

Assignment No 7: based on Spring Framework

Q1) Create a simple spring program

Code:

HelloWold.java

```
package Spring;
public class HelloWorld {
    public void display() {
        System.out.println("HelloWorld!");
    }
}
```

App.java

```
package Spring;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
public class App1 {
    public static void main(String[] args) {
        // TODO Auto-generated method stub

        ApplicationContext con =new
ClassPathXmlApplicationContext("config.xml");
        HelloWorld h=(HelloWorld) con.getBean("HW");
        h.display();
    }
}
```

Config.java

```
<?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="HW" class="Spring.HelloWorld"></bean>
<bean id="acc" class="Srting_1.Account"></bean>
</beans>
```

Output:

```
log4j:WARN No appenders could be found for logger (org.springframework)
log4j:WARN Please initialize the log4j system properly.
HelloWorld!
```

Q2) Write a program to demonstrate dependency injection via setter method. (Bank Account Application)

Account.java

```
package Spring_1;
public class Account {
    private int accNo;
    private String accName;
    public int getAccNo() {
        return accNo;
    }
    public void setAccNo(int accNo) {
        this.accNo = accNo;
    }
    public String getAccName() {
        return accName;
    }
    public void setAccName(String accName) {
        this.accName = accName;
    }
}
```

App1.java

```
package Spring_1;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class App {
    public static void main(String[] args) {
        ApplicationContext context = new
        ClassPathXmlApplicationContext("config.xml");
        Account acc = (Account) context.getBean("acc");
        System.out.println("Account no: " + acc.getAccNo());
        System.out.print("Account Name: " + acc.getAccName());
    }
}
```

Config.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="HW" class="Spring.HelloWorld"></bean>
<bean id="acc" class="Spring_1.Account">
<property name="accNo" value="69"></property>
<property name="accName" value="Aditya Pandey"></property></bean>
</beans>
```

Output:

```
<terminated> App (3) [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe [Jan 6, 2023, 4:12:38 PM]
log4j:WARN No appenders could be found for logger (org.spr
log4j:WARN Please initialize the log4j system properly.
Account no: 69
Account Name: Aditya pandey
```

Q3) Write a program to demonstrate dependency injection via Constructor (Employee Application).

App2.java

```
package Spring_2;
import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import org.springframework.core.io.ClassPathResource;
public class App2 {
    public static void main(String[] args) {
        ApplicationContext context = new
ClassPathXmlApplicationContext("config.xml");
        ClassPathResource r=new ClassPathResource("config.xml");
        BeanFactory factory=new XmlBeanFactory(r);
        Employee s=(Employee)factory.getBean("e");
        s.show();
    }
}
```

```
}
```

Employee.java

```
package Spring_2;
public class Employee {
    private int id;
    private String name;
    private Address address;//Aggregation
    public Employee() {System.out.println("def cons");}
    public Employee(int id, String name, Address address) {
        super();
        this.id = id;
        this.name = name;
        this.address = address;
    }
    void show()
    {
        System.out.println(id+" "+name);
        System.out.println(address.toString());
    }
}
```

Address.java

```
package Spring_2;
public class Address {
    private String city;
    private String state;
    private String country;

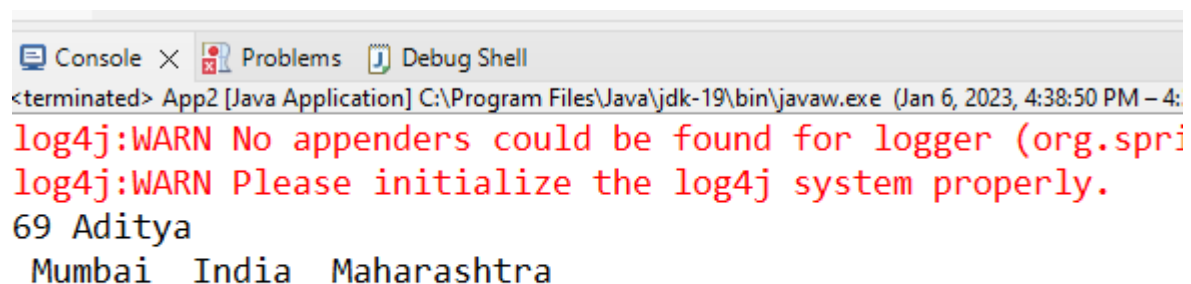
    public Address(String city,String state, String country) {
        super();
        this.city = city;
        this.state=state;
        this.country=country;
    }
    public String toString() {
        return city + " " + state + " " + country;
    }
}
```

Config.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="a1" class="Spring_2.Address">
<constructor-arg value=" Mumbai " index="0"> </constructor-arg>
<constructor-arg value=" Maharashtra " index="2"> </constructor-arg>
<constructor-arg value=" India " index="1"> </constructor-arg>
</bean>
<bean id="e" class="Spring_2.Employee">
<constructor-arg value="69" type="int"></constructor-arg>
<constructor-arg value="Aditya"></constructor-arg>
<constructor-arg> <ref bean="a1"/> </constructor-arg> </bean>
</beans>
```

Output:



The screenshot shows a Java IDE console window with three tabs: Console, Problems, and Debug Shell. The Console tab is active, displaying the following output:

```
<terminated> App2 [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (Jan 6, 2023, 4:38:50 PM - 4:
log4j:WARN No appenders could be found for logger (org.spr
log4j:WARN Please initialize the log4j system properly.
69 Aditya
Mumbai India Maharashtra
```