

스프링 배치 & 스케줄링 (Spring Batch & Scheduling)

1. 스프링 배치 & 스케줄링

- 스프링 배치 스케줄러는 주기적으로 배치 작업을 실행하기 위한 메커니즘을 제공한다.
- 스프링은 스프링 배치(Spring Batch)와 스프링 스케줄링(Spring Scheduling)을 통합하여 강력하고 유연한 배치 처리 및 스케줄링 기능을 지원한다.
- 이를 통해 개발자는 크론 표현식이나 간격을 기준으로 정기적으로 또는 지연된 배치 작업을 계획할 수 있으며 복잡한 배치 처리 작업도 쉽게 스케줄링하고 관리할 수 있다.
- 이를 통해 정기적인 데이터 처리, 보고서 생성, 데이터베이스 유지 관리 작업을 수월하게 진행할 수 있다.

스프링 배치(Spring Batch)

스프링 배치는 대량의 데이터를 처리하고, 반복적인 작업을 관리하기 위한 프레임워크이다. 파일 처리, 대량의 데이터베이스 작업, 복잡한 비즈니스 규칙 처리 등에 이상적이다. 스프링 배치는 잡(Job), 스텝(Step), 리더(Reader), 프로세서(Processor), 라이터(Writer) 등의 개념을 제공하여, 배치 작업의 구성요소를 세밀하게 제어할 수 있다.

스프링 스케줄링(Spring Scheduling)

스프링 스케줄링은 스프링 프레임워크가 제공하는 스케줄링 지원 기능으로, 주기적으로 작업을 실행할 수 있게 해준다. @Scheduled 어노테이션을 사용하여 메서드를 스케줄링할 수 있으며, 크론 표현식, 고정된 지연 시간, 고정된 주기 등을 기준으로 실행 시간을 정의할 수 있다.

2. 스케줄링 구현

1) 설정

- Application Class에 @EnableScheduling 어노테이션을 사용하여 스케줄링을 활성화하고, 배치 작업을 스케줄링한다.

@EnableScheduling (어노테이션추가)

2) 기능 구현

- Spring Boot에서 스케줄링 기능을 구현하기 위해서는 `@Scheduled` 어노테이션을 사용하여 메서드를 스케줄링 대상으로 지정할 수 있다.
- `@Scheduled` 어노테이션은 주기적으로 실행되어야 하는 메서드에 부여된다.
- Service로직에서 `@Scheduled(cron="초 분 시 일 월 요일 (연도)")` 선언 후 배치를 사용한다.
- 스케줄링에서는 `return`과 `parameter`를 사용할 수 없다.

3. cron 스케줄링 속성

fixedRate: 이 속성을 사용하면 이전 실행이 완료된 시점부터 고정된 간격으로 메서드를 반복해서 실행한다.

fixedDelay: 이 속성을 사용하면 이전 실행이 완료된 시점부터 고정된 시간이 경과한 후에 메서드를 실행한다.

initialDelay: 이 속성을 사용하면 스케줄링 시작 시 최초 실행까지의 지연 시간을 설정할 수 있다.

cron: 이 속성을 사용하면 cron 표현식을 사용하여 메서드를 실행할 시간을 지정할 수 있다.

`@Scheduled(cron="1 2 3 4 5 6 (7)")`

[필드]

1) 초 2) 분 3) 시 4) 일 5) 월 6) 요일 7) 연도(optional)

[표기법]

- * : 모든 경우의 수
- ? : 사용하지 않음
- : 기간
- , : 복수 선택
- / : 시작시간과 반복 간격 설정
- L : 마지막 기간
- W : 가장 가까운 평일 동작

[샘플 예시]

@Scheduled(cron="0 0 15 * * *")
@Scheduled(cron="0 30 15 * * *")
@Scheduled(cron="0 0/15 * * * *")
@Scheduled(cron="0 0/10 12,0 * * *")
@Scheduled(cron="0 0 9-12 * * *")
@Scheduled(cron="0 0 0 ? * MON")
@Scheduled(cron="0 0 18 ? * MON-FRI")

> 매일 15시에 실행
> 매일 15시 30분에 실행
> 매일 15분 간격으로 실행
> 매일 12시,0시에 10분 간격으로 실행
> 매일 9시 ~ 12시 정각에 실행
> 매주 월요일 0시에 실행
> 매주 월~금(평일) 18시에 실행