Program 2: constructor based dependency injection

create a bean class college and department:

```java
package com.example.Institute;

public class College {
        String Name, Address;
        public String getName() {
                return Name;
        }
        public void setName(String name) {
                Name = name;
        }
        public String getAddress() {
                return Address;
        }
        public void setAddress(String
address) {
                Address = address;
        }
}
```

```java
package com.example.Institute;

public class Department {
        String name, description;
        int id;
        College college;
        public Department(College college) {
                this.college = college;
        }
        public College getCollege() {
                return college;
        }
        public void setCollege(College college) {
                this.college = college;
        }
        public String getName() {
                return name;
        }
        public void setName(String name) {
                this.name = name;
        }
        public String getDescription() {
                return description;
        }
        public void setDescription(String
description) {
                this.description = description;
        }
        public int getId() {
                return id;
        }
        public void setId(int id) {
                this.id = id;
        }
}
```

<u>Mainclass for the spring boot to execute:</u>

```java
package com.example.Institute;
//Imports

@SpringBootApplication
public class DemoApplication {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
```

```java
SpringApplication.run(DemoApplication.class, args);
ApplicationContext ac = new ClassPathXmlApplicationContext("testBean.xml");
Department dt = (Department)ac.getBean("department");

while(true) {
        System.out.println("1. insert the Department details\n "
                        + "2. Display Department with College details \n"
                        + "3. exit");
        System.out.println("enter the choice");
        int a = sc.nextInt();
        switch(a) {
                case 1:
                        System.out.println("enter the Department name");
                                dt.setName(sc.next());
                                System.out.println("enter the Department no");
                                dt.setId(sc.nextInt());
                                System.out.println("enter the Department Decription");
                                dt.setDescription(sc.next());
                                System.out.println("thanks for input");
                                break;
                case 2:
                        System.out.println("Department Details:");
                        System.out.println("Name:"+dt.getName()+" "+"
ID:"+dt.getId()+
                                        " Department Description"+dt.getDescription());
                        System.out.println("College Details");
                        College c = dt.getCollege();
                        System.out.println("College Name: "+c.getName());
                        System.out.println("College Address: "+c.getAddress());
                        break;
                case 3:
                        System.exit(0);
                }
        }
    }
}
```

TestBean.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"
        xsi:schemaLocation="http://www.springframework.org/schema/beans
                https://www.springframework.org/schema/beans/spring-beans.xsd">

        <bean id="department" class="com.example.Institute.Department"
scope="prototype">
                <constructor-arg name="college" ref="college"/>
        </bean>
```

```xml
    <bean id="college" class="com.example.Institute.College" scope="singleton">
        <property name="name" value="RV College of Engineering"/>
        <property name="address" value="Mysuru Road, Bengaluru"/>
    </bean>
</beans>
```