

Chapter 01. 배치와 스프링 배치 이해

Spring Batch

- 큰 단위의 작업을 **일괄 처리**
- 대부분 처리량이 많고 비 **실시간성 처리**에 사용
 - 대용량 데이터 계산, 정산, 통계, 데이터베이스, 변환 등
- 컴퓨터 **자원을 최대한로 활용**
 - 컴퓨터 자원 사용이 낮은 시간대에 배치를 처리하거나
 - 배치만 처리하기 위해 사용자가 사용하지 않는 또 다른 컴퓨터 자원을 사용할 수 있다.
- 사용자 상호작용으로 실행되기 보단, **스케줄러와 같은 시스템에 의해 실행**되는 대상
 - 예를 들면 매일 오전 10시에 배치 실행, 매주 월요일 12시 마다 실행
 - crontab, jenkins ...

- 배치 처리를 하기 위한 **Spring Framework** 기반 기술
 - **Spring**에서 지원하는 기술 적용 가능
 - **DI, AOP**, 서비스 추상화
- 간단한 작업(**Tasklet**) 기반 처리와, 대량 처리(**Chunk**) 기반 기능 지원

- 개발 환경
 - IntelliJ IDEA
 - Mac OS
- 프로젝트 생성
 - Java8 +
 - Gradle
 - Spring Boot 2.x +
 - Spring Batch, Spring JDBC, Spring Data JPA, Lombok, etc...
 - H2 DB
 - MySQL DB (<https://dev.mysql.com/downloads/mysql/>)

예제 - Hello Spring Batch