PRACTICAL 8(B) – SPRING AOP (AFTER ADVICE)

AIM

To demonstrate Spring AOP – After Advice.

THEORY

After Advice in Spring AOP executes after the target method completes execution, regardless of its outcome. It is commonly used for logging and cleanup operations.

SOURCE CODE – Target Class (BusinessLogic.java)

```
package aopdemo;

public class BusinessLogic {
    public void process() {
        System.out.println("Business logic executed");
    }
}
```

SOURCE CODE – Aspect Class (AfterAspect.java)

```
package aopdemo;

import org.aspectj.lang.annotation.After;
import org.aspectj.lang.annotation.Aspect;

@Aspect
public class AfterAspect {

    @After("execution(* aopdemo.BusinessLogic.process(..))")
    public void afterAdvice() {
        System.out.println("After advice executed");
    }
}
```

SOURCE CODE – applicationContext.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:aop="http://www.springframework.org/schema/aop"
       xsi:schemaLocation="
       http://www.springframework.org/schema/beans
       http://www.springframework.org/schema/beans/spring-beans.xsd
       http://www.springframework.org/schema/aop
       http://www.springframework.org/schema/aop/spring-aop.xsd">

    <aop:aspectj-autoproxy/>

    <bean id="business" class="aopdemo.BusinessLogic"/>
    <bean class="aopdemo.AfterAspect"/>
</beans>
```

SOURCE CODE – MainApp.java

```
package aopdemo;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class MainApp {
    public static void main(String[] args) {
        ApplicationContext context =
            new ClassPathXmlApplicationContext("applicationContext.xml");

        BusinessLogic obj = (BusinessLogic) context.getBean("business");
        obj.process();
    }
}
```

NETBEANS 7 – EXECUTION STEPS

1. Use the same Spring AOP project and libraries from Practical 8(A)
2. Create AfterAspect.java in aopdemo package
3. Update applicationContext.xml
4. Run MainApp.java using Shift + F6

OUTPUT

Business logic executed After advice executed

RESULT

Thus, Spring AOP After Advice was successfully implemented.