona");
        personaDos.setNombre("Tomas");
        personaDos.getCoche().getModeloCoche(TiposCarro.FAMILIAR).crear();
        System.out.println(personaDos.toString());
        System.out.println("-----");
        System.out.println(persona.toString());
        System.out.println("-----");
        ((ClassPathXmlApplicationContext) contexto).close();
    }
}
```

Correar la aplicación. Como Empleado es singleton, si cambiamos un valor del bean, se refleja en las demás instancias que lo ocupen.

4. Archivo de configuración bean-configuration.xml

Modificamos el archivo bean-configuration.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">
  <!-- bean definitions here -->
  <bean id="persona" class="dgtic.core.modelo.Persona"
        scope="prototype">
    <property name="nombre" value="Pedro" />
    <property name="edad" value="27" />
  </bean>
</beans>
```

Correr la aplicación. Cada bean Empleado tiene su propia instancia.