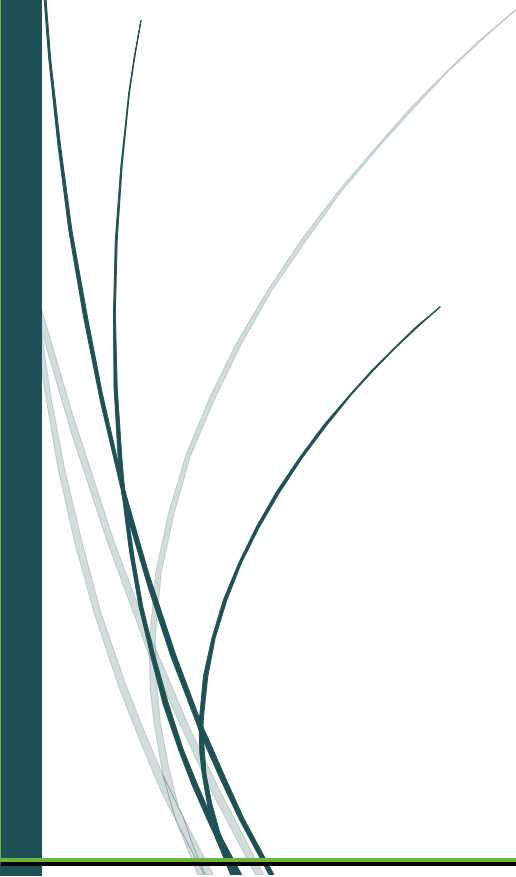


The Spring logo is a red arrow pointing to the right, with the word "Spring" written in white inside it.

Spring

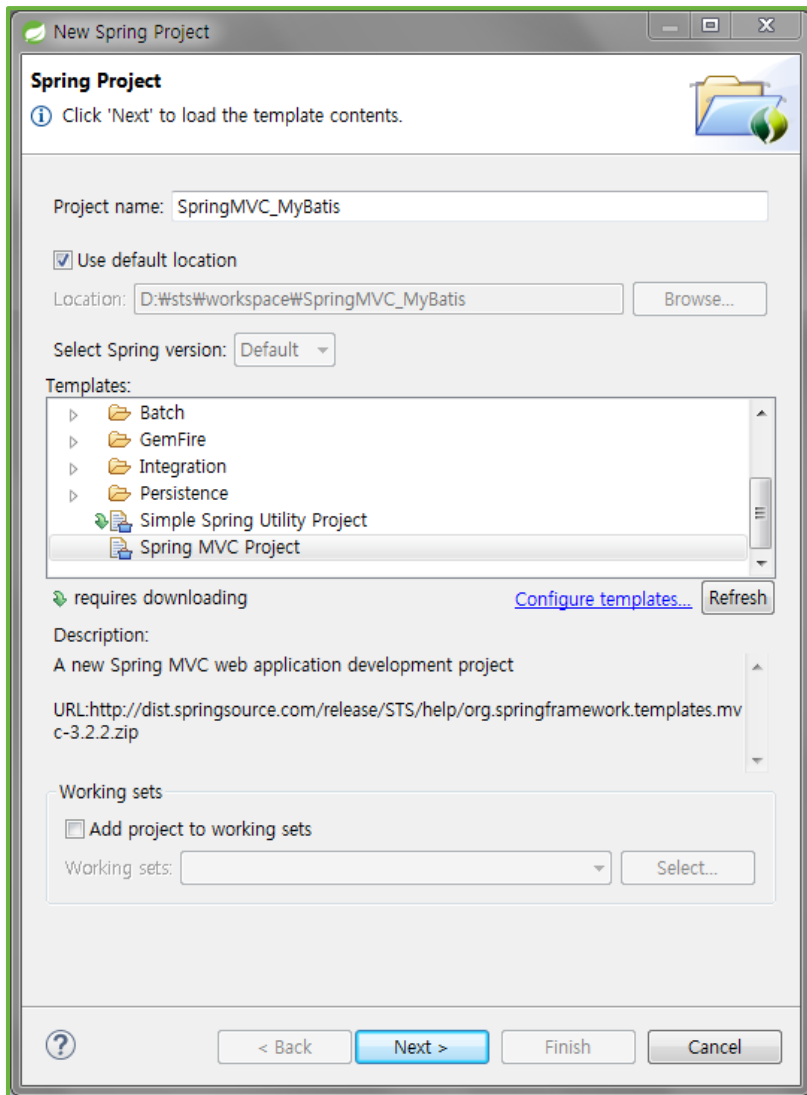
# MVC + MyBatis

Several thin, curved lines in shades of blue and grey originate from the left edge and curve upwards and to the right, creating a decorative element.

권기웅

# 1.Spring MVC Project 작성

## 1.1 Spring MVC Project 작성



## 2.pom.xml

### 2.1 jdbc6.jar 용 외부 저장소 등록

```
<repositories>
  <repository>
    <id>code.lds</id>
    <url>https://code.lds.org/nexus/content/groups/main-repo</url>
  </repository>
</repositories>
```

### 2.2 dependency 추가

```
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-orm</artifactId>
  <version>3.1.1.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-tx</artifactId>
  <version>3.1.1.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-jdbc</artifactId>
  <version>3.0.6.RELEASE</version>
</dependency>
```

```
<dependency>
  <groupId>commons-dbcp</groupId>
  <artifactId>commons-dbcp</artifactId>
  <version>1.2.2</version>
</dependency>
```

```
<dependency>
  <groupId>org.mybatis</groupId>
  <artifactId>mybatis</artifactId>
  <version>3.0.6</version>
</dependency>
<dependency>
  <groupId>org.mybatis</groupId>
  <artifactId>mybatis-spring</artifactId>
  <version>1.0.2</version>
</dependency>
<dependency>
  <groupId>com.oracle</groupId>
  <artifactId>ojdbc6</artifactId>
  <version>12.1.0.2</version>
</dependency>
```



## 3. SqlSession bean 등록&설정

### 3.1 src/main/webapp/WEB-INF/spring/root-context.xml 수정

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans xmlns="http://www.springframework.org/schema/mvc" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:beans="http://www.springframework.org/schema/beans" xmlns:context="http://www.springframework.org/schema/context"
  xmlns:tx="http://www.springframework.org/schema/tx" xmlns:jdbc="http://www.springframework.org/schema/jdbc"
  xmlns:jee="http://www.springframework.org/schema/jee" xmlns:aop="http://www.springframework.org/schema/aop"
  xsi:schemaLocation="http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc.xsd
    http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans.xsd
    http://www.springframework.org/schema/context http://www.springframework.org/schema/context.xsd
    http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx.xsd
    http://www.springframework.org/schema/jdbc http://www.springframework.org/schema/jdbc.xsd
    http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop.xsd
    http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee.xsd" />

  <annotation-driven />
  <context:annotation-config />
  <context:component-scan base-package="com.kwon.mvcmybatis" />
  <tx:annotation-driven />
  <beans:bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource">
    <property name="driverClassName" value="com.mysql.jdbc.Driver" />
    <property name="url" value="jdbc:mysql://localhost:3306/mybatis" />
    <property name="username" value="root" />
    <property name="password" value="1234" />
  </beans:bean>
  <beans:bean id="transactionManager" class="org.springframework.jdbc.datasource.DataSourceTransactionManager">
    <property name="dataSource" ref="dataSource" />
  </beans:bean>
  <beans:bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource" />
    <property name="mapperLocations" value="classpath*:com/kwon/mybatis/**/*.xml" />
  </beans:bean>
  <beans:bean id="sqlSession" class="org.mybatis.spring.SqlSessionTemplate">
    <property name="sqlSessionFactory" ref="sqlSessionFactory" />
  </beans:bean>
</beans:beans>
```



# 4. JavaBean

## 4.1 Product.java 작성

```
src/main/java
├── com.kwon.mvcmybatis
│   ├── HomeController.java
│   └── com.kwon.mvcmybatis.product
│       └── Product.java
```

```
public class Product {
    private String p_id;
    private String p_name;
    private BigDecimal p_price;

    public Product() {
        // TODO Auto-generated constructor stub
    }

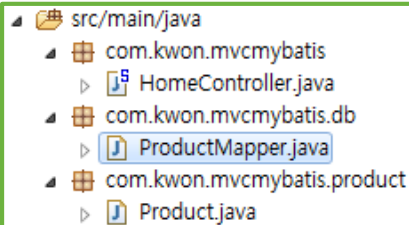
    public Product(String p_id, String p_name, BigDecimal p_price) {
        super();
        this.p_id = p_id;
        this.p_name = p_name;
        this.p_price = p_price;
    }

    public String getP_id() {return p_id;}
    public void setP_id(String p_id) {this.p_id = p_id;}
    public String getP_name() {return p_name;}
    public void setP_name(String p_name) {this.p_name = p_name;}
    public BigDecimal getP_price() {return p_price;}
    public void setP_price(BigDecimal p_price) {this.p_price = p_price;}
}
```



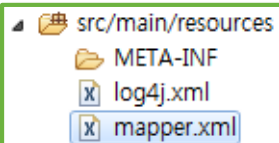
# 5.mapper

## 5.1 mapper 에 대응하는 Interface 작성



```
public interface ProductMapper {  
    public abstract List<Product> getProduct(Product p);  
}
```

## 5.2 src/main/resources/mapper.xml 파일 작성



```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"  
<mapper namespace="com.kwon.mvcmybatis.db.ProductMapper">  
    <select id="getProduct"  
        parameterType="com.kwon.mvcmybatis.product.Product"  
        resultType="com.kwon.mvcmybatis.product.Product">  
        select *  
        from jun11_product_kwon  
        where p_price <= #{p_price}  
        order by p_name  
    </select>  
</mapper>
```



## 6.DBDAO

### 6.1 DBDAO.java 작성

```
src/main/java
├── com.kwon.mvcmybatis
│   └── HomeController.java
├── com.kwon.mvcmybatis.db
│   ├── DBDAO.java
│   └── ProductMapper.java
└── com.kwon.mvcmybatis.product
    └── Product.java
```

```
@Service
public class DBDAO {

    @Autowired
    private SqlSession ss;

    public void getProduct(Model m, Product p){
        ProductMapper pm = ss.getMapper(ProductMapper.class);
        m.addAttribute("products", pm.getProduct(p));
    }
}
```



# 7.Controller

## 7.1 HomeController.java

```
@Controller
public class HomeController {

    @Autowired
    private DBDAO d;

    @RequestMapping(value = "/", method = RequestMethod.GET)
    public String home(Locale locale, Model model) {
        return "input";
    }

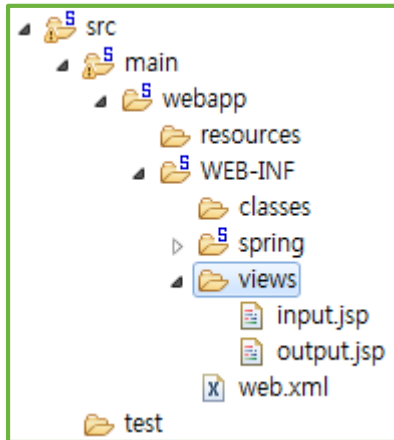
    @RequestMapping(value = "/getProduct.do", method = RequestMethod.GET)
    public String home(Model m,
        @RequestParam(value = "maxPrice") BigDecimal max) {
        Product p = new Product(null, null, max);
        d.getProduct(m, p);
        return "output";
    }
}
```





## 8. View

### 8.1 구조



### 8.2 input.jsp

```
<body>
  <form action="getProduct.do">
    최대가격 : <input name="maxPrice"><p>
      <input type="submit" value="확인">
  </form>
</body>
```



### 8.3 output.jsp

```
<%@ page language="java" contentType="text/html; charset=EUC-KR"
    pageEncoding="EUC-KR"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http:
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
<title>Insert title here</title>
</head>
<body>
    <table border="1">
        <tr>
            <th>품명</th>
            <th>가격</th>
        </tr>
        <c:forEach var="p" items="${products}">
            <tr>
                <td>${p.p_name }</td>
                <td>${p.p_price }</td>
            </tr>
        </c:forEach>
    </table>
</body>
</html>
```



## 9. 실행결과

