

# Spring Batch 기본 프로젝트

## 1 프로젝트 세팅

## 개발 환경과 기술 스택

3.

Spring Batch  
기본 프로젝트

### 개발 환경

- Java 8, Docker 20.10.5, MySQL 5.7

### 기술 스택

- Spring Boot 2.5, Spring Boot Batch 2.5, Spring Boot Data JPA 2.5, Junit 5.7.2, lombok 1.18.20, gradle 6.8

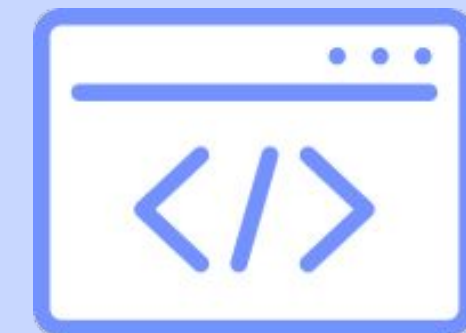
## 실습

3.

Spring Batch  
기본 프로젝트

 Let's code

1. 프로젝트 생성
2. Docker로 batch mysql 구축



# Spring Batch 기본 프로젝트

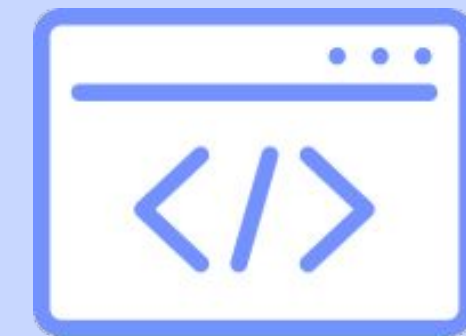
## 2 Spring Batch 실행 환경 - Batch 스케줄링

👋 Let's code

```
./gradlew bootJar
```

```
java -jar ${빌드한파일}.jar --spring.batch.job.names=${잡이름}
```

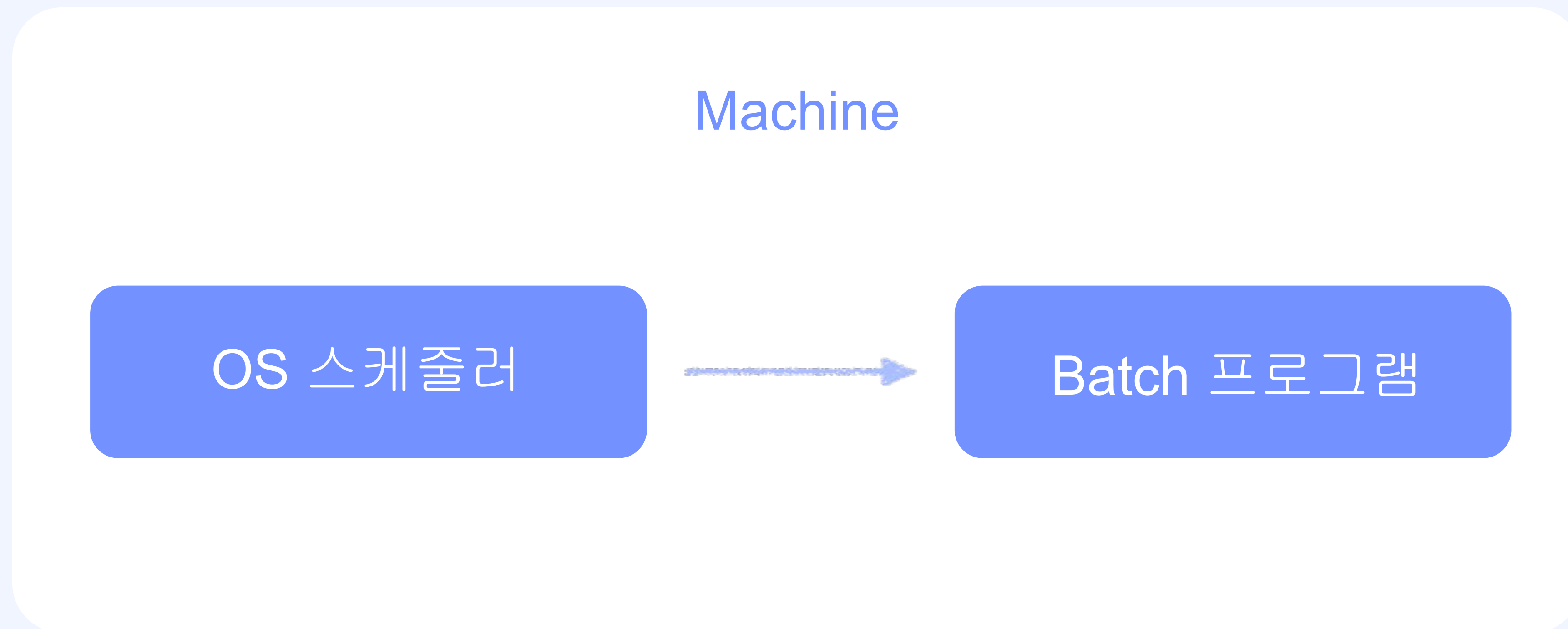
\*윈도우/맥에서 빌드한 파일이 있는 곳까지  
명령프롬프트/터미널을 통해 접근해서 실행



## Batch 실행 방식 - OS 스케줄러 이용

3.

Spring Batch  
기본 프로젝트



한 머신 내부에서 스케줄러를 통해 **Batch** 프로그램을 실행한다 - 리눅스  
crontab

- 실행 결과를 log 파일을 남겨 확인한다

## Batch 실행 방식 - Quartz Scheduler 이용

3.

Spring Batch  
기본 프로젝트



어플리케이션

Quartz Scheduler



Batch 로직

Quartz + Spring Batch 프레임워크를 함께 사용해 어플리케이션을 개발한다.

- 어플리케이션 내부에서 스케줄링하므로 **Batch** 실행이 빠르다.
- 스케줄링 데이터가 **DB**에 저장되어 **Admin**을 따로 만들어야 한다.

## Batch 실행 방식 - CI 프로그램 이용

3.

Spring Batch  
기본 프로젝트



# Jenkins

Master Node

Slave Node

Jenkins



Batch 프로그램

마스터에서 슬레이브로 명령을 전달해 배치 프로그램을 실행시킨다 - 젠킨스

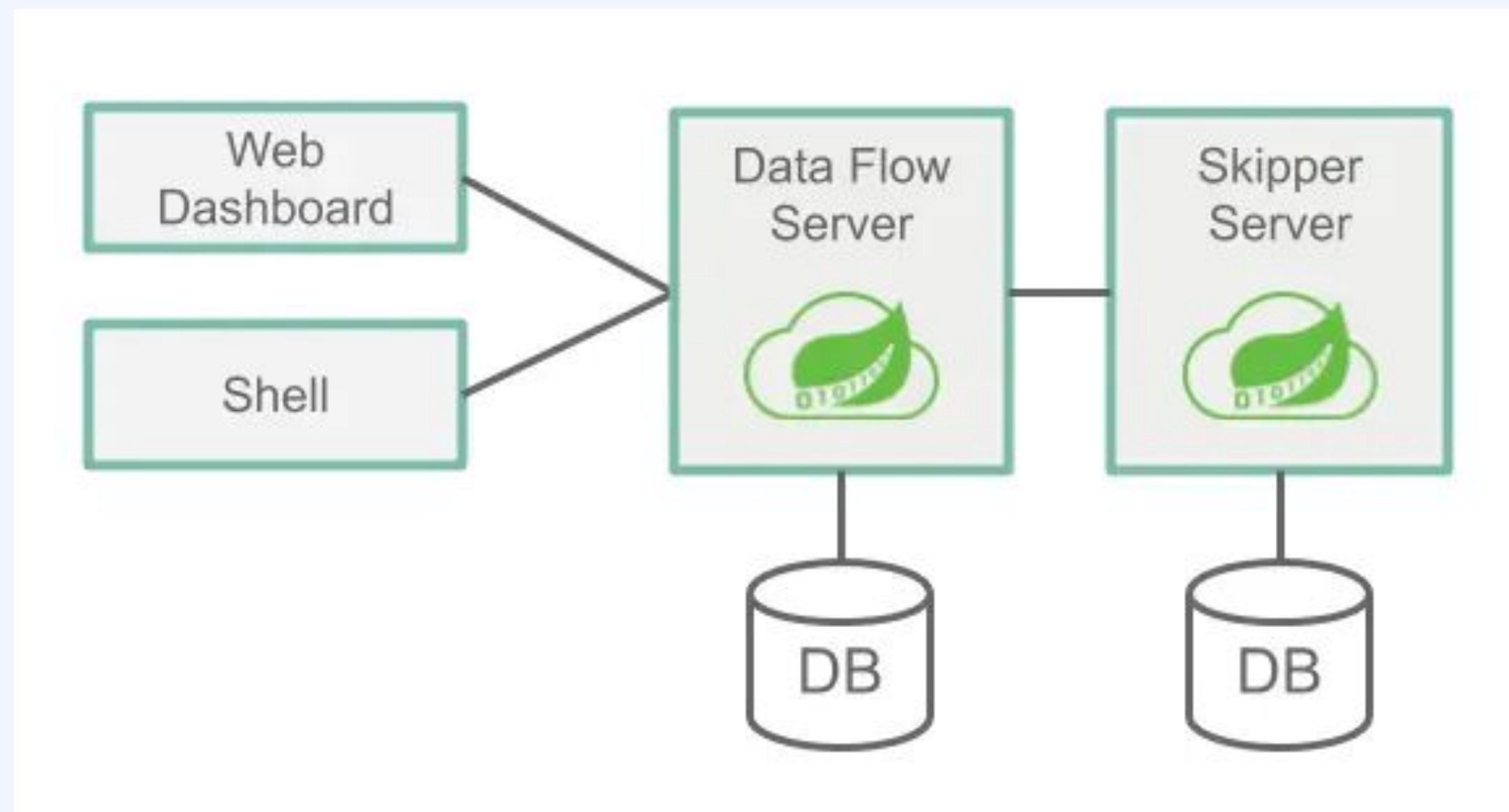
- 젠킨스에서 지원하는 스케줄링 기능을 통해 배치 프로그램을 실행시킬 수 있다.



## Batch 실행 방식 - Spring Cloud Data Flow

3.

Spring Batch  
기본 프로젝트



Spring Cloud Data Flow를 활용해 잡을 실행시킨다.

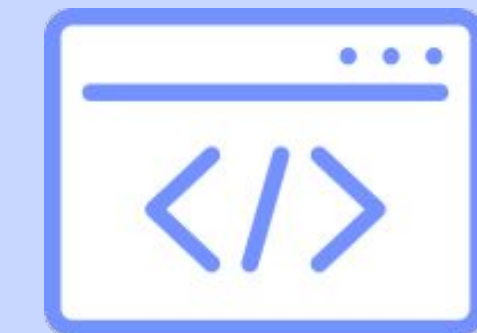
- Kubernetes 클러스터를 구축해야 한다.
- Admin이 제공되고, 잡의 실행을 연결할 수 있는 등 다양한 기능을 제공한다.

# Spring Batch 기본 프로젝트

3 데이터를 읽고, 처리하고, 쓰기

## Let's code

1. 데이터를 읽기 - `ItemReader`
2. 데이터를 처리하기 - `ItemProcessor`
3. 데이터를 쓰기 - `ItemWriter`



# Spring Batch 기본 프로젝트

## 4 Spring Batch 테스트

 Let's code

테스트 환경에서 Job을 실행하기 위한 설정

- H2
- @SpringBatchTest - JobLauncherTestUtils

