

2025년 상반기 K-디지털 트레이닝

# 스프링과 MySQL Database 연동

[KB] IT's Your Life



## 1 데이터베이스 및 계정 만들기

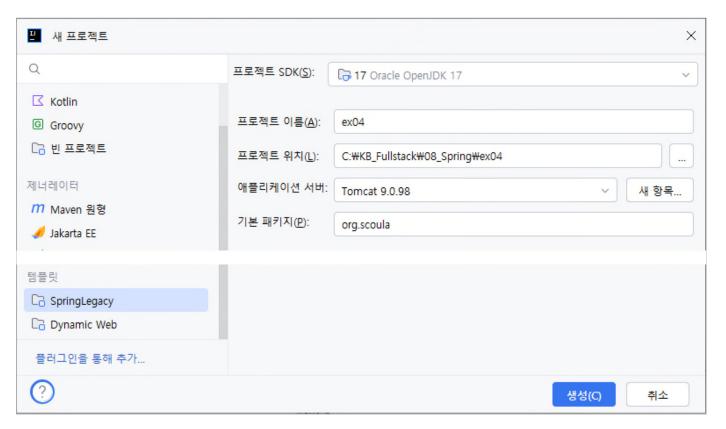
## 🗹 데이터베이스 생성 및 계정 생성, 권한 설정

```
create database scoula_db;
create user 'scoula'@'%' identified by '1234';
grant all privileges on scoula_db.* to 'scoula'@'%';
```

#### 🗸 프로젝트 만들기

Templates: SpringLegacy

o Project name: ex04

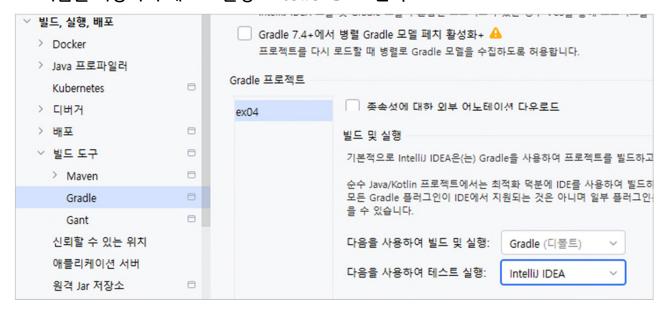


## settings.gradle

rootProject.name = 'ex04'

#### ☑ 설정

- O Enable Annotation Processor 활성화
- 빌드, 실행, 배포 > 빌드 도구 > Gradle
  - 다음을 사용하여 테스트 실행: IntelliJ IDEA 선택



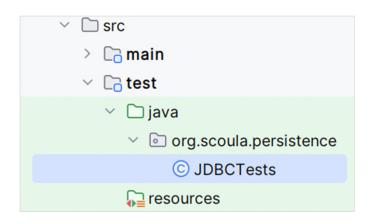
## build.gradle

// 데이터베이스 implementation 'com.mysql:mysql-connector-j:8.1.0'

→ gradle sync

#### JDBC 테스트 코드

- o src/test/java
  - org.scoula.persistence
    - JDBCTests.java



## 

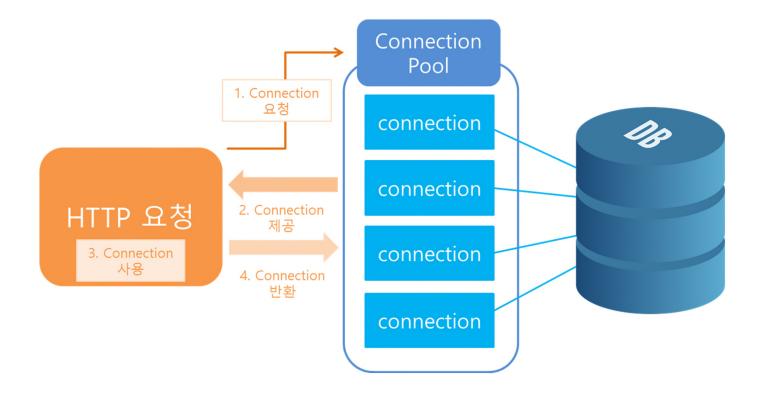
```
package org.scoula.persistence;
import lombok.extern.log4j.Log4j2;
import org.junit.jupiter.api.BeforeAll;
import org.junit.jupiter.api.DisplayName;
import org.junit.jupiter.api.Test;
import java.sql.Connection;
import java.sql.DriverManager;
import static org.junit.jupiter.api.Assertions.fail;
@Log4j2
public class JDBCTest {
  @BeforeAll
  public static void setup() {
    try {
      Class.forName("com.mysql.cj.jdbc.Driver");
    } catch(Exception e) {
      e.printStackTrace();
```

#### 

## 4 커넥션 풀 설정

#### DataSource

Connection Pool



#### 4 <mark>커넥션 풀 설정</mark>

#### ☑ 라이브러리 추가와 DataSource 설정

## build.gradle

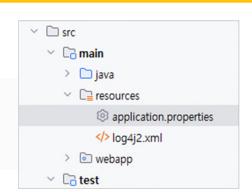
```
// 데이터베이스
implementation 'com.mysql:mysql-connector-j:8.1.0'
implementation 'com.zaxxer:HikariCP:2.7.4'
```

→ gradle sync

## 4 **커넥션 풀 설정**

## resources/application.properties

jdbc.driver=com.mysql.cj.jdbc.Driver jdbc.url=jdbc:mysql://127.0.0.1:3306/scoula\_db jdbc.username=scoula jdbc.password=1234



#### 4 <mark>커넥션 풀 설</mark>정

#### ☑ .properties에서 값 읽어 오기

- @PropertySource({"properties 경로 문자열"})
  - 클래스 레벨 어노테이션
  - 사용할 .properties 경로를 지정
  - 예@PropertySource({"classpath:/application.properties"})

#### ○ @Value("\${키:기본값}")

- 필드 레벨 어노테이션
- 기본값은 생략 가능
- 예@Value("\${jdbc.driver}") String driver;

## RootConfig.java

```
package org.scoula.config;
import com.zaxxer.hikari.HikariConfig;
import com.zaxxer.hikari.HikariDataSource;
import org.apache.ibatis.session.SqlSessionFactory;
import org.mybatis.spring.SqlSessionFactoryBean;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.context.ApplicationContext;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.context.annotation.PropertySource;
import org.springframework.jdbc.datasource.DataSourceTransactionManager;
import javax.sql.DataSource;
@Configuration
@PropertySource({"classpath:/application.properties"})
public class RootConfig {
  @Value("${jdbc.driver}") String driver;
  @Value("${idbc.url}") String url;
  @Value("${jdbc.username}") String username;
  @Value("${jdbc.password}") String password;
```

## RootConfig.java

```
@Bean
public DataSource dataSource() {
  HikariConfig config = new HikariConfig();
 config.setDriverClassName(driver);
 config.setJdbcUrl(url);
 config.setUsername(username);
 config.setPassword(password);
 HikariDataSource dataSource = new HikariDataSource(config);
 return dataSource;
```

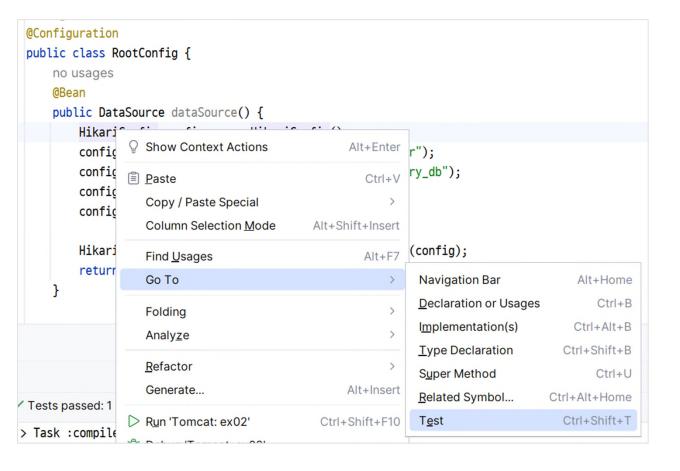
#### 4 <mark>커넥션 풀 설정</mark>

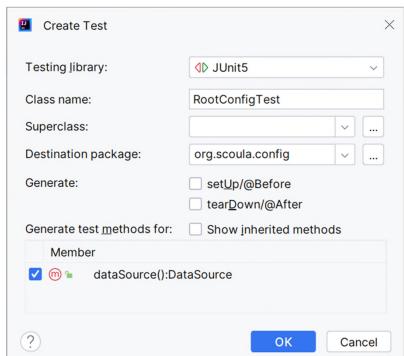
#### 라이브러리 추가와 DataSource 설정



### 4 커넥션 풀 설정

### ☑ RootConfig 테스트 클래스 만들기





## test ∷ config.DataSourceTests.java

```
@ExtendWith(SpringExtension.class)
@ContextConfiguration(classes= {RootConfig.class})
@Log4j2
class RootConfigTest {
  @Autowired
  private DataSource dataSource;
  @Test
  @DisplayName("DataSource 연결이 된다.")
  public void dataSource() throws SQLException {
    try(Connection con = dataSource.getConnection()){
      log.info("DataSource 준비 완료");
      log.info(con);
                                        G G B - V O F E O O 7 G :
                                                              ✓ 테스트 통과: 1 /1개 테스트 - 35ms

✓ RootConfigTest (org.scoula.co 35ms

                                       ✓ DataSource 연결이 된다. 35ms
                                                                INFO org.scoula.config.RootConfigTest(dataSource:30) - DataSource 준비 완료
                                                                INFO org.scoula.config.RootConfigTest(dataSource:31) - HikariProxyConnection@1263841085
                                                                  wrapping com.mysql.cj.jdbc.ConnectionImpl@7bb35cc6
```