Spring AOP

Teaching Faculty: Umur INAN

Prepared by Muhyieddin **AL-TARAWNEH**, Umur **INAN**

Single Responsibility Principle

- Every class should have responsibility over a single piece of functionality provided.

- That responsibility should be entirely encapsulated by the class. All its services should be narrowly aligned with that responsibility.

The General AOP Use Cases

- A functional implementation is scattered if its code is spread out over multiple modules. Its implementation is not modular.

- A small change in functionality needs to be made in multiple modules.
  - Logging is “scattered” throughout an application

The General AOP Use Cases

- A functional implementation is tangled if its code is intermixed with code that implements other functionality.
- The module in which tangling occurs is not cohesive.
  - [Programmatic] Transaction Management is “tangled” within a method

Aspect-oriented approach identifies code scattering and tangling as the indicators of crosscutting concerns.

AOP Value Added

- Separation of Concerns
- Increased Modularity
- Reduces “spaghetti” code
- Code reduction
- Removes “hard” dependencies

Some use cases

- Boilerplate/repetitive code
- Transaction
- Security
- Logging
- Authorization
- Validation

AOP

- One of the key components of Spring is the AOP framework.

- The Spring Framework uses Spring AOP internally for transaction management, security, remote access, and Cache Abstraction.

AOP Definitions

- Cross-cutting Concern
  - Another name for an Aspect.
  - An Aspect “crosscuts” core functionality.

- Aspect
  - Functionality fundamental to application.
  - BUT not the primary business function.

AOP Definitions

- Aspect
  - implemented by applying Advice (additional behavior) at various Join points (methods in Spring application) specified by a Pointcut (criteria to match Join points to Advice)

AOP Definitions

- Advice
  - Implementation code of the aspect.
  - [executed Around, Before or After Join point]
  - [Associated with Join Point through a Pointcut]
- Join point
  - Where Advice code in applied in application
  - [Always class methods in Spring AOP]
- Pointcut
  - An expression that defines a set of Join points

Dynamic AOP in Spring

- Cross-cutting logic applied at run time
- Proxy based approach [simple to use]
- Aspects applied to methods only
- Spring Managed beans

ADVICE TYPES

- Before
  - executes before a join point
- AfterReturning
  - Executes if a join point completes normally
- AfterThrowing
  - executes if a join point throws an exception
- After
  - executes if a join point executes normally OR throws an exception
- Around
  - Before AND after the joint point. Also, can end execution or throw exception

Downside to AOP

- Functionality gets fragmented across source files and hard to understand.
- You don't have a clear overview of what code runs when.
- Problematic to debug the AOP based application code.
- Code Inspection and reviews are challenging.
- AOP can be too complex for application in the Enterprise.
- There are only a handful of cross-cutting concerns.

Prepared by Muhyieddin **AL-TARAWNEH**, Umur **INAN**

Point Cut Designators

POINTCUT	DESCRIPTION	SYNTAX
Execution	Matches methods. Including visibility, return & parameters.	("execution(public * *.*.*(..))")
within	Matches join points within “range" of types[Class].	(“within(*.*.*.*)")
target	Matches where the target object is an instance of the given specific type [Class].	(“target(pkg.pkg.pkg.class)")
args	Matches where the arguments are instances of the given types.	(“args(..)")
annotation	Matches methods where the given annotation exists.	@annotation(anotationName)

Prepared by Muhyieddin **AL-TARAWNEH**, Umur **INAN**

Main points

- AOP applied judiciously can make an application more efficient and effective.
- Regular practice of the TM technique makes an individual more efficient and effective.
- Aspect Oriented Programming allows us to consolidate in one place the repetitive code that is scattered throughout an application, increasing the maintainability and clarity of the logic.
- Refining the function of the nervous system leads to improvement in physical health and mental clarity.