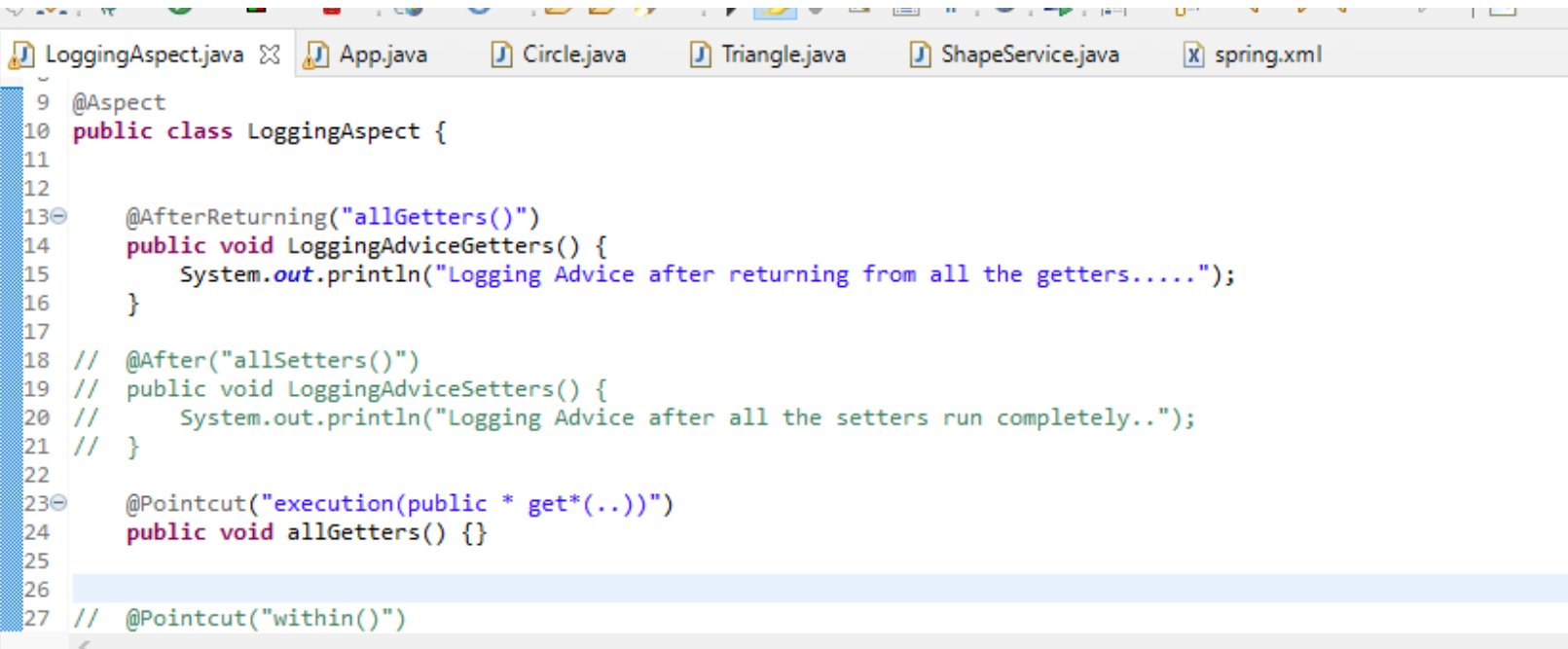


2) Create aspect using After, AfterReturning, AfterThrowing, Around advices in AOP.

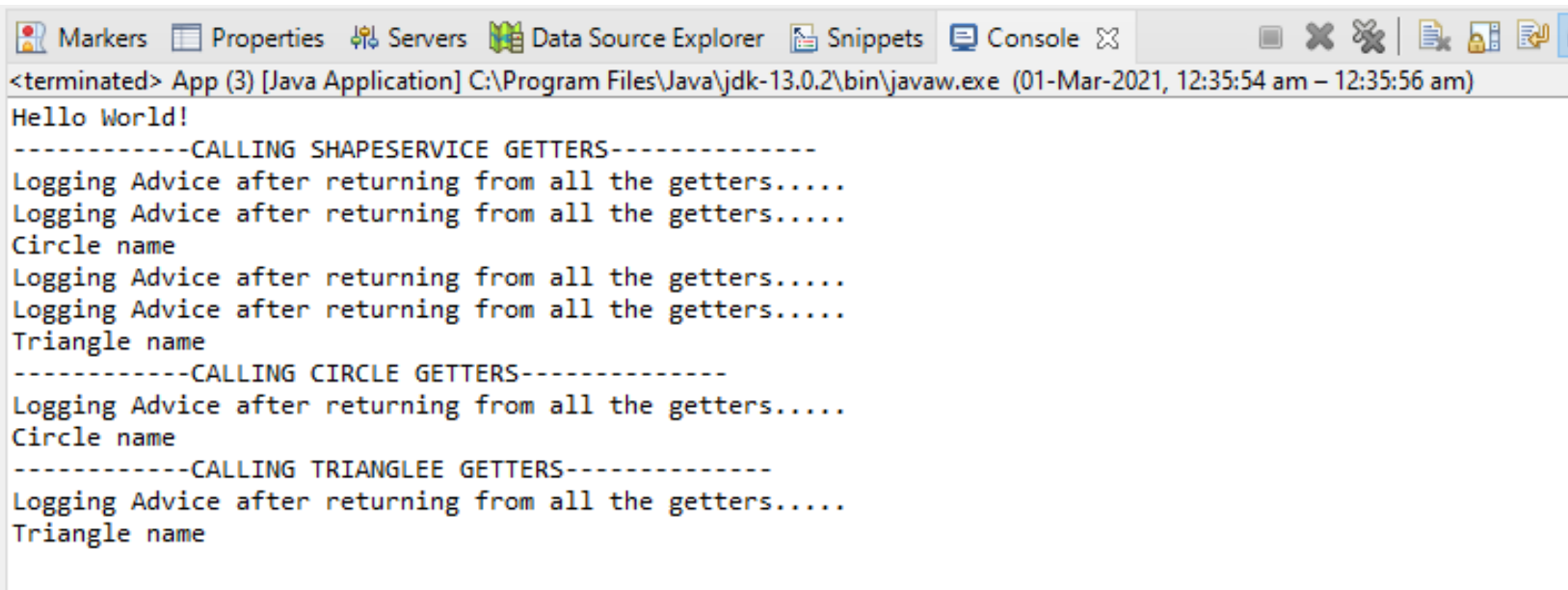
Use pointcut types within, args also along with execution.

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



```
9 @Aspect
10 public class LoggingAspect {
11
12
13     @AfterReturning("allGetters()")
14     public void LoggingAdviceGetters() {
15         System.out.println("Logging Advice after returning from all the getters.....");
16     }
17
18     // @After("allSetters()")
19     // public void LoggingAdviceSetters() {
20     //     System.out.println("Logging Advice after all the setters run completely..");
21     // }
22
23     @Pointcut("execution(public * get*(..))")
24     public void allGetters() {}
25
26
27     // @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 TRIANGLEEE GETTERS-----
Logging Advice after returning from all the getters.....
Triangle name
```

USING @After

The screenshot shows an IDE with several tabs: `LoggingAspect.java`, `App.java`, `Circle.java`, `Triangle.java`, `ShapeService.java`, and `spring.xml`. The `LoggingAspect.java` file is active, displaying the following code:

```
16 // }
17
18 @After("allGetters()")
19 public void LoggingAfter() {
20     System.out.println("Logging Advice after all the getters run completely..");
21 }
22
23 // @After("allGettersWithinModel()")
24 // public void LoggingModelGetters() {
25 //     System.out.println("Logging Advice after all the getters inside model package run completely..");
26 // }
27
28 @Pointcut("execution(public * get*(..))")
29 public void allGetters() {}
30
31 @Pointcut("execution(public * set*(..))")
32 public void allSetters() {}
33
```

The IDE's interface includes tabs for `Markers`, `Properties`, `Servers`, `Data Source Explorer`, `Snippets`, and `Console`. The `Console` tab is selected, showing the following output:

```
<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 TRIANGLE GETTERS-----
Logging Advice after all the getters run completely..
Triangle name
```

USING @AFTERTHROWING

```
27 //      System.out.println( Logging Advice around all the setters or circle run completely.. );
28 //  }
29
30 @AfterThrowing("execution(public int com.himanshu.MySpringAop.service.*.throw*(..))")
31 public void LoggingAdvice() {
32     System.out.println("Logging Advice after throwing exception....");
33 }
34
35 @Pointcut("execution(public * get*(..))")
36 public void allGetters() {}
```

Markers Properties Servers Data Source Explorer Snippets Console

<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

LoggingAspect.java App.java Circle.java Triangle.java ShapeService.java spring.xml

```
15 // @AfterReturning("allGetters()")
16 // public void LoggingAdviceGetters() {
17 //     System.out.println("Logging Advice after returning from all the getters.....");
18 // }
19
20 // @After("allGetters()")
21 // public void LoggingAfter() {
22 //     System.out.println("Logging Advice after all the getters run completely..");
23 // }
24
25 @Around("execution(public * com.himanshu.MySpringAop.service.*.say*(..))")
26 public void LoggingAround() {
27     System.out.println("Logging Advice around method of shape service ...");
28 }
29
30 // @AfterThrowing("execution(public int com.himanshu.MySpringAop.service.*.throw*(..))")
31 // public void LoggingAdvice() {
```

Markers Properties Servers Data Source Explorer Snippets Console

<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

```
52  
53 @Pointcut("within(com.himanshu.MySpringAop.service.ShapeService)")  
54 public void allShapeServiceMethods() {}  
55  
56 @Pointcut("(args(String))")  
57 public void allStringArgs() {}  
58 }  
59
```

OUTPUT

```
34  
35  
36 @AfterReturning("allShapeServiceMethods() && allStringArgs()")  
37 public void LoggingAdviceGetters() {  
38     System.out.println("Logging Advice after returning from the methods inside shapeService taking  
39 }  
40  
41 @Pointcut("execution(public * get*(..))")  
42 public void allGetters() {}
```

Markers Properties Servers Data Source Explorer Snippets Console

<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