



[한국ICT인재개발원] 스프링 프레임워크

5. 스프링 CRUD와 테스트

前) 광고데이터 분석 1년

前) IT강의 경력 2년 6개월

前) 머신러닝을 활용한 데이터 분석 프로젝트반 운영 1년

前) 리그오브 레전드 데이터 분석 등...

現) 국비반 강의 진행중

```
11 <java-version>1.8</java-version>
12 <org.springframework-version>5.0.7.RELEASE</org.
13 <org.aspectj-version>1.6.10</org.aspectj-versior
14 <org.slf4j-version>1.6.6</org.slf4j-version>
--
```

```
95 <groupId>javax.servlet</groupId>
96 <artifactId>javax.servlet-api</ar
97 <version>3.1.0</version>
98 <scope>provided</scope>
```

```
112 <!-- Test -->
113 <dependency>
114     <groupId>junit</groupId>
115     <artifactId>junit</artif
116     <version>4.12</version>
117     <scope>test</scope>
```

```
140 <configuration>
141     <source>1.8</s
142     <target>1.8</t
143     <compilerArgun
144     <showWarnings>
```

Pom.xml 내부 수치변경을 진행합니다.
위 코드 라인과 실제 수정값을 보고 그대로 적용해주세요.



1. Spring TestContext Framework

org.springframework » spring-test

Spring TestContext Framework

Last Release on Feb 16, 2021

spring-test 5.0.7.RELEASE버전



1. HikariCP

com.zaxxer » HikariCP

Ultimate JDBC Connection Pool

Last Release on Mar 3, 2021

hikariCP 4.0.3버전



1. Log4Jdbc Log4j2 JDBC 4

org.bgee.log4jdbc-log4j2 » log4jdbc-log4j2-jdbc4

Log4Jdbc Log4j2 JDBC 4

Last Release on Dec 12, 2013

log4jdbc-log4j2-jdbc4 1.16버전
(한 칸 아래 JDBC 4.1 과 혼동X)



1. Spring JDBC

org.springframework » spring-jdbc

Spring JDBC

Last Release on Feb 16, 2021

spring-jdbc 5.0.7.RELEASE버전



1. MyBatis

org.mybatis » mybatis

The MyBatis SQL mapper framework
XML descriptor or annotations. E

Last Release on Oct 6, 2020

mybatis 3.5.6버전



1. Project Lombok

org.projectlombok » lombok

Spice up your java: Automatic

Last Release on Jan 28, 2021

Lombok 1.18.18버전



1. Spring Transaction

org.springframework » spring-tx

Spring Transaction

Last Release on Feb 16, 2021

spring-tx 5.0.7.RELEASE버전



1. MyBatis Spring

org.mybatis » mybatis-spring

An easy-to-use Spring bridge for

Last Release on Nov 14, 2020

mybatis-spring 2.0.6버전



1. MySQL Connector/J

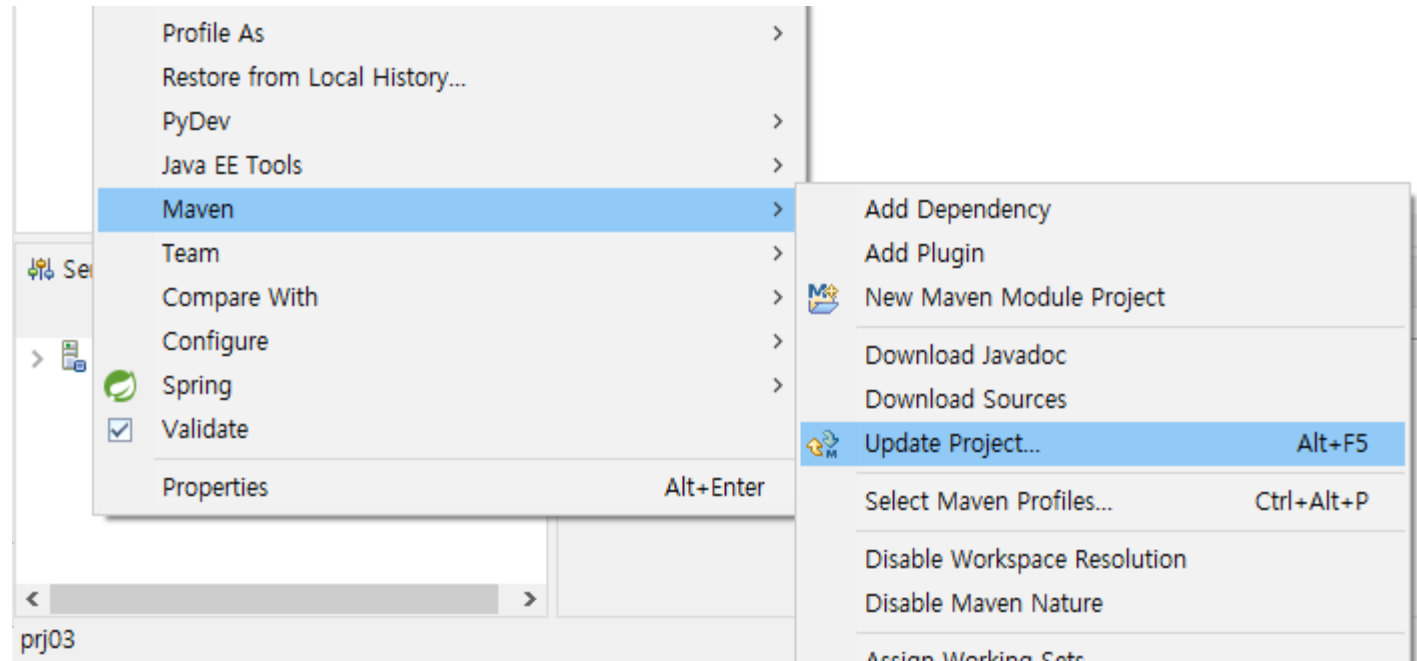
mysql » mysql-connector-java

JDBC Type 4 driver for MySQL

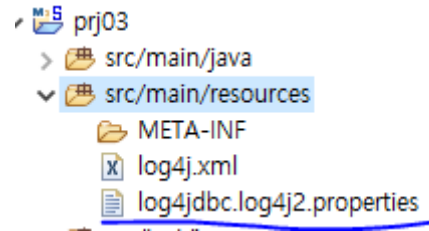
Last Release on Jan 17, 2021

mysql-connector-java 8.0.21
(설치된 mysql버전과 동일하게)

Mvnrepository에서 검색어를 정확히 검색해 맞는 버전을
Pom.xml의 dependencies에 추가해주세요



프로젝트 파일 우클릭 -> Maven -> Update Project를 클릭해 자바버전을 1.8버전으로 업데이트합니다.



```
log4jdbc.log4j2.properties  ⌕  
1 log4jdbc.spylogdelegator.name=net.sf.log4jdbc.log.slf4j.Slf4jSpyLogDelegator
```

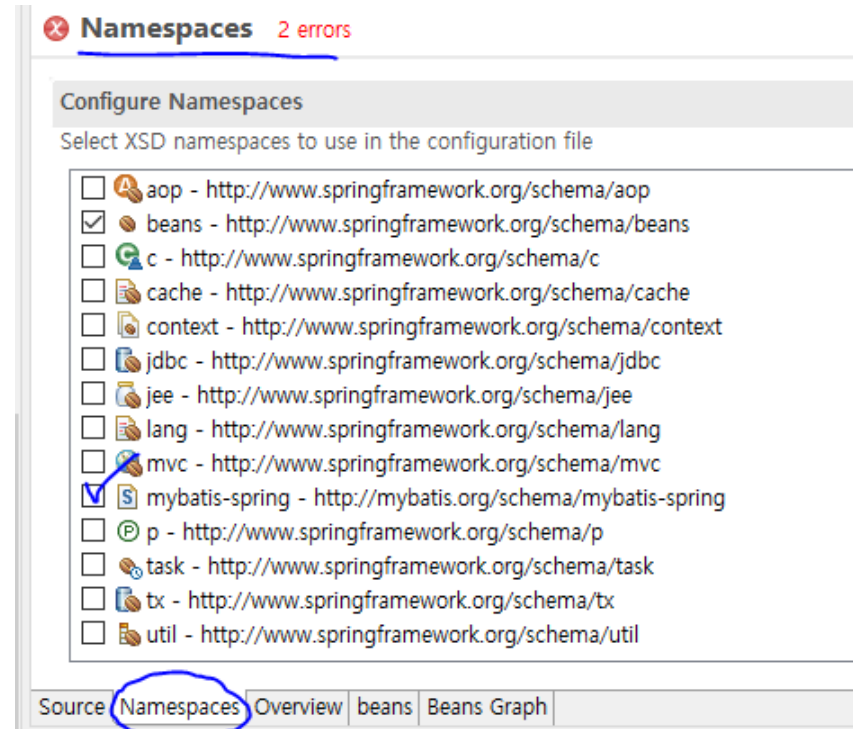
프로젝트의 src/main/resources 폴더에
log4jdbc.log4j2.properties 라는 이름으로 폴더를 생성하고
내용을 위와 같이 적습니다.

```
<bean id="hikariConfig" class="com.zaxxer.hikari.HikariConfig">
  <property name="driverClassName"
    value="net.sf.log4jdbc.sql.jdbcapi.DriverSpy"></property>
  <property name="jdbcUrl"
    value="jdbc:log4jdbc:mysql://localhost:3306/ict3?serverTimezone=UTC">
    </property>
  <property name="username" value="root"></property>
  <property name="password" value="1111"></property>
</bean>

<bean id="dataSource"
  class="com.zaxxer.hikari.HikariDataSource" destroy-method="close">
  <constructor-arg ref="hikariConfig"></constructor-arg>
</bean>
```

Root-context.xml로 이동해 hikariCP 관련 설정과 마이바티스 관련 설정을 진행합니다.

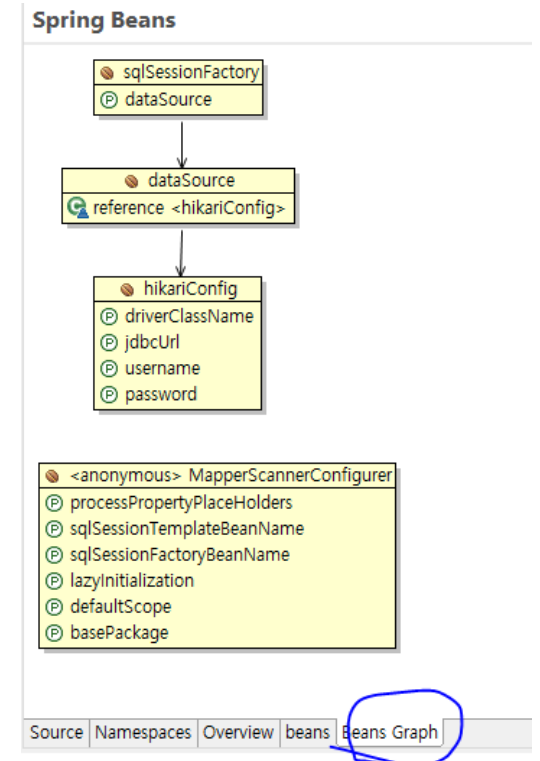
먼저 hikariCP의 HikariConfig와 HikariDataSource 객체를 bean-container에 넣어보겠습니다.



다음으로 root-context.xml 파일 하단의 Namespaces 탭을 클릭하고 mybatis-spring을 체크합니다.


```
<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource"></property>
</bean>

<mybatis-spring:scan base-package="org.ict.mapper"/>
```



이후 위와 같이 `sqlSessionFactory` 객체를 생성한 다음 밑에 마이바티스 스캔을 하고, `root-context.xml` 하단의 **Beans Graph** 탭을 클릭해서 제대로 객체들이 들어왔나 체크합니다.

```
<!-- 한글 인코딩 설정 -->
<filter>
    <filter-name>encoding</filter-name>
    <filter-class>
        org.springframework.web.filter.CharacterEncodingFilter
    </filter-class>
    <init-param>
        <param-name>encoding</param-name>
        <param-value>UTF-8</param-value>
    </init-param>
</filter>

<filter-mapping>
    <filter-name>encoding</filter-name>
    <servlet-name>appServlet</servlet-name>
</filter-mapping>
```

그리고, 한글인코딩을 처리하기 위해
Web.xml 파일의 <web-app> 태그 사이에 위와 같이 적어넣습니다.

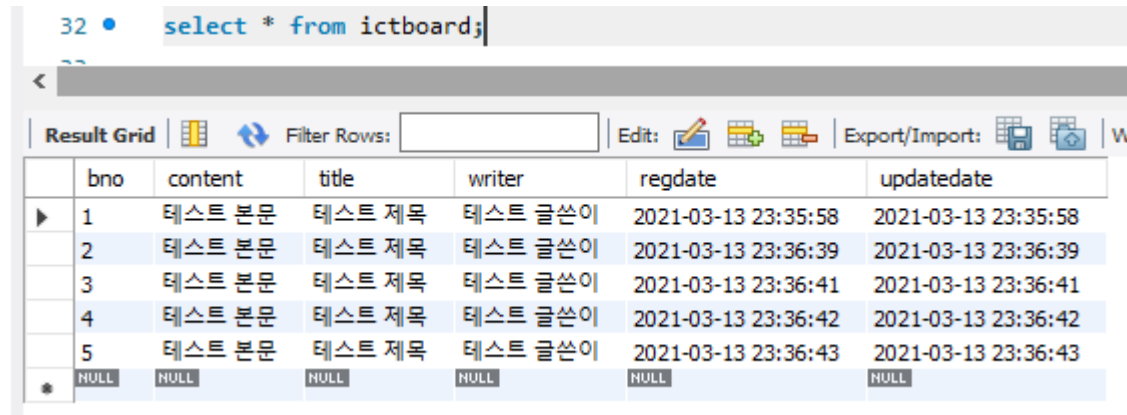
```
create table ictboard(  
    bno int not null auto_increment,  
    content varchar(1000) not null,  
    title varchar(100) not null,  
    writer varchar(20) not null,  
    regdate timestamp default now(),  
    updatedate timestamp default now(),  
    primary key(bno)  
);
```

MySQL 테이블은 위와 같이 구성합니다.

글번호, 본문, 제목, 글쓴이, 등록날짜, 수정날짜를 가지며
글번호를 프라이머리키로 가집니다.

Bno의 auto_increment는 글 번호를 1부터 자동으로 insert구문 실행시마다
1씩 증가시켜서 입력하라는 옵션입니다.

```
insert into ictboard (content, title, writer) values ('테스트 본문', '테스트 제목', '테스트 글쓴이');
```



32 • select * from ictboard;

Result Grid | Filter Rows: | Edit: | Export/Import: |

	bno	content	title	writer	regdate	updatedate
▶	1	테스트 본문	테스트 제목	테스트 글쓴이	2021-03-13 23:35:58	2021-03-13 23:35:58
	2	테스트 본문	테스트 제목	테스트 글쓴이	2021-03-13 23:36:39	2021-03-13 23:36:39
	3	테스트 본문	테스트 제목	테스트 글쓴이	2021-03-13 23:36:41	2021-03-13 23:36:41
	4	테스트 본문	테스트 제목	테스트 글쓴이	2021-03-13 23:36:42	2021-03-13 23:36:42
	5	테스트 본문	테스트 제목	테스트 글쓴이	2021-03-13 23:36:43	2021-03-13 23:36:43
★	NULL	NULL	NULL	NULL	NULL	NULL

Insert 구문으로 5개의 글을 집어넣고 select 구문을 이용해 제대로 자료가 들어갔는지 체크합니다.

Bno에 걸린 auto_increment 제약으로 인해 bno에도 값을 넣지 않아도 됩니다.

기본 설정이 끝났습니다.