

# MyBatis in Spring Boot

# Introduction

MyBatisとは、DBとの対応づけを自動で行なってくれる、O/Rマッピングフレームワークである。

SQL文を、開発者が記述出来るという特徴がある。



# 目次

## 1, MyBatis

1-1, メリット

1-2, 挙動

## 2, サンプル

2-1, プロジェクトの作成

2-2, プロジェクト実行

# 1, MyBatis

## 1-1, メリット

**SQLの実行結果**と、Javaオブジェクトを対応付けることが出来る。

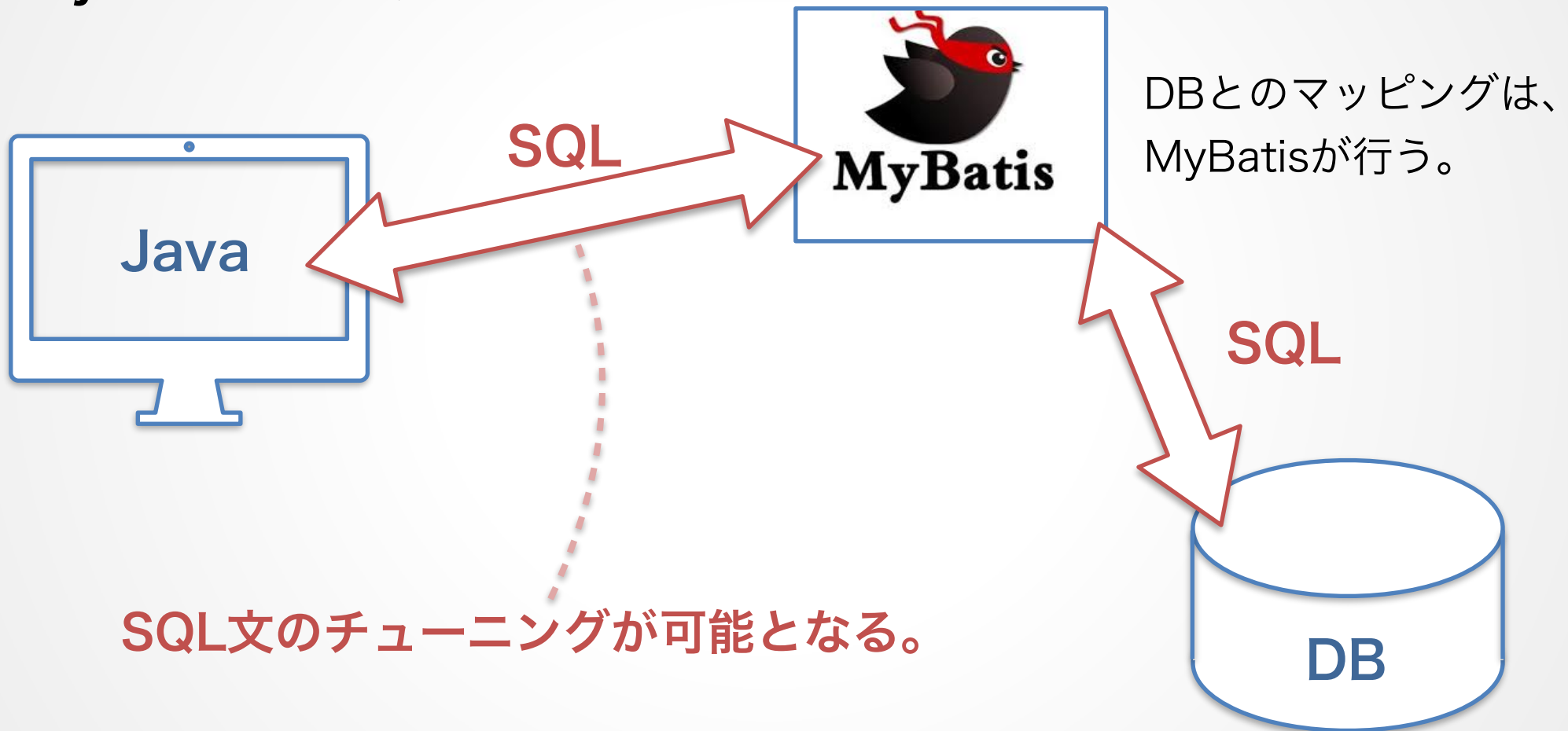
それにより、複雑なテーブル構造を持つDBに対しても、開発者がそれに対応したSQL文を記述すれば、どんなDB構成に対しても対応することが出来る。

対して、SQL文を全て自動生成にしてしまうと、複雑なテーブル構成に対して対応できる限界が発生してしまう。

# 1, MyBatis

## 1-2, 挙動

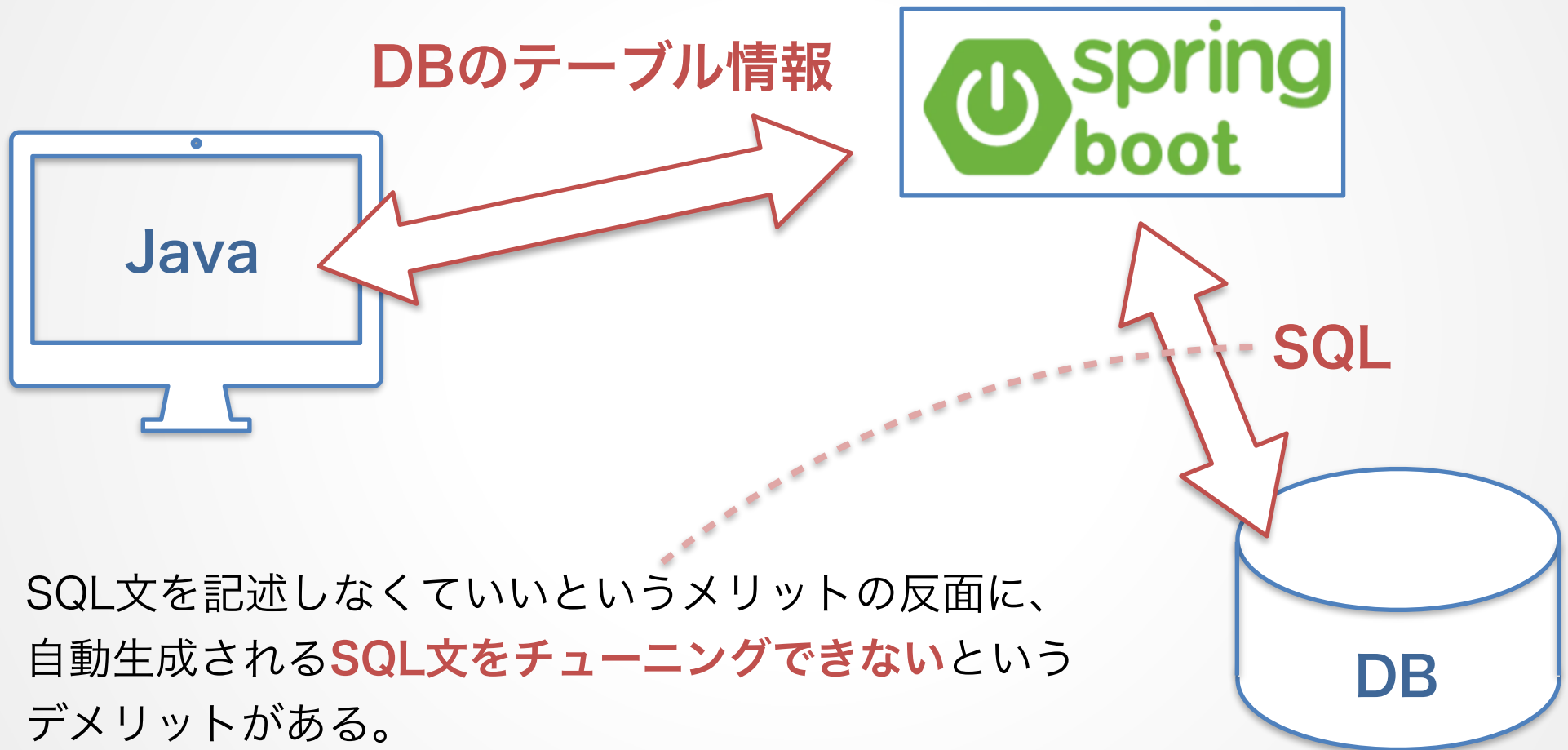
### MyBatis使用の場合



# 1, MyBatis

## 1-2, 挙動

### MyBatis未使用の場合



# 2, サンプル

## 2-1, プロジェクトの作成

### 1, Spring スターター・プロジェクト作成

名前 : SpringSampleMyBatis

型 : Gradle (Buildship 3.x)

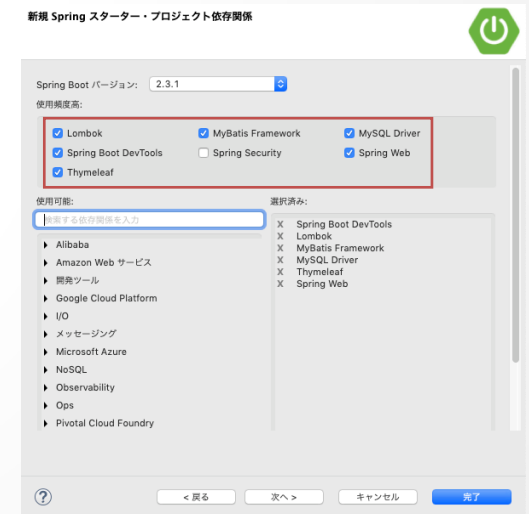
Java バージョン : 11



### 2, 新規 Spring スターター・プロジェクト依存関係

下記の項目をチェック。

- Spring Boot DevTools
- Lombok
- MyBatis Framework
- MySQL Driver
- Thymeleaf
- Spring Web



※画面上部に表示されていない場合は、検索欄を使用。

## 2, サンプル

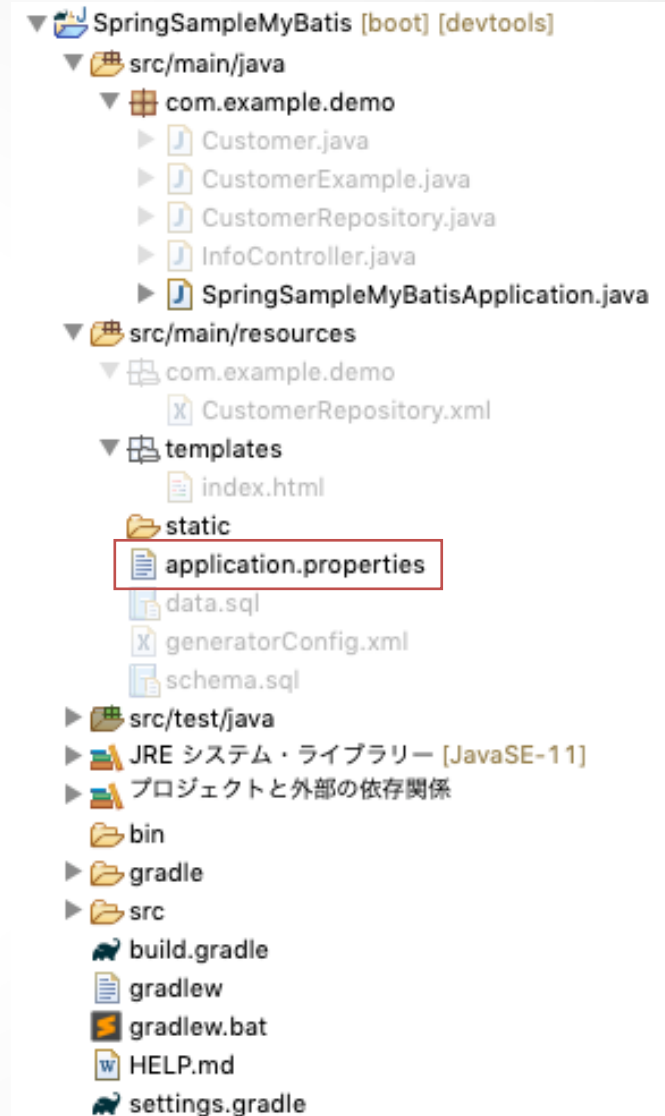
### 2-1, プロジェクトの作成

### 3, propertiesファイル修正

対象ファイル

application.properties

※修正内容は、サンプルファイルよりコピー&ペースト。





## 2, サンプル

### 2-1, プロジェクトの作成

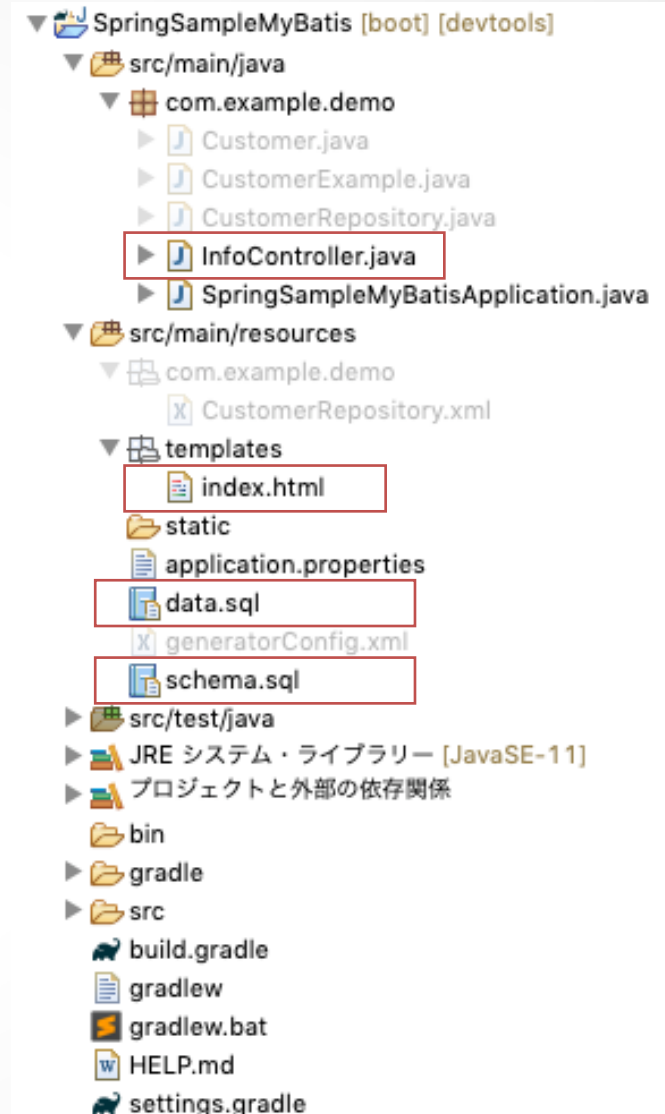
#### 4, 追加ファイル

Javaクラスファイル  
InfoController.java

HTMLファイル  
index.html

SQLファイル  
data.sql  
schema.sql

※ファイル内容は、サンプルファイルよりコピー&ペースト。



## 2, サンプル

### 2-1, プロジェクトの作成

#### 5, MyBatis Generetor ファイルの作成

MySQLサーバーを起動

scr/main/resourcesを右クリック

新規

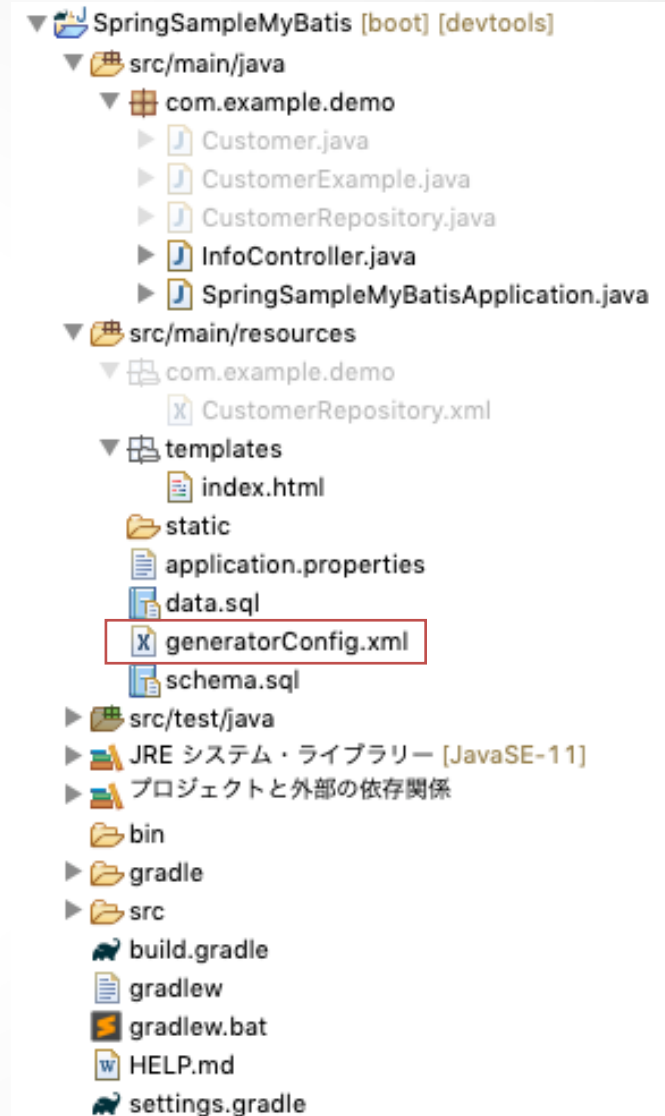
→その他

→MyBatis

→MyBatis Generator Configuration File

ファイル名 : generatorConfig.xml

※ファイル内容は、サンプルファイルよりコピー&ペースト。



## 2, サンプル

### 2-1, プロジェクトの作成

#### 5, MyBatis Generetor の実行

パッケージ名を右クリック

→実行

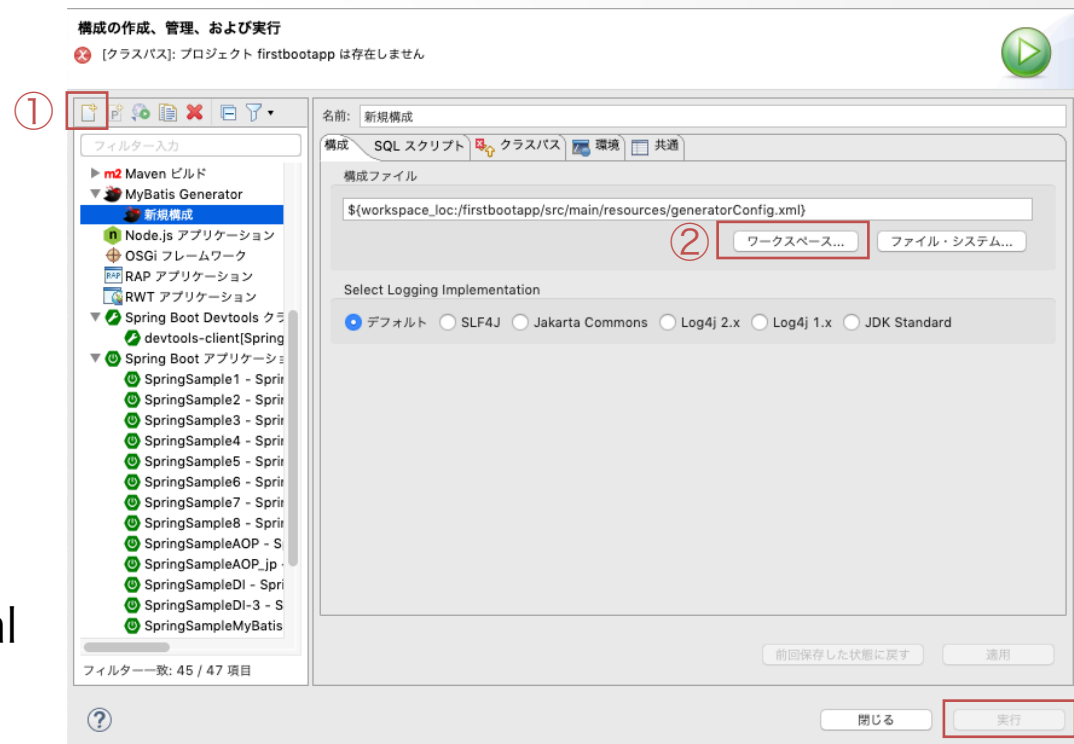
→実行の構成

→MyBatis Generator

①をクリックして、新規作成。

②をクリックして、generatorConfig.xml  
を設定。

「実行」をクリック。



## 2, サンプル

### 2-1, プロジェクトの作成

#### 6, MyBatis Generetor ファイル生成の確認

以下の4ファイルが生成されたことを確認する。

Javaクラスファイル

Customer.java

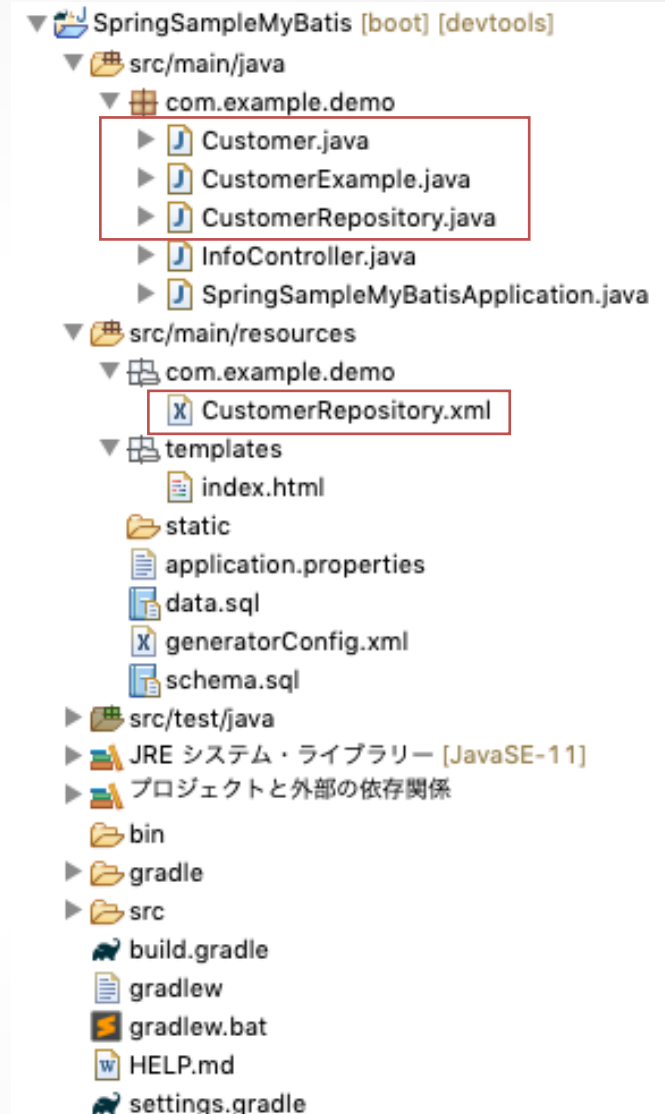
CustomerExample.java

Javaインターフェースファイル

CustomerRepository.java

XMLファイル

CustomerRepository.xml



## 2, サンプル

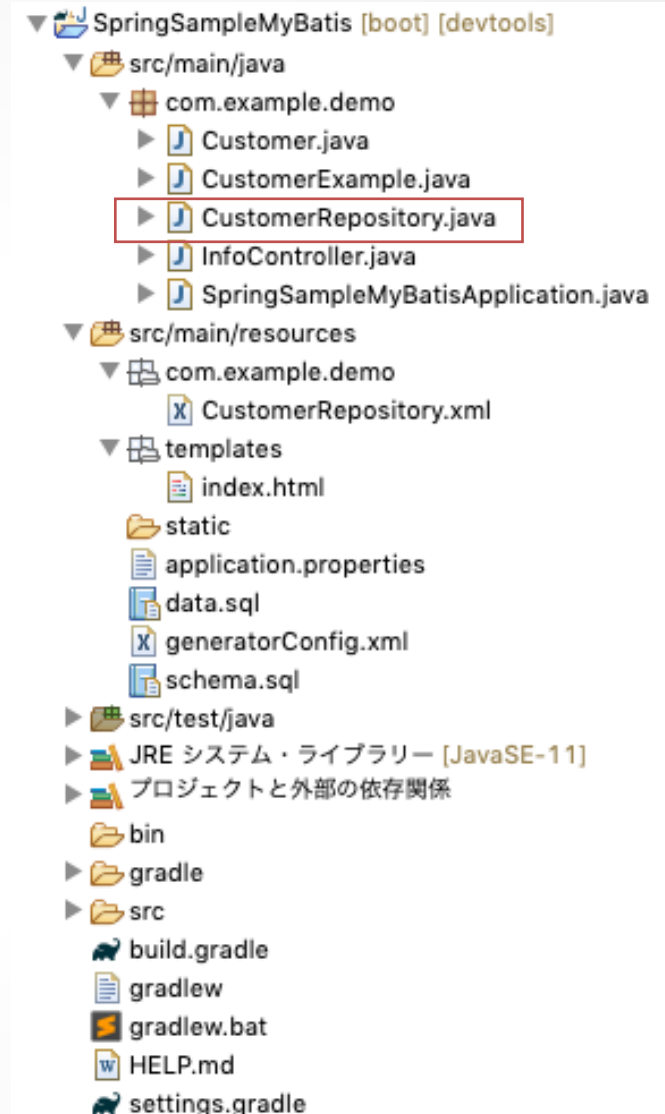
### 2-1, プロジェクトの作成

#### 7, @Mapperアノテーションの付加

CustomerRepository.javaに以下のように、  
@Mapperアノテーションを付加する。

**@Mapper**

```
public interface CustomerRepository {  
    ...  
}
```



## 2, サンプル

### 2-1, プロジェクトの作成

8, toString() メソッドのオーバーライド

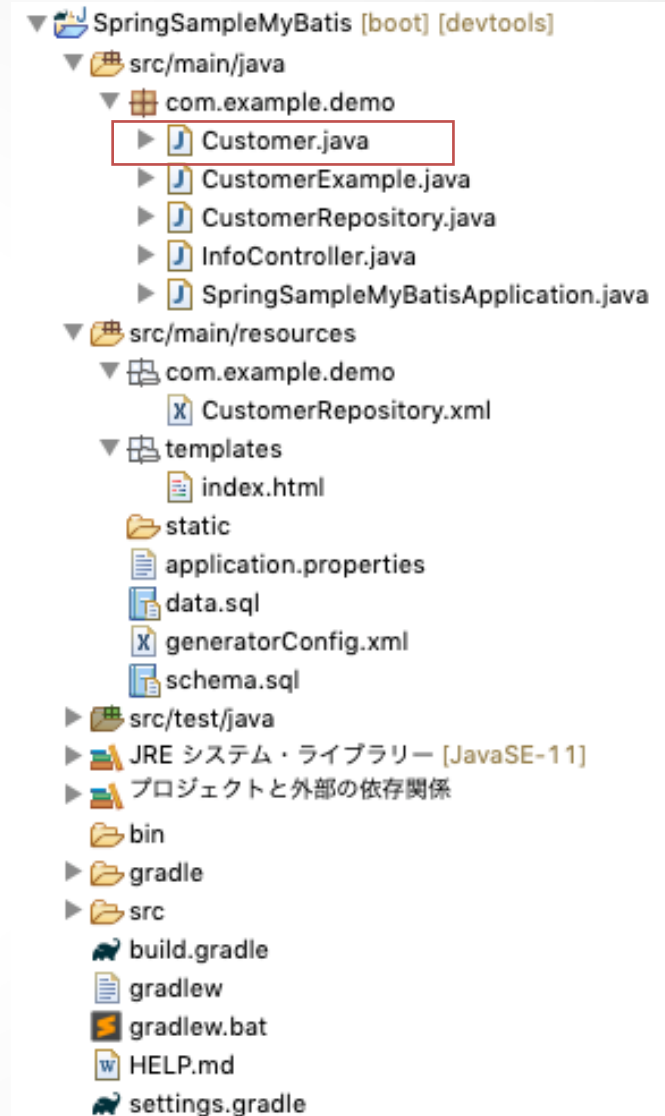
Customer.java を選択。

Eclipse のツールバー「ソース」

→ toString() の生成

以下のようなコードが自動生成される。

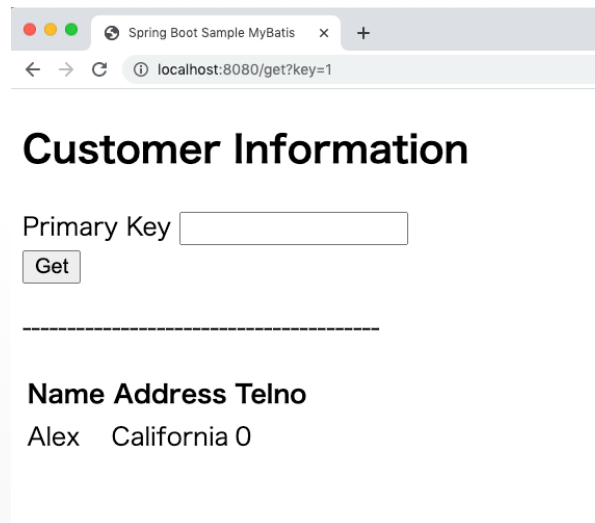
```
@Override  
public String toString() {  
    ...  
}
```



## 2, サンプル

### 2-2, プロジェクト実行

- 1, MySQLサーバーを起動する。
- 2, プロジェクトを実行する。
- 3, “<http://localhost:8080/>”へアクセスする。
- 4, Primary Key欄に、対象の数字を入れる。
- 5, Getボタンを押下。
- 6, 画面下部に、DBから取得してきた顧客情報が表示される。



Spring Boot Sample MyBatis x +

localhost:8080/get?key=1

### Customer Information

Primary Key

Get

---

Name	Address	Telno
Alex	California	0

# MyBatis in Spring Boot

Fin.