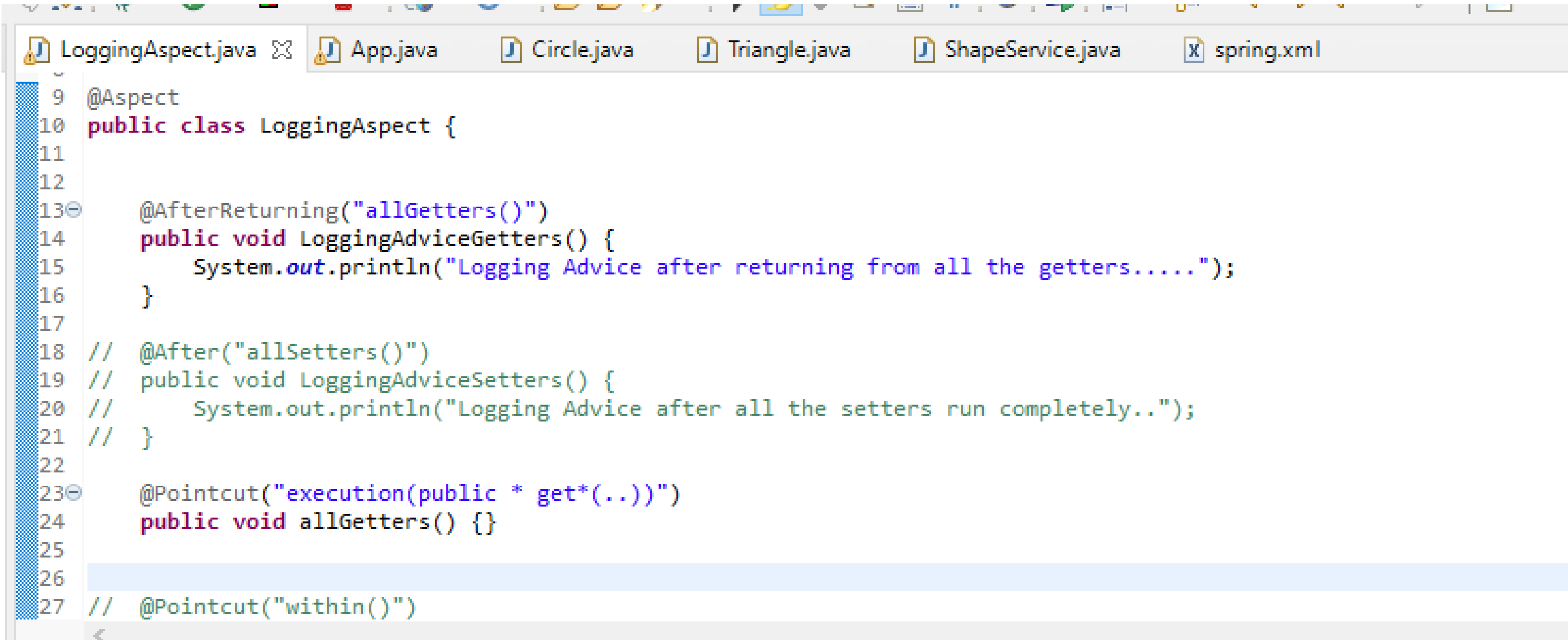
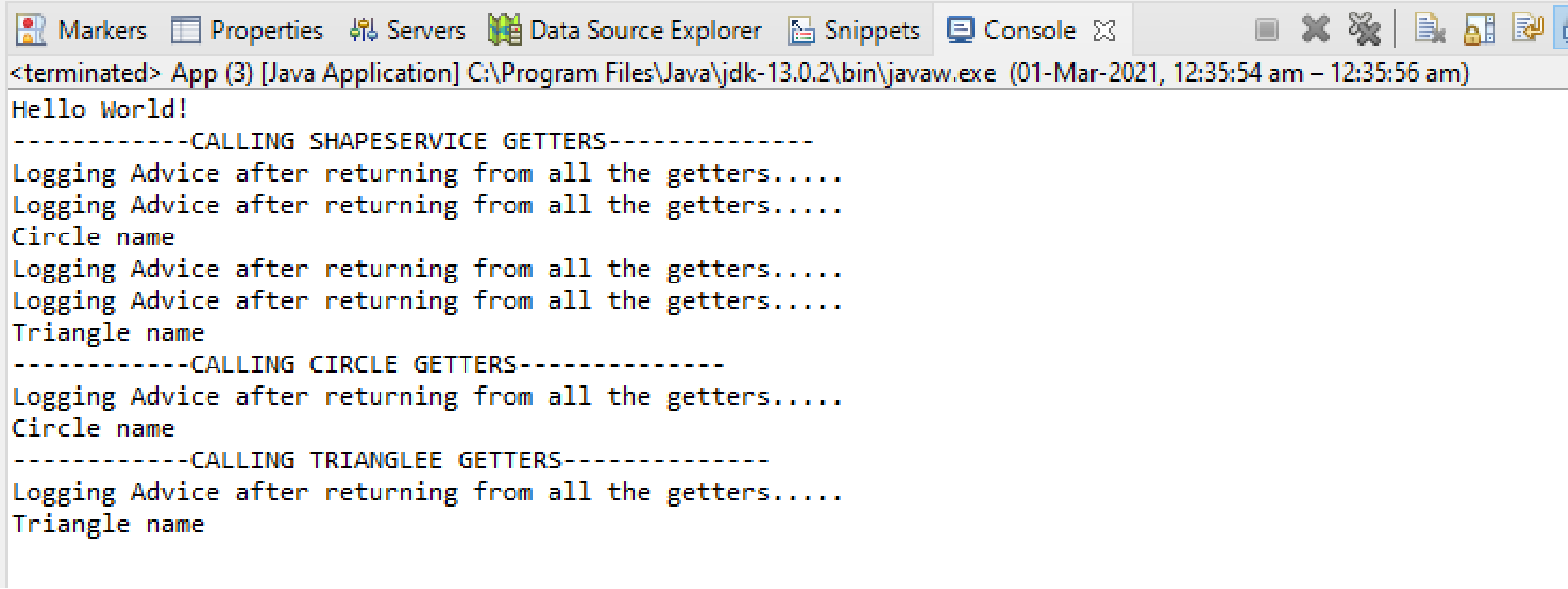
2) Create aspect using After, AfterReturning, AfterThrowing, Around advices in AOP.
Use pointcut types within, args also along with execution.

USING **@AfterReturning and pointcut** type **execution** to run all getters.



```
@Aspect
public class LoggingAspect {


    @AfterReturning("allGetters()")
    public void LoggingAdviceGetters() {
        System.out.println("Logging Advice after returning from all the getters.....");
    }

//  @After("allSetters()")
//  public void LoggingAdviceSetters() {
//      System.out.println("Logging Advice after all the setters run completely..");
//  }

    @Pointcut("execution(public * get*(..))")
    public void allGetters() {}


//  @Pointcut("within()")
```

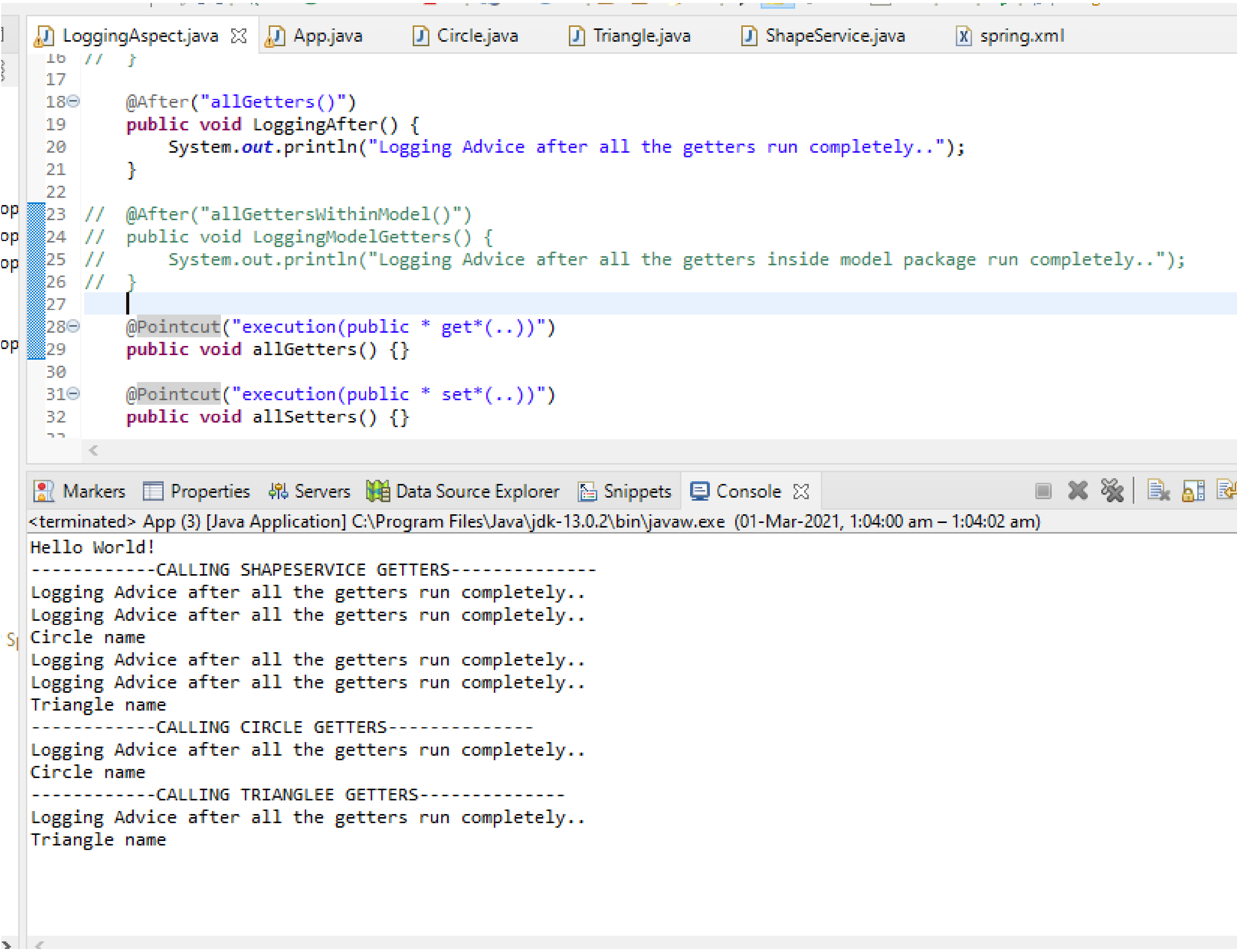
OUTPUT



<terminated> App (3) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (01-Mar-2021, 12:35:54 am – 12:35:56 am)

```
Hello World!
------------CALLING SHAPESERVICE GETTERS--------------
Logging Advice after returning from all the getters.....
Logging Advice after returning from all the getters.....
Circle name
Logging Advice after returning from all the getters.....
Logging Advice after returning from all the getters.....
Triangle name
------------CALLING CIRCLE GETTERS--------------
Logging Advice after returning from all the getters.....
Circle name
------------CALLING TRIANGLEE GETTERS--------------
Logging Advice after returning from all the getters.....
Triangle name
```

USING **@After**

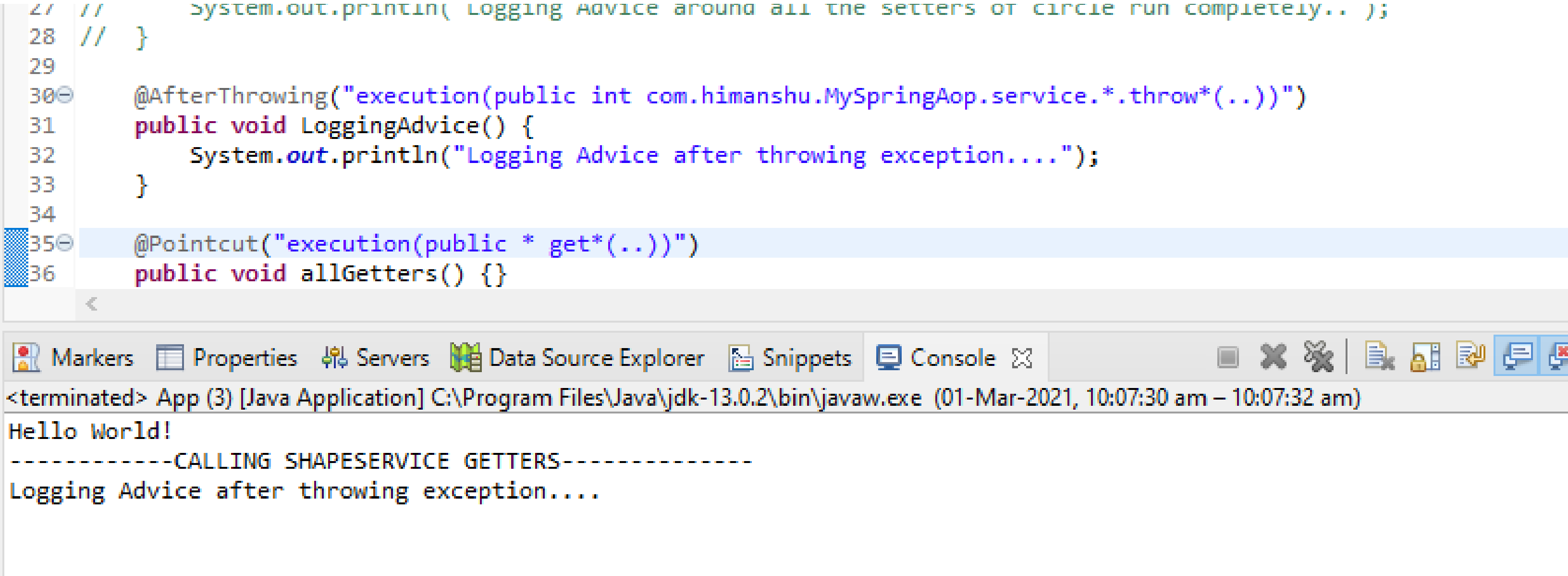


```java
//    }

    @After("allGetters()")
    public void LoggingAfter() {
        System.out.println("Logging Advice after all the getters run completely..");
    }

//  @After("allGettersWithinModel()")
//  public void LoggingModelGetters() {
//      System.out.println("Logging Advice after all the getters inside model package run completely..");
//  }

    @Pointcut("execution(public * get*(..))")
    public void allGetters() {}

    @Pointcut("execution(public * set*(..))")
    public void allSetters() {}
```

<terminated> App (3) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (01-Mar-2021, 1:04:00 am – 1:04:02 am)

```
Hello World!
------------CALLING SHAPESERVICE GETTERS--------------
Logging Advice after all the getters run completely..
Logging Advice after all the getters run completely..
Circle name
Logging Advice after all the getters run completely..
Logging Advice after all the getters run completely..
Triangle name
------------CALLING CIRCLE GETTERS--------------
Logging Advice after all the getters run completely..
Circle name
------------CALLING TRIANGLEE GETTERS--------------
Logging Advice after all the getters run completely..
Triangle name
```

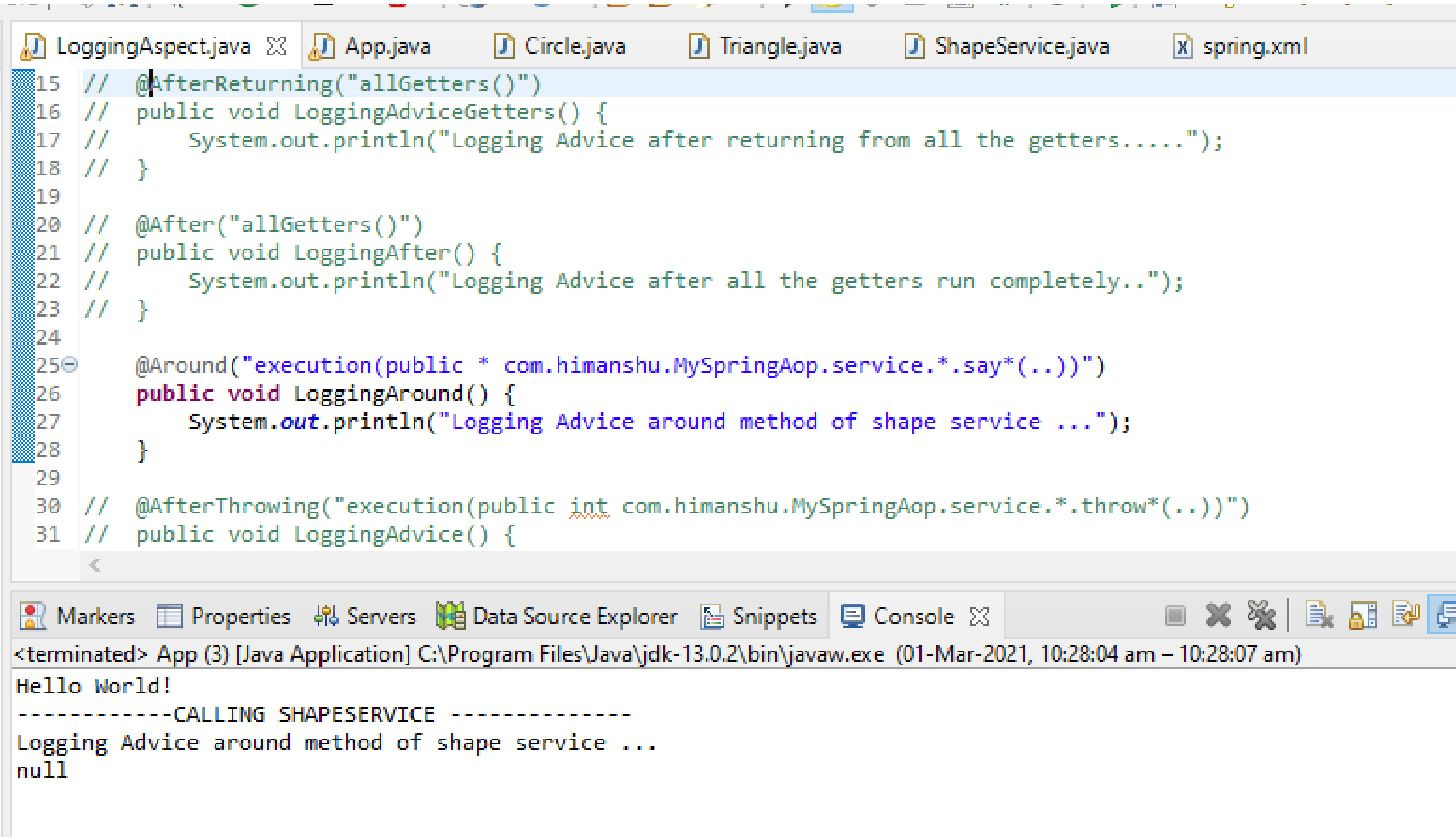
USING **@AFTERTHROWING**



```
//        System.out.println("Logging Advice around all the setters of circle run completely..");
//    }

    @AfterThrowing("execution(public int com.himanshu.MySpringAop.service.*.throw*(..))")
    public void LoggingAdvice() {
        System.out.println("Logging Advice after throwing exception....");
    }

    @Pointcut("execution(public * get*(..))")
    public void allGetters() {}
```

```
<terminated> App (3) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (01-Mar-2021, 10:07:30 am – 10:07:32 am)
Hello World!
------------CALLING SHAPESERVICE GETTERS--------------
Logging Advice after throwing exception....
```

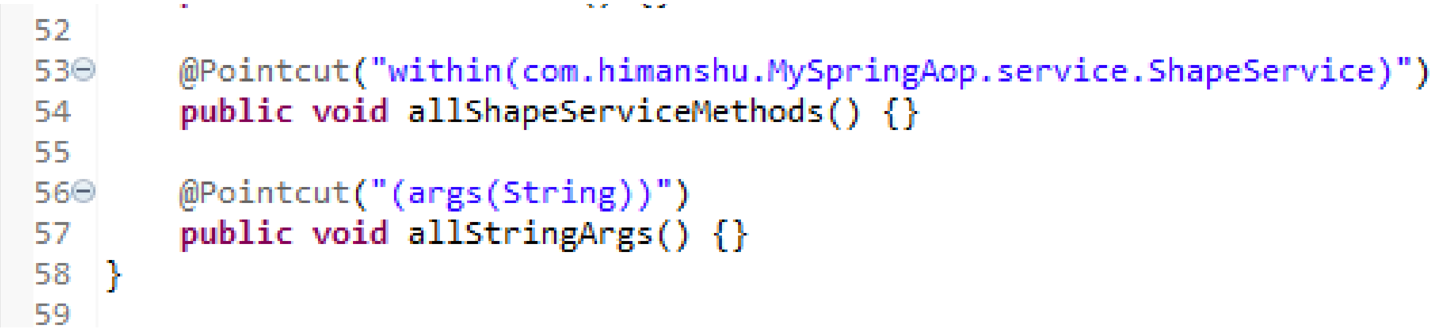
USING **@AROUND**



```
//  @AfterReturning("allGetters()")
//  public void LoggingAdviceGetters() {
//      System.out.println("Logging Advice after returning from all the getters.....");
//  }

//  @After("allGetters()")
//  public void LoggingAfter() {
//      System.out.println("Logging Advice after all the getters run completely..");
//  }

    @Around("execution(public * com.himanshu.MySpringAop.service.*.say*(..))")
    public void LoggingAround() {
        System.out.println("Logging Advice around method of shape service ...");
    }

//  @AfterThrowing("execution(public int com.himanshu.MySpringAop.service.*.throw*(..))")
//  public void LoggingAdvice() {
```

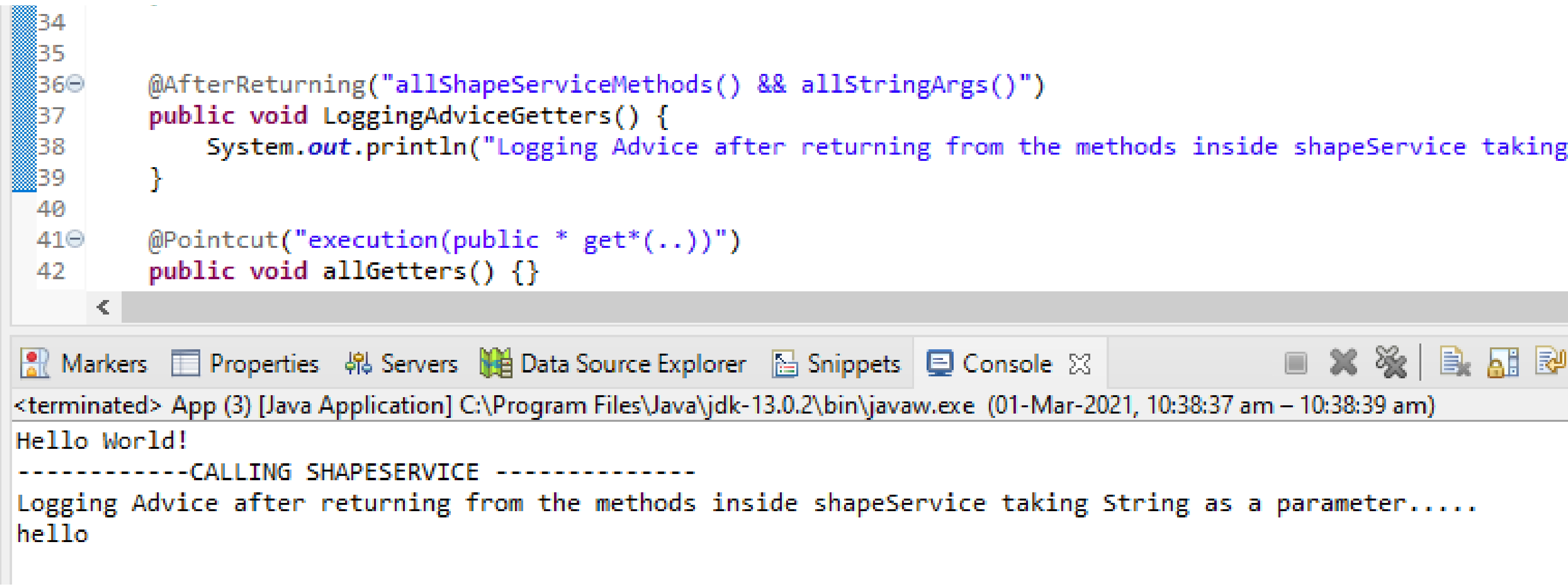
```
<terminated> App (3) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (01-Mar-2021, 10:28:04 am – 10:28:07 am)
Hello World!
------------CALLING SHAPESERVICE --------------
Logging Advice around method of shape service ...
null
```

USING **WITHIN** AND **ARGS**



```
    @Pointcut("within(com.himanshu.MySpringAop.service.ShapeService)")
    public void allShapeServiceMethods() {}

    @Pointcut("(args(String))")
    public void allStringArgs() {}
}
```

OUTPUT



```
    @AfterReturning("allShapeServiceMethods() && allStringArgs()")
    public void LoggingAdviceGetters() {
        System.out.println("Logging Advice after returning from the methods inside shapeService taking
    }

    @Pointcut("execution(public * get*(..))")
    public void allGetters() {}
```

```
<terminated> App (3) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (01-Mar-2021, 10:38:37 am – 10:38:39 am)
Hello World!
------------CALLING SHAPESERVICE --------------
Logging Advice after returning from the methods inside shapeService taking String as a parameter.....
hello
```