```
==================================pom.xml==========================================
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>2.3.0.RELEASE</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>
    <groupId>com.ameya</groupId>
    <artifactId>002-AOP</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>002-AOP</name>
    <description>Demo project for Spring Boot</description>
    <properties>
        <java.version>1.8</java.version>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-aop</artifactId>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-web</artifactId>
        </dependency>

        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-test</artifactId>
            <scope>test</scope>
            <exclusions>
                <exclusion>
                    <groupId>org.junit.vintage</groupId>
                    <artifactId>junit-vintage-engine</artifactId>
                </exclusion>
            </exclusions>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <plugin>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-maven-plugin</artifactId>
            </plugin>
        </plugins>
    </build>

</project>
=================================================application.properties================================
server.port=9001
logging.level.web=trace
===============================================Employee.java============================================
package com.ameya.models;

public class Employee {
    private int age;
    private String firstName;
    private String lastName;
    private double salary;
    public Employee() {}
    public Employee(int age, String firstName, String lastName, double salary) {
        super();
        this.age = age;
        this.firstName = firstName;
        this.lastName = lastName;
```

```java
        this.salary = salary;
    }
    public int getAge() {
        return age;
    }
    public void setAge(int age) {
        this.age = age;
    }
    public String getFirstName() {
        return firstName;
    }
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }
    public String getLastName() {
        return lastName;
    }
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
    public double getSalary() {
        return salary;
    }
    public void setSalary(double salary) {
        this.salary = salary;
    }
    @Override
    public String toString() {
        return "Employee [age=" + age + ", firstName=" + firstName + ", lastName=" + lastName + ", salary=" + salary
                + "]";
    }


}
========================================EmployeeService.java===========================
package com.ameya.services;

import org.springframework.stereotype.Service;

import com.ameya.models.Employee;

@Service
public class EmployeeService {

    public Employee createEmployee(int age,String firstName,
            String lastName,double salary) {
        Employee emp=new Employee(age, firstName, lastName, salary);
        System.out.println("++++ EmployeeService.createEmployee ++++"+emp);

        return emp;
    }
    public void updateEmployee(Employee employee) {
        System.out.println("++++ EmployeeService.updateEmployee "
                + "Employee Updated ++++");
    }
    public void deleteEmployee() {
        System.out.println("++++ EmployeeService.deleteEmployee "
                + "Employee Deleted ++++");
    }
}
===================================EmployeeController.java===================================
package com.ameya.controllers;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.RestController;

import com.ameya.models.Employee;
import com.ameya.services.EmployeeService;
```

```java

@RestController
public class EmployeeController {

    @Autowired
    private EmployeeService employeeService;
    @GetMapping(path="/add/employee")
    public Employee addEmployee(@RequestParam("age") int age,
                                @RequestParam("fname") String firstName,
                                @RequestParam("lname") String lastName,
                                @RequestParam("sal") double salary) {
        return employeeService.createEmployee(age, firstName, lastName, salary);
    }
    @GetMapping(path="/delete/employee")
    public String delete() {
        employeeService.deleteEmployee();
        return "Employee Deleted !!";
    }
    @GetMapping(path="/update/employee")
    public String update(Employee employee) {
        employeeService.updateEmployee(employee);
        return "Employee Updated !!";
    }
}
========================================LoggingAspectJ.java================================
package com.ameya.aspects;

import org.aspectj.lang.JoinPoint;
import org.aspectj.lang.ProceedingJoinPoint;
import org.aspectj.lang.annotation.After;
import org.aspectj.lang.annotation.Around;
import org.aspectj.lang.annotation.Aspect;
import org.aspectj.lang.annotation.Before;
import org.aspectj.lang.annotation.Pointcut;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.stereotype.Component;

@Component
@Aspect
public class LoggingAspectJ {

    private static final Logger logger=LoggerFactory.
            getLogger(LoggingAspectJ.class);
    @Pointcut("execution(* com.ameya.services.EmployeeService.cre*(..))")
    public void myPointcut() {}

    @Before(value="myPointcut()")
    public void beforeAdvice(JoinPoint joinPoint) {
        logger.info("******* Before Advice ---> Before Method =>"+
    joinPoint.getSignature());
    }
    @After(value="myPointcut()")
    public void afterAdvice(JoinPoint joinPoint) {
        logger.info("******* After Advice ---> After Method =>"+
                joinPoint.getSignature());
    }
    @Around(value="myPointcut()")
    public Object aroundAdvice(ProceedingJoinPoint joinPoint)
            throws Throwable {
        Object object=null;
        logger.info("******* Around Advice ---> Before Method =>"+
        joinPoint.getSignature());
        try {
            object=joinPoint.proceed();
            logger.info("******* Around Advice ---> After Method =>"+
                    joinPoint.getSignature());
        }catch(Throwable t) {
            throw t;
        }
        return object;
    }
}
====================================Application.java===========================
package com.ameya;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.annotation.EnableAspectJAutoProxy;

@SpringBootApplication
@EnableAspectJAutoProxy(proxyTargetClass = true)
public class Application {

    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }

}
=================================================================

http://localhost:9001/add/employee?age=45&fname=Ameya&lname=Joshi&sal=45000
http://localhost:9001/delete/employee
```