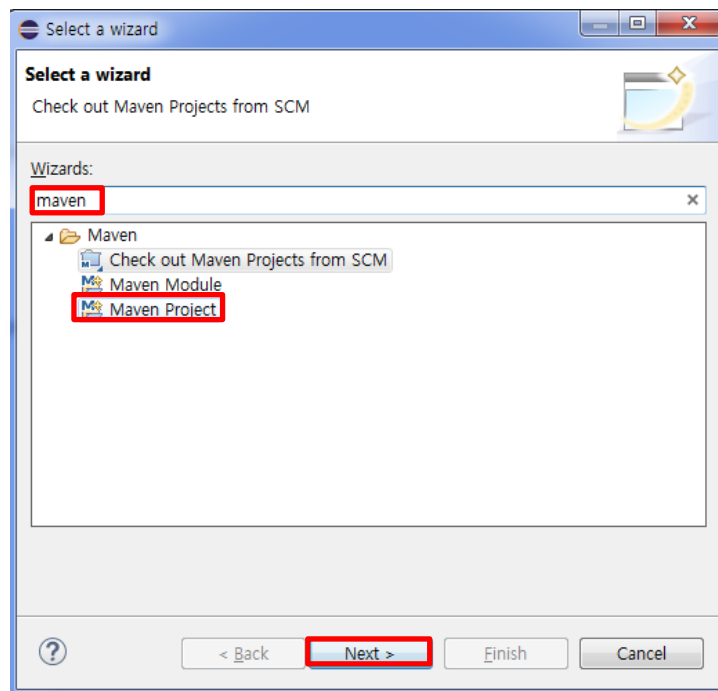
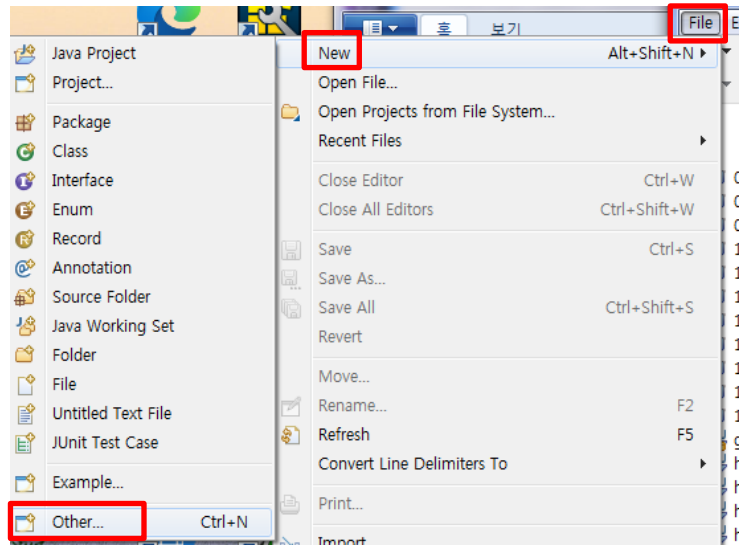
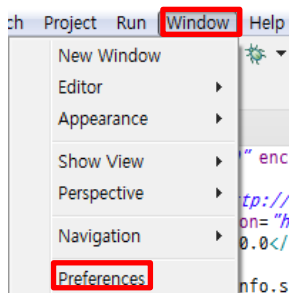
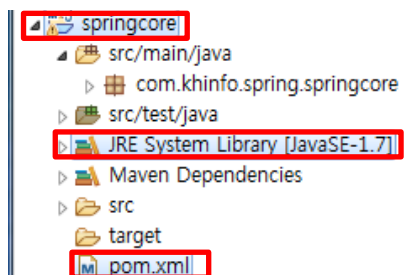
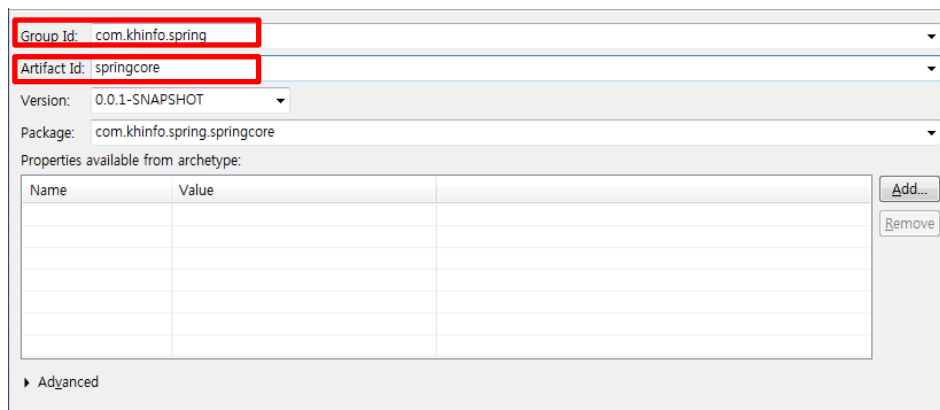
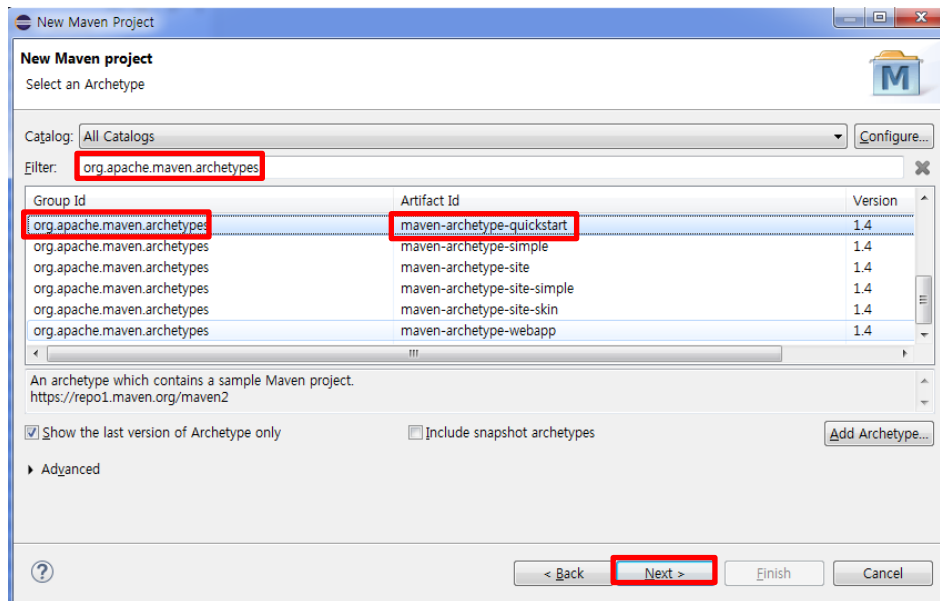
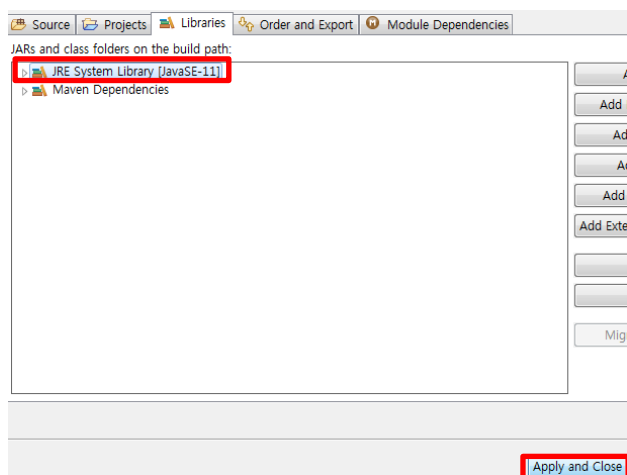
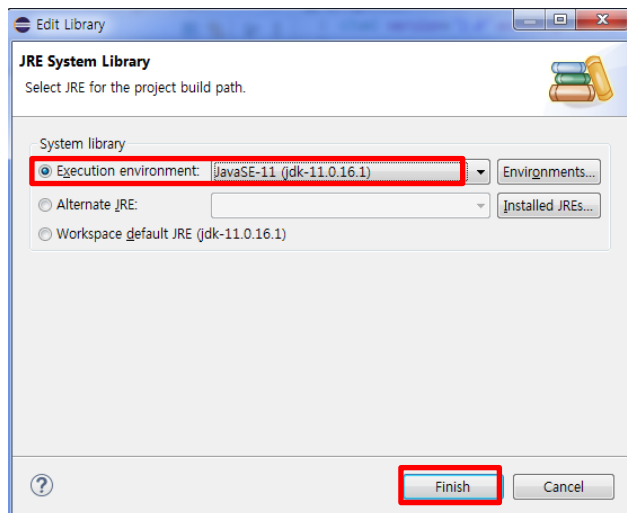
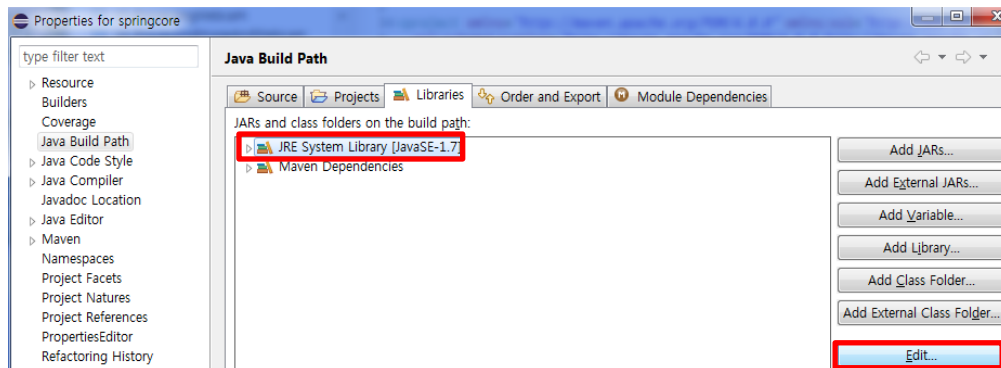
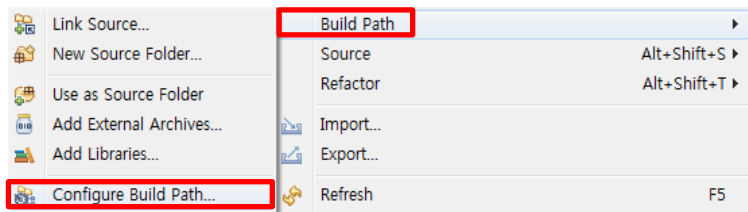
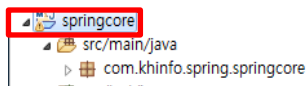
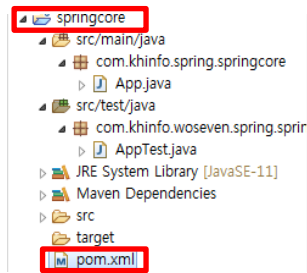


## 1. 메이븐 프로젝트 만들기









**pom.xml**

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
```

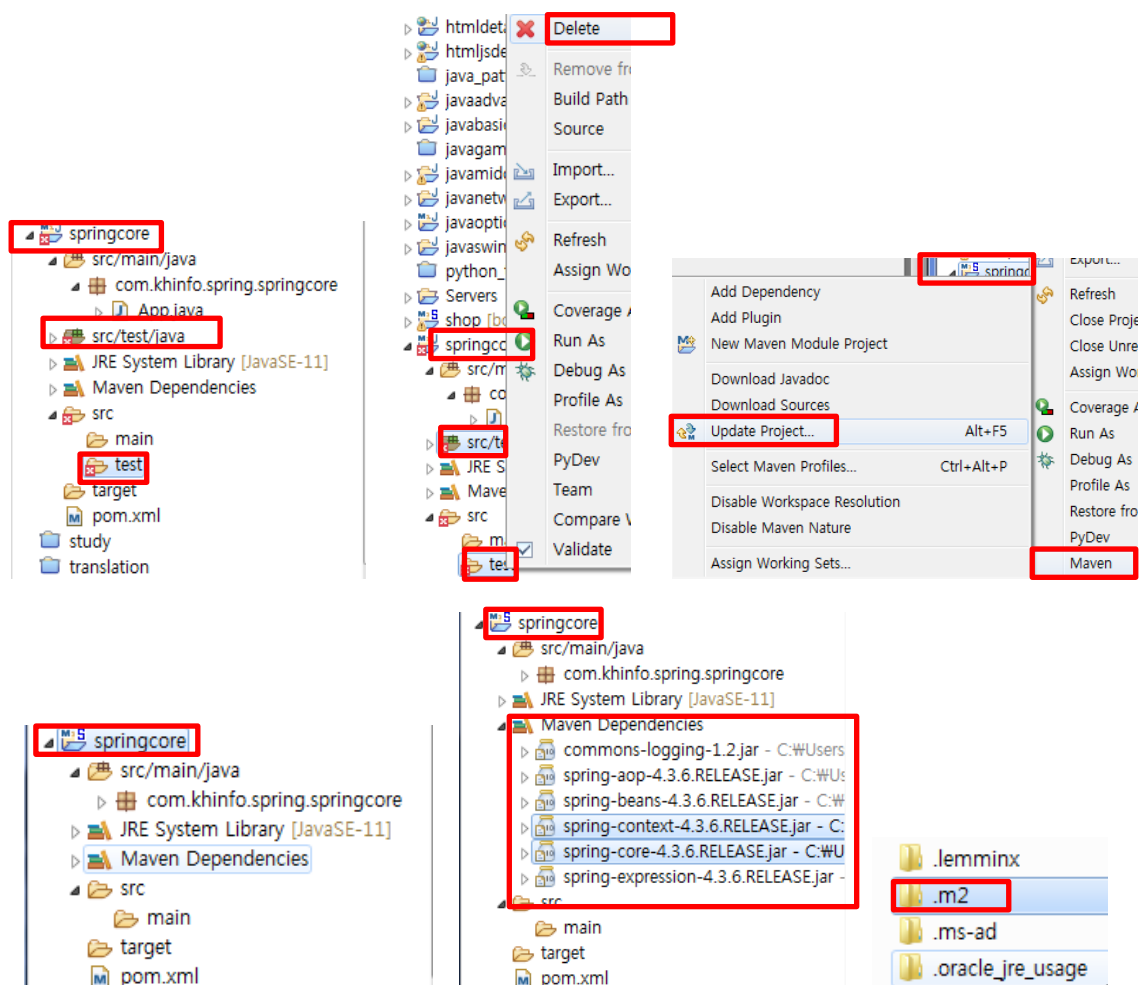
```
<name>springcore</name>
<url>http://maven.apache.org</url>
```

```
<dependencies>
  <dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-core</artifactId>
    <version>${springframework.version}</version>
  </dependency>
  <dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-context</artifactId>
    <version>${springframework.version}</version>
  </dependency>
```

```

        </configuration>
    </plugin>
</plugins>
</pluginManagement>
</build>
</project>

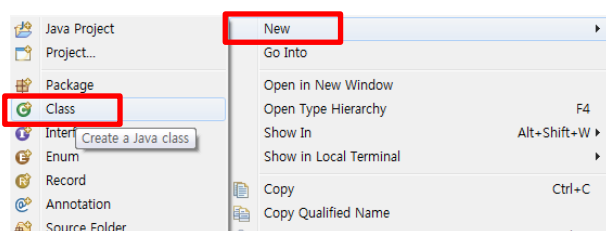
```



## 2. 자바빈 생성

### 2.1 DI에 대한 3단계

#### 가. POJO(Plain Old Java Object) 클래스 만들기



**Java Class**  
Create a new Java class.

Source folder:

Package:

☐ Enclosing type:

Name:

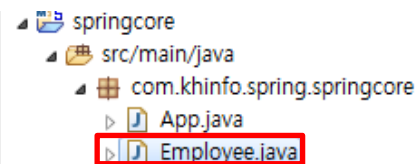
Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?  
☐ public static void main(String[] args)  
☐ Constructors from superclass  
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))  
☐ Generate comments



**Employee.java**

```

1 package com.khinfo.spring.springcore;
2
3 public class Employee {
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

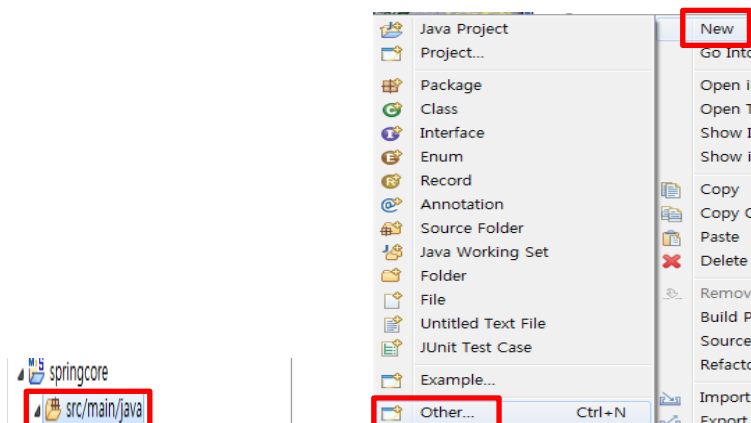
**Employee.java**

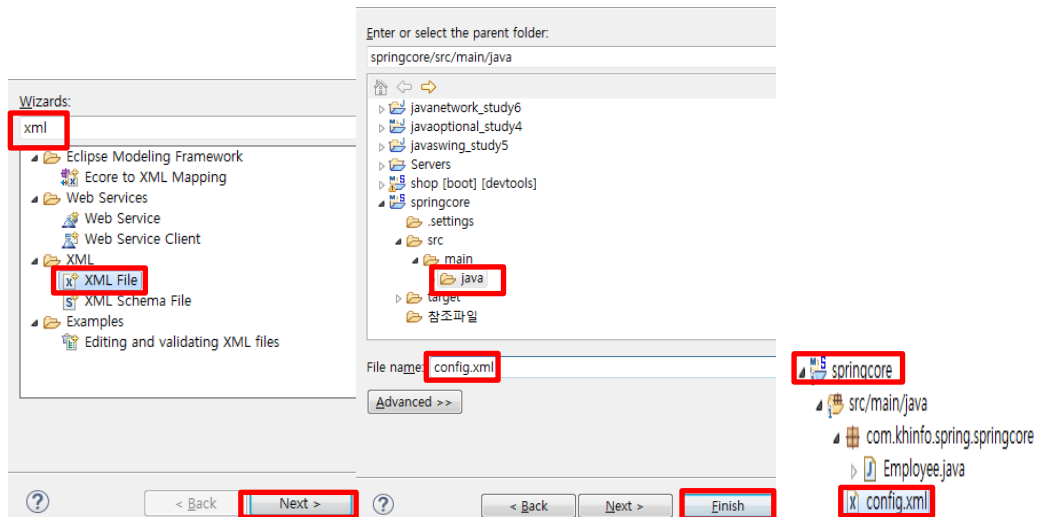
```

1 package com.khinfo.spring.springcore;
2
3 public class Employee {
4
5     private int id;
6     private String name;
7
8     public int getId() {
9         return id;
10    }
11
12    public void setId(int id) {
13        this.id = id;
14    }
15
16    public String getName() {
17        return name;
18    }
19
20    public void setName(String name) {
21        this.name = name;
22    }
23
24 }

```

## 나. 설정파일 만들기





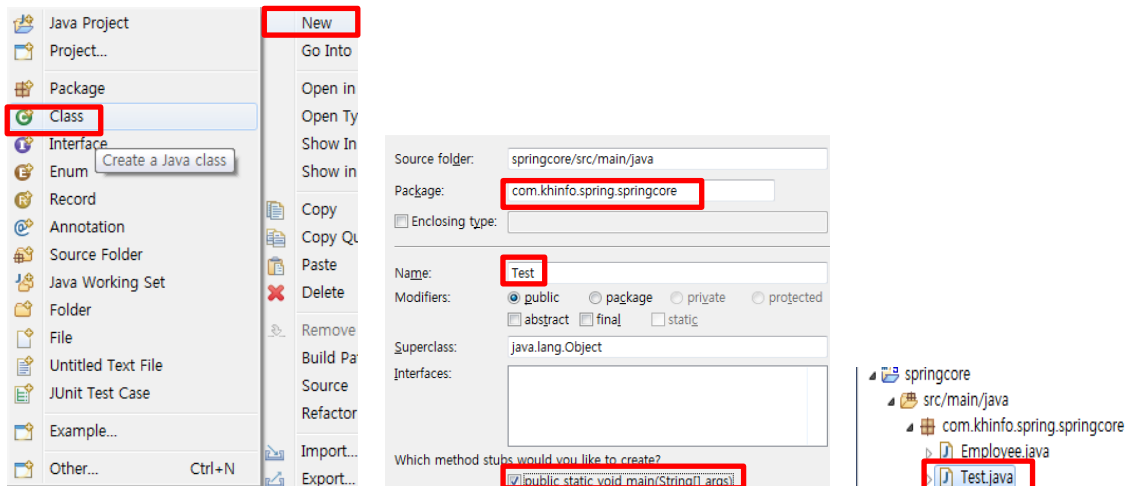
```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:p="http://www.springframework.org/schema/p"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context.xsd">
```

```
<bean name="emp" class="com.khinfo.spring.springcore.Employee">
    <property name="id">
        <value>20</value>
    </property>
    <property name="name">
        <value>KHT</value>
    </property>
</bean>
```

```
</beans>
```

다. 테스트 클래스 만들기



### 3. 컬렉션 주입

#### 가. 컬렉션 타입

- List

```
<bean>
  <property name="productIds">
    <list>
      <value>10 </value>
      <value> 20 </value>
    </list>
  </property>
</bean>
```

- Set: 중복을 허용하지 않음

```
<bean>
  <property name="productIds">
    <set>
      <value>10 </value>
      <value> 20 </value>
    </set>
  </property>
</bean>
```

- Map:

```
<bean>
  <property name="productsInOrder">
    <map>
      <entry key="100" value="iPhone" />
    </map>
  </property>
</bean>
```

- Properties: Hashtable의 하위 클래스, 키와 값을 문자열로 제한



```
<bean>
  <property name="languages">
    <props>
      <prop key="USA">English</prop>
```

#### 4. 리스트

##### 가. 개발 요약

Hospital	Spring Bean
String name	Configuration File
List<String> departments	Test Class

##### 나. 스프링 빈 생성

##### 다. 설정 파일 생성

##### 라. 테스트 클래스 생성

##### 마. 실행

#### 5. Set

##### 가. 개발 요약

CarDealer	Spring Bean
String name	Configuration File
List<String> models	Test Class

##### 나. 스프링 빈 생성

##### 다. 설정 파일 생성

##### 라. 테스트 클래스 생성

##### 마. 실행

#### 6. Map

```
Customer
    int id;
    Map<Integer,String> products;
```

## 7. Properties

```
Languages
    Properties countryAndLangs;
```

## 8. 레퍼런스 타입

```

A      HAS - A      B
B b;

<bean class="B" name="b"/>

    <bean class="A" name="a">
        <property name="b">
            <ref bean="b"/>
        </property>
    </bean>
```

```

Student ----- Scores
                Double maths;
                Double physics;
                Double chemistry;
```

## 9. 레퍼런스 타입: 속성과 P 스키마

## 10. 세터주입 정리

```
Three Steps:

    Creates the Spring Beans
    Create the Spring Configuration File
    Use the Spring Bean from the container
```

Primitives	Collections	Objects
value element	List	<ref>
value attribute	Set	<bean ref="">
P Schema	Map	
	Properties	

## 11. 과제

