KB금융그룹 | 국민의 평생 금융파트너

2024년 상반기 K-디지털 트레이닝

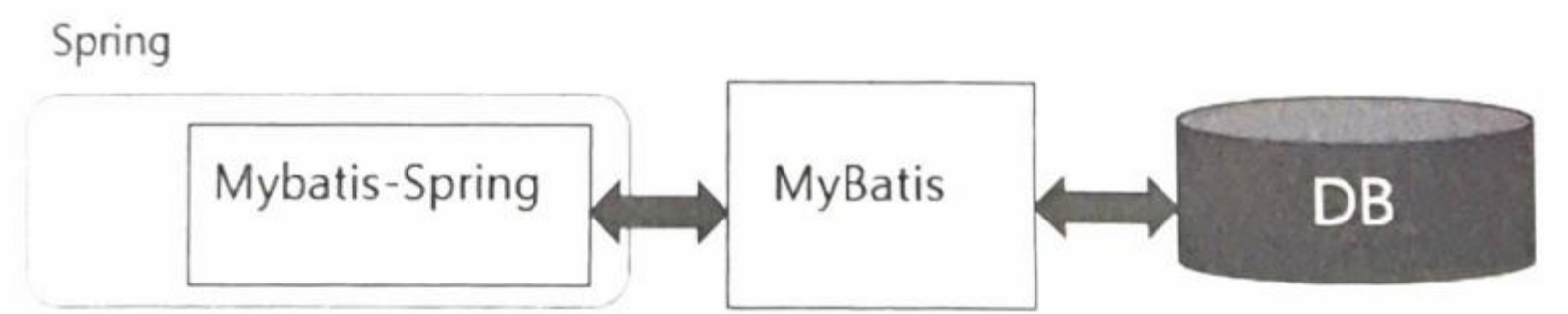
스프링과 MySQL Database 연동

[KB] IT's Your Life

KB 국민은행

MyBatis

- https://mybatis.org/mybatis-3/
- SQL 매핑 프레임워크



MyBatis 관련 라이브러리 추가

- spring-jdbc/spring-tx
  - 스프링에서 데이터베이스 처리와 트랜잭션 처리
- mybatis/mybatis-spring
  - MyBatis와 스프링 연동용 라이브러리

build.gradle

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

implementation "org.springframework:spring-tx:${springVersion}"
implementation "org.springframework:spring-jdbc:${springVersion}"

implementation 'org.mybatis:mybatis:3.4.6'
implementation 'org.mybatis:mybatis-spring:1.3.2'
…
```

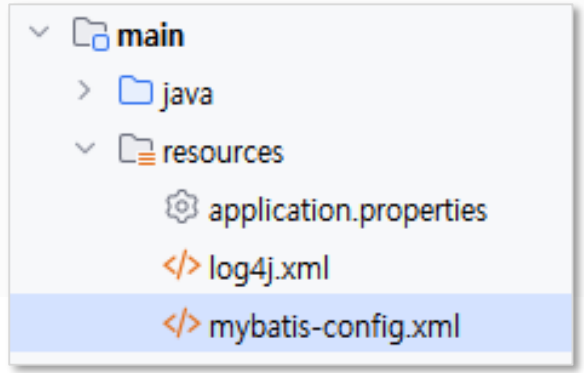
→ gradle sync

SQLSessionFactory

- SQLSession
  - Connection 생성, SQL 전달, 결과 리턴 등 처리
- SQLSessionFactory
  - SQLSession 객체 생성

resources/mybatis-config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE configuration
        PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
        "http://mybatis.org/dtd/mybatis-3-config.dtd">

<configuration>

</configuration>
```



config.RootConfig.java

```
…
@Configuration
public class RootConfig {
  …
  @Autowired
  ApplicationContext applicationContext;

  @Bean
  public SqlSessionFactory sqlSessionFactory() throws Exception {
     SqlSessionFactoryBean sqlSessionFactory = new SqlSessionFactoryBean();
     sqlSessionFactory.setConfigLocation(
              applicationContext.getResource("classpath:/mybatis-config.xml"));
     sqlSessionFactory.setDataSource(dataSource());
     return (SqlSessionFactory) sqlSessionFactory.getObject();
  }

  @Bean
  public DataSourceTransactionManager transactionManager(){
    DataSourceTransactionManager manager = new DataSourceTransactionManager(dataSource());
    return manager;
  }
}
```

test:: RootConfigTest.java

```
class RootConfigTest {
    …
    @Autowired
    private SqlSessionFactory sqlSessionFactory;
    …
    @Test
    public void testSqlSessionFactory() {
        try (
           SqlSession session = sqlSessionFactory.openSession();
           Connection con = session.getConnection();
        ) {
            log.info(session);
            log.info(con);
        } catch (Exception e) {
            fail(e.getMessage());
        }
    }
}
```

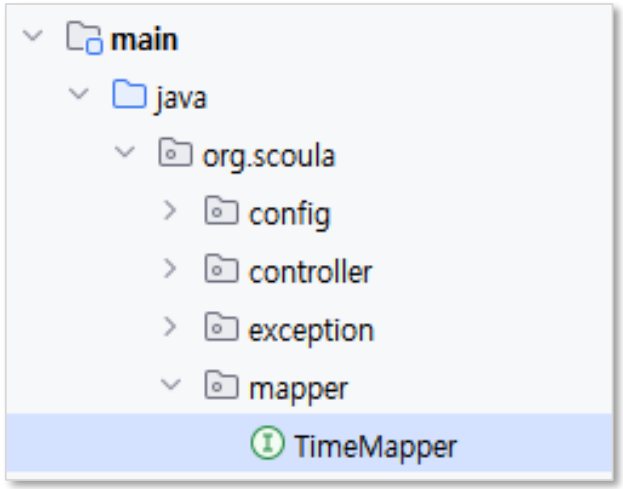
```
INFO : org.scoula.config.RootConfigTest - org.apache.ibatis.session.defaults.DefaultSqlSession@6f6621e3
INFO : org.scoula.config.RootConfigTest - HikariProxyConnection@967643830 wrapping
com.mysql.cj.jdbc.ConnectionImpl@4eb45fec
```

Mapper

- SQL과 그에 대한 처리를 지정하는 역할
- XML과 인터페이스+어노테이션 형태로 작성

Mapper 인터페이스

- org.scoula.mapper 패키지
  - TimeMapper 인터페이스 추가

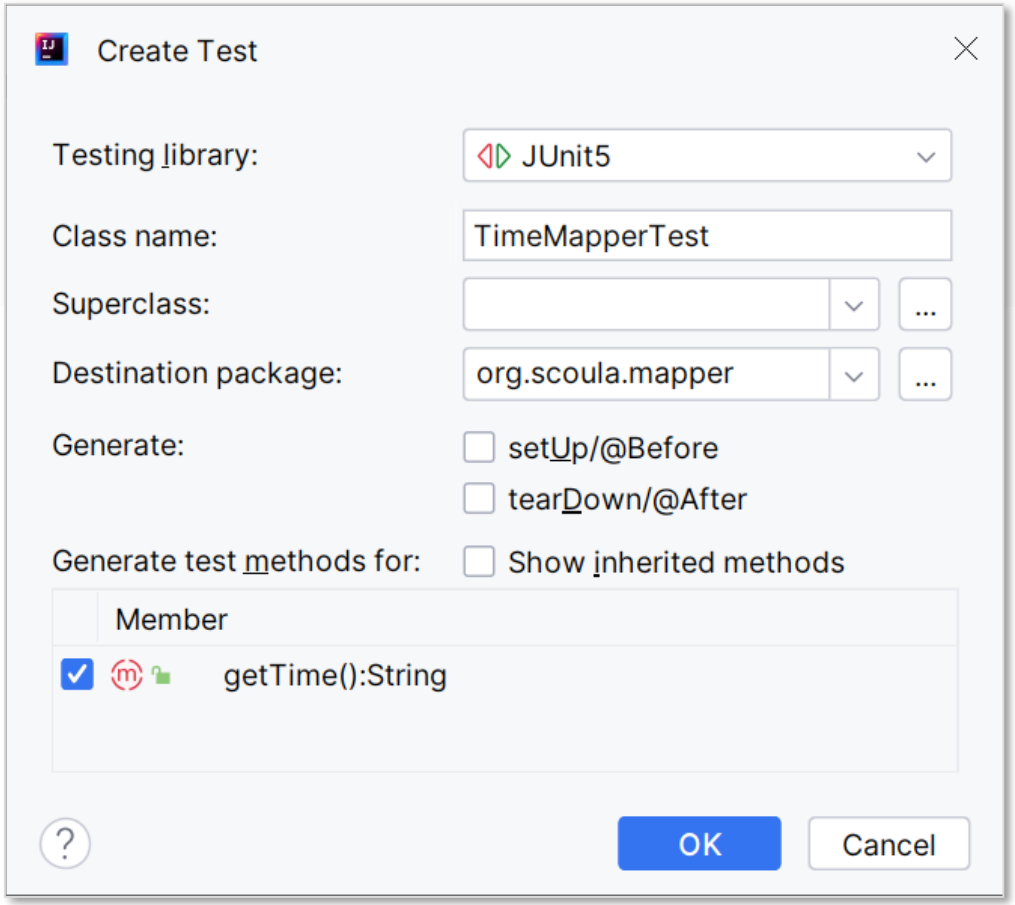


mapper.TimeMapper.java

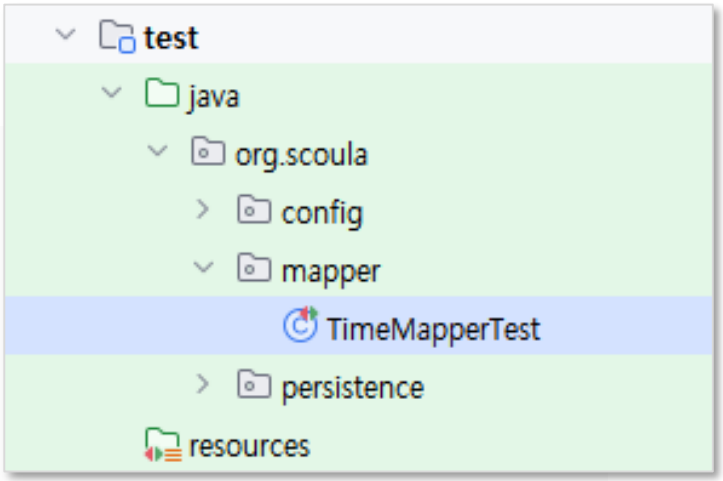
```
package org.scoula.mapper;

import org.apache.ibatis.annotations.Select;

public interface TimeMapper {
  @Select("SELECT sysdate()")
  public String getTime();

}
```

RootConfig.java – Mapper 설정

```
…
import org.mybatis.spring.annotation.MapperScan;
…

@Configuration
@MapperScan(basePackages  = {"org.scoula.mapper"})
public class RootConfig {

   …

}
```



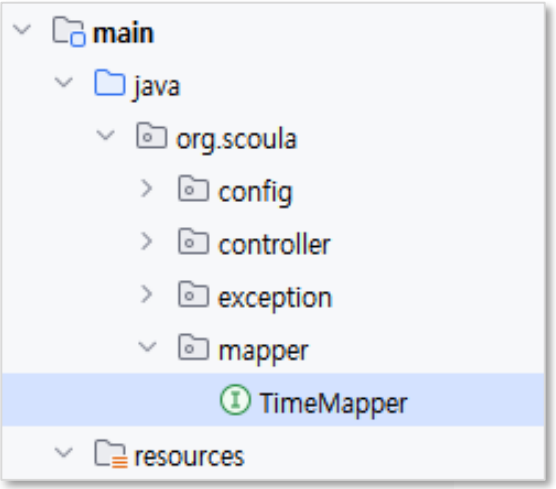
test :: mapper.TimeMapperTest.java




```
…

@ExtendWith(SpringExtension.class)
@ContextConfiguration(classes = { RootConfig.class })
@Log4j
public class TimeMapperTest {

  @Autowired
  private TimeMapper timeMapper;

  @Test
  @DisplayName("TimeMapper의 getTime()")
  public void getTime () {
    log.info(timeMapper.getClass().getName());
    log.info(timeMapper.getTime());
  }
}

INFO : org.scoula.mapper.TimeMapperTest - jdk.proxy3.$Proxy32
INFO : org.scoula.mapper.TimeMapperTest - 2023-11-01 17:20:49
```

TimeMapper.java

```
package org.scoula.mapper;

import org.apache.ibatis.annotations.Select;

public interface TimeMapper {
  @Select("SELECT sysdate FROM dual")
  public String getTime();

  public String getTime2();
}
```

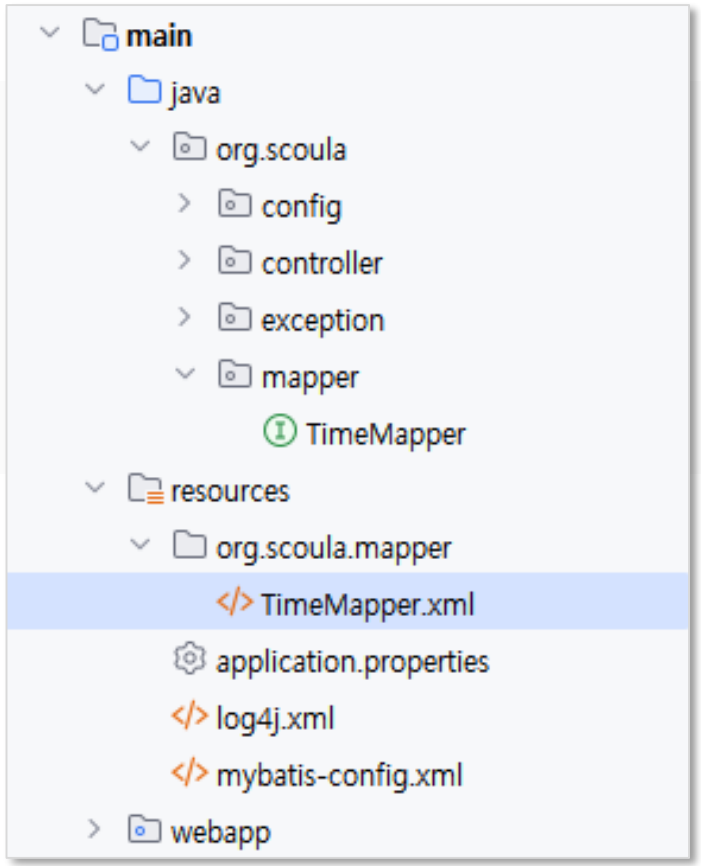


XML 매퍼와 같이 쓰기

- src/main/resources
  - org/scoula/mapper --> Mapper 인터페이스와 같은 패키지 경로여야 함
    - TimeMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="org.scoula.mapper.TimeMapper">

</mapper>
```

TimerMapper 인터페이스 경로



resources:: TimeMapper.xml

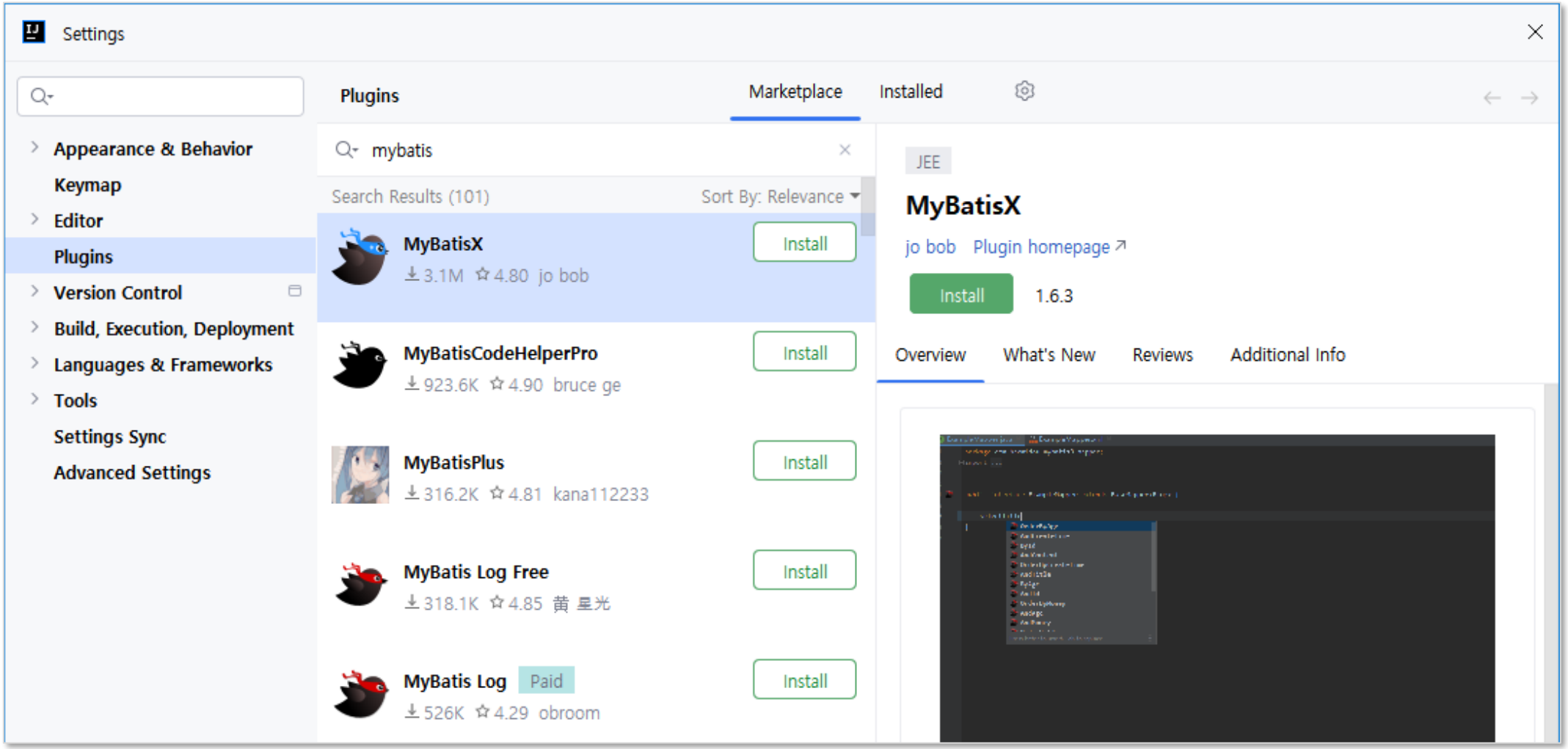
```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="org.scoula.mapper.TimeMapper">

  <select id="getTime2" resultType="string">
    SELECT sysdate()
  </select>
</mapper>
```

String getTime2()의 메서드명

String getTime2()의 리턴타입

MyBatisX 플러그인 설치

- mybatis 검색



resources:: TimeMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="org.scoula.mapper.TimeMapper">

</mapper>
```

TimeMapper.java

TimeMapper.xml로 이동

```
5  public interface TimeMapper {  2 usages
6      @Select("SELECT sysdate()")  1 usage
7      public String getTime();
8
9      public String getTime2();  1 usage
10 }
11
```

Statement with id="getTime2" not defined in mapper xml

Generate statement Alt+Shift+Enter More actions... Alt+Enter

org.scoula.mapper.TimeMapper

public String getTime2()

ex04.main

Alt+Enter

```
public String getTime2();  1 usage
```

Generate statement

Make 'getTime2()' default

[MybatisX] Generate new statement

Add Javadoc

Press Ctrl+Q to toggle preview

resources:: TimeMapper.xml

TimeMapper.java로 이동

getTime2() 선언으로 이동

```
<mapper namespace="org.scoula.mapper.TimeMapper">

    <select id="getTime2" resultType="java.lang.String"></select>
</mapper>
```

TimeMapper.java

Mapper xml의 getTime2로 이동

```
public interface TimeMapper {  2 usages
    @Select("SELECT sysdate()")  1 usage
    public String getTime();

    public String getTime2();  1 usage
}
```

test::TimeMapperTest.java

```
package org.scoula.persistence;
…
public class TimeMapperTest {
  …

  @Test
  @DisplayName("TimeMapper의 getTime2()")
  public void getTime2() {

    log.info("getTime2");
    log.info(timeMapper.getTime2());

  }
}
```

```
INFO : org.scoula.mapper.TimeMapperTest - getTime2
INFO : org.scoula.mapper.TimeMapperTest - 2023-11-01 17:21:58
```

log4jdbc-log4j2

- PreparedStatement로 SQL 처리시 ?에 실제 치환값을 출력해줌

build.gradle

```
 // logging
implementation 'org.slf4j:slf4j-api:2.0.9'
runtimeOnly 'org.slf4j:jcl-over-slf4j:2.0.9'
runtimeOnly 'org.slf4j:slf4j-log4j12:2.0.9'
implementation 'log4j:log4j:1.2.17'

implementation 'org.bgee.log4jdbc-log4j2:log4jdbc-log4j2-jdbc4:1.16'
implementation 'org.apache.logging.log4j:log4j-api:2.0.1'
implementation 'org.apache.logging.log4j:log4j-core:2.0.1'
```

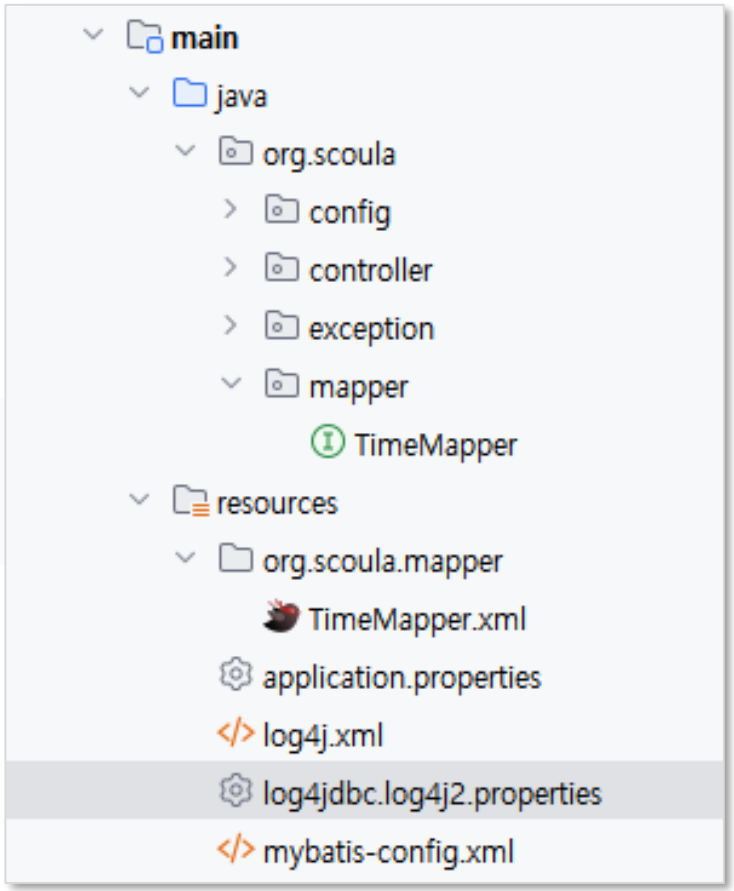
→ gradle sync

log4jdbc-log4j2설정

- src/main/resources
  - log4jdbc.log4j2.properties

resources/log4jdbc.log4j2.properties

```
log4jdbc.spylogdelegator.name=net.sf.log4jdbc.log.slf4j.Slf4jSpyLogDelegator
log4jdbc.auto.load.popular.drivers=false
log4jdbc.drivers=com.mysql.cj.jdbc.Driver
```



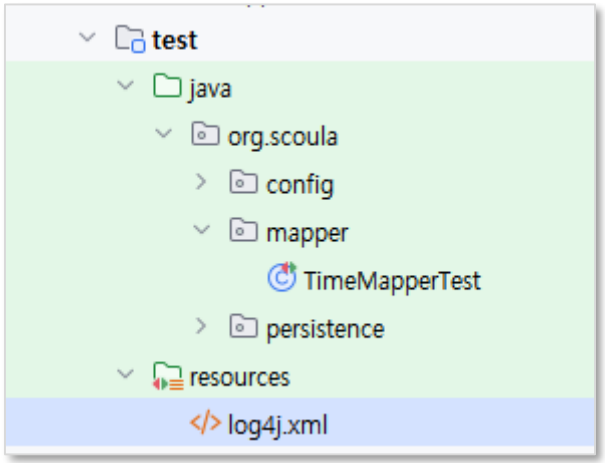
resources/application.properties

```
#driver=com.mysql.cj.jdbc.Driver
#url=jdbc:mysql://127.0.0.1:3306/scoula_db
jdbc.driver=net.sf.log4jdbc.sql.jdbcapi.DriverSpy
jdbc.url=jdbc:log4jdbc:mysql://localhost:3306/scoula_db
jdbc.username=scoula
jdbc.password=1234
```

로그의 레벨 설정

- 너무 많은 로그 메시지 출력 → 로그 레벨 이용하여 수정 가능
- 로그레벨
  - FATAL : 가장 크리티컬한 에러가 일어 났을 때만 로깅
  - ERROR : 일반 에러가 일어 났을 때만 로깅
  - WARN : 에러는 아니지만 주의할 필요가 있을 때만 로깅
  - INFO : 일반 정보를 나타낼 때 로깅 (INFO + ERROR)
  - DEBUG : 일반 정보를 상세히 나타낼 때 로깅 (디버깅 정보)
  - TRACE : 경로추적을 위해 사용

- src/test/resources
  - log4j.xml



test :: resources/log4j.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration PUBLIC "-//APACHE//DTD LOG4J 1.2//EN" "log4j.dtd">
<log4j:configuration xmlns:log4j="http://jakarta.apache.org/log4j/">

  …

  <!-- Root Logger -->
  <root>
    <priority value="info" />
    <appender-ref ref="console" />
  </root>

</log4j:configuration>
```

테스트 로그 출력

```
INFO : com.zaxxer.hikari.HikariDataSource - HikariPool-1 - Starting...
INFO : jdbc.connection - 1. Connection opened
…
INFO : jdbc.audit - 1. PreparedStatement.new PreparedStatement returned
INFO : jdbc.audit - 1. Connection.prepareStatement(SELECT sysdate()) returned
net.sf.log4jdbc.sql.jdbcapi.PreparedStatementSpy@34d52ecd
INFO : jdbc.sqlonly - SELECT sysdate()

INFO : jdbc.sqltiming - SELECT sysdate()
 {executed in 21 msec}
…
INFO : jdbc.resultsettable -
|--------------------|
|sysdate()           |
|--------------------|
|2023-11-01 17:26:25 |
|--------------------|
….
```

로그의 레벨 설정

log4jdbc 출력 로그 조절

- <logger> 태그 지정

핵심 로그만 출력하기

- jdbc.sqlonly

→ 로그 레벨 info로 설정

테스트용 log4j.xml

- jdbc.audit
- jdbc.connection
- jdbc.resultset

→ 해당 패키지의 로그 레벨을 warn으로 설정

main :: resources/log4j.xml

```
    …
    <logger name="jdbc.audit">
        <level value="warn" />
    </logger>

    <logger name="jdbc.resultset">
        <level value="warn" />
    </logger>

    <logger name="jdbc.connection">
        <level value="warn" />
    </logger>

    <logger name="jdbc.sqlonly">
        <level value="info" />
    </logger>

    <!-- Root Logger -->
    <root>
        <priority value="info" />
        <appender-ref ref="console" />
    </root>

</log4j:configuration>
```

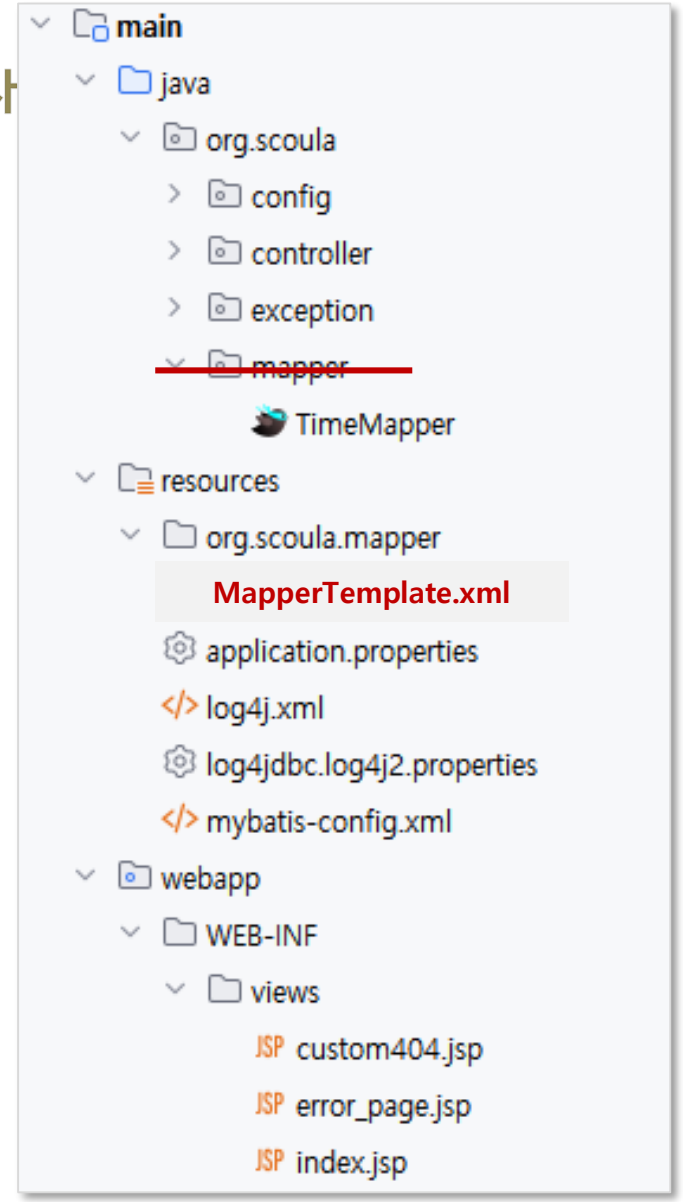
test :: resources/log4j.xml

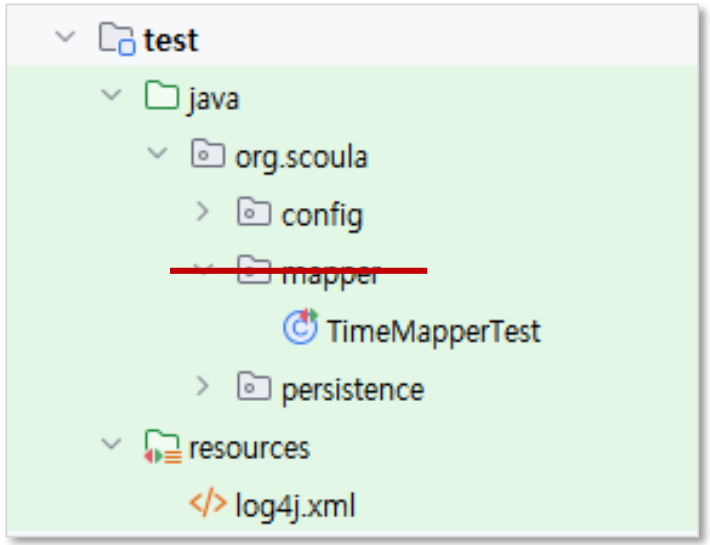
```
  …
    <logger name="jdbc">
        <level value="warn" />
    </logger>

    <logger name="jdbc.sqlonly">
        <level value="info" />
    </logger>


    <!-- Root Logger -->
    <root>
        <priority value="info" />
        <appender-ref ref="console" />
    </root>

</log4j:configuration>
```

프로젝트 템플릿 만들기

- ex04 → MyBatisSpringLegacy로 복사

- 파일 정리
  - mapper 관련 패키지/파일 삭제
  - mapper 관련 테스트 파일 삭제
  - TimeMapper.xml
    → MapperTemplate.xml





settings.gradle

```
rootProject.name = "MyBatisSpringLegacy"
```

RootConfig.java

```
@Configuration
@PropertySource({"classpath:/application.properties"})
// @MapperScan(basePackages  = {})
public class RootConfig {
  …
}
```

resources/org/scoula/mapper/MapperTemplate.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
        PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
        "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="">

</mapper>
```

프로젝트 템플릿 만들기

- File > New Project Setup > Save Project As Template...