MUMBAI EDUCATIONAL TRUST

MET INSTITUTE OF COMPUTER SCIENCE

Practical No. :	**11**
Roll No. :	**1520**
Title of Practical :	**Spring Application**
Objective :	**Write a Program to Demonstrate dependency injection and bean scopes in a Spring application.**

CODE :

Singer.java

```java
package edu.met.p1;

public class Singer
{
        int sid;
        String Sname;

        //Default Constructor
        public Singer()
        {

        }
        //Parameterized Constructor
        public Singer(int sid, String sname) {
                super();
                this.sid = sid;
                Sname = sname;
        }

        public int getSid() {
```

```java
            return sid;
    }

    public void setSid(int sid) {
            this.sid = sid;
    }

    public String getSname() {
            return Sname;
    }

    public void setSname(String sname) {
            Sname = sname;
    }

    @Override
    public String toString() {
            return "Singer [sid=" + sid + ", Sname=" + Sname + "]";
    }

}
```

appctx.xml

```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"
    xmlns:p="http://www.springframework.org/schema/p"
    xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
    <bean id="singer" class="edu.met.p1.Singer" scope="prototype">
        <constructor-arg value="101"></constructor-arg>
        <constructor-arg value="Om"></constructor-arg>
    </bean>
</beans>
```

SingerImpl.java

```java
package edu.met.p1;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class SingerImpl {
        static ApplicationContext ctx;
        public static void main(String[] args) {
                ctx = new ClassPathXmlApplicationContext("appctx.xml");
                Singer s1=(Singer)ctx.getBean("singer");
                System.out.println(s1);

                Singer s2=(Singer)ctx.getBean("singer");
                s2.setSid(102);
                s2.setSname("Neha");
                System.out.println(s2);


        }
}
```
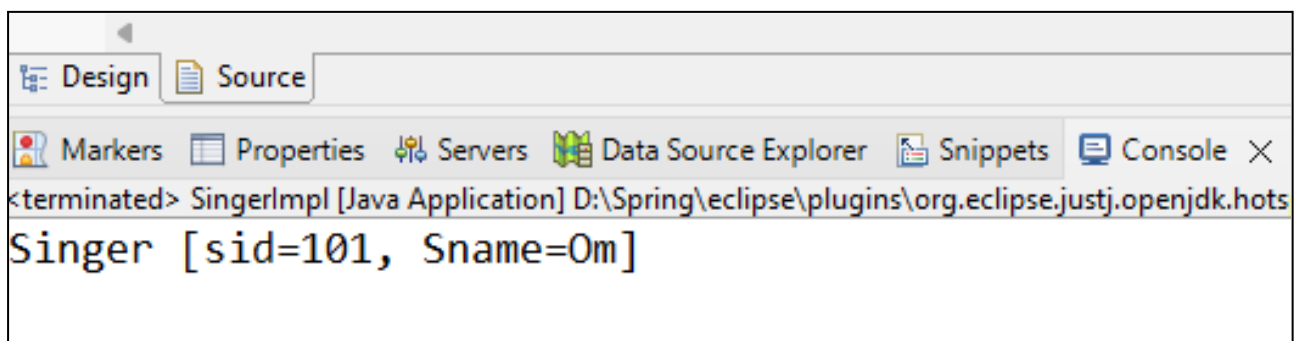
OUTPUT :
Scope: Singleton


```
Markers   Properties   Servers   Data Source Explorer   Snippets   Console  X
<terminated> SingerImpl [Java Application] D:\Spring\eclipse\plugins\org.eclipse.justj.openjdk.hots
Singer [sid=101, Sname=Om]
```

Scope: Prototype


```
Markers   Properties   Servers   Data Source Explorer   Snippets   Console  X
<terminated> SingerImpl [Java Application] D:\Spring\eclipse\plugins\org.eclipse.justj.openjdk.hots
Singer [sid=101, Sname=Om]
Singer [sid=102, Sname=Neha]
```