Scopes Of Bean

Scenario: Now let's try to create two bean of same class and print the object

MyApp5.java

```
package com.tapacad.spring;

import
org.springframework.context.support.ClassPathXmlApplicationCo
ntext;

public class MyApp5 {

	public static void main(String[] args) {
//		Load Application context
		ClassPathXmlApplicationContext context =
				new
ClassPathXmlApplicationContext("applicationContext.xml");

//		Get bean
		Car car1 = (Audi)context.getBean("audi");
		Car car2 = (Audi)context.getBean("audi");

//		print object reference of car1 and car2()
		System.out.println(car1);
		System.out.println(car2);

//		close context
		context.close();

	}

}
```

Output:

```
com.tapacad.spring.Audi@6f03482
com.tapacad.spring.Audi@6f03482
```

Now if you observe from the above output both car1 and car2 are referring to the same object. This is only a **singleton where the container creates a single instance of that bean.** By default **scope is singleton.**

The Spring Framework supports the following five scopes, three of which are available only if you use a web-aware ApplicationContext.

- Singleton
- Prototype
- Request (web-aware ApplicationContext)
- Session (web-aware ApplicationContext)
- Global-session (web-aware ApplicationContext)

Singleton scope:

- By default spring will be singleton scope. If you want spring to create different objects instead of one then you need to make use of prototype scope.
- Singleton will create bean of all the classes that is mentioned in the applicationContext

Prototype scope:

- A bean with the *prototype* scope will return a different instance every time it is requested from the container.
- Now to set the scope to prototype scope you need to make use of scope attribute and specify the type of scope as shown below

```
<bean id ="audi" class="com.tapacad.spring.Audi"
scope="prototype">
<property name="rocketEngine" ref = "engine"></property>
<property name="colour" value = "${colour}"></property>
<property name="price" value = "${price}"></property>
</bean>
```

MyApp5.java

```
package com.tapacad.spring;
import
org.springframework.context.support.ClassPathXmlApplicationCo
ntext;
public class MyApp5 {

     public static void main(String[] args) {
//         Load Application context
           ClassPathXmlApplicationContext context =
                     new
ClassPathXmlApplicationContext("applicationContext.xml");

//         Get bean
           Car car1 = (Audi)context.getBean("audi");
           Car car2 = (Audi)context.getBean("audi");

//         print object reference of car1 and car2()
           System.out.println(car1);
           System.out.println(car2);

//         close context
           context.close();

     }
}
```

Output:

```
com.tapacad.spring.Audi@179ece50
com.tapacad.spring.Audi@3b0090a4
```

Now if you observe from the above output two different beans have been created.