제4장 스프링프레임워크 개발 환경 구축

스프링 개발환경 구축

- ▶ Java 설치 (https://www.oracle.com/index.html)
 - ▶ JDK 1.8(안전) 이상 설치
- ▶ STS 설치 (https://spring.io/tools)
 - Eclipse + Spring Plugin
- ▶ Tomcat 설치
 - ▶ Tomcat 8.x

Java 설치 (https://www.oracle.com/index.html)

- https://www.oracle.com/java/technologies/downloads/
- ▶ JDK 1.8 이상 설치
- ▶ 오라클 회원가입 (계정 만들기)
- ▶ JAVA SE버전 설치(무료임) JavaSE 8u212 무료
- ▶ 본인의 OS에 맞는 JDK를 설치

- Eclipse + Spring Plugin
- ▶ IDE툴이란 통합 개발환경으로 코딩, 디버그, 배포 등 프로그램 개발에 관련된 모든 작업을 하나의 프로그램 안에서 처리하는 환경
- ▶ 스프링을 개발하기 위해서는 개발에 필요한 Spring plugin 설치
- ▶ Eclipse이외의 tool에서도 가능함
- ▶ 본인의 OS에 맞는 IDE 버전을 설치

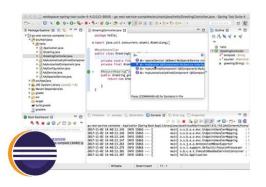


Spring Tools 4 for Eclipse

The all-new Spring Tool Suite 4. Free. Open source.

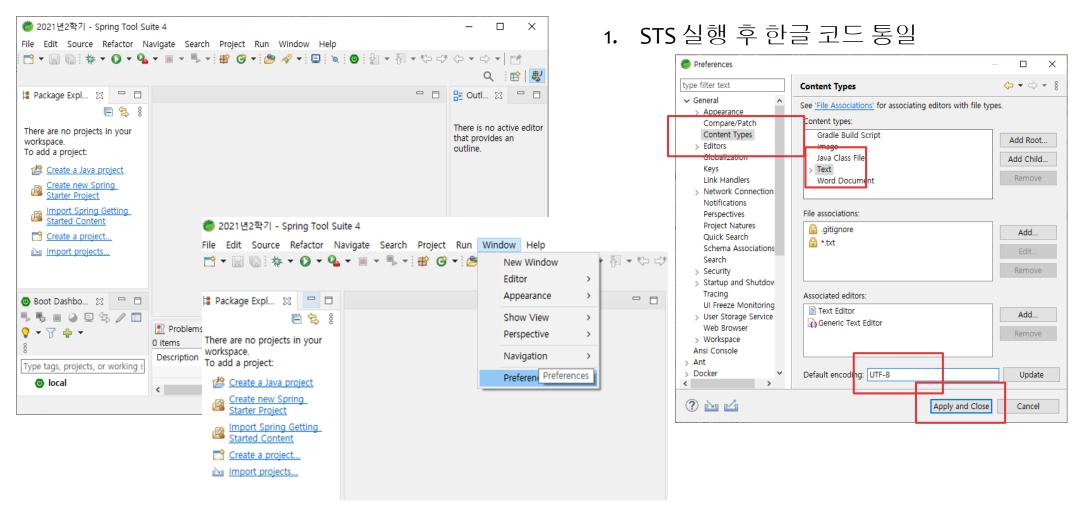
4.11.1 - LINUX X86_64 4.11.1 - MACOS X86_64

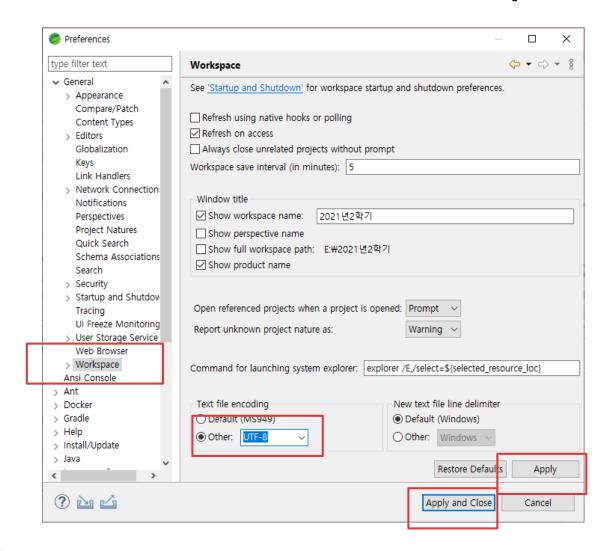
4.11.1 - WINDOWS X86_64



- 1. 사이트에 접속
- 2. 본인의 OS에 맞는 스프링 다운로드
- 3. 압축해제
- 4. SpringToolSuite4.exe 응용프로그램 실행







2. 한글 코드 통일

Tomcat 설치 (http://tomcat.apache.org/)

- ▶ Tomcat 8.x (안전) 이상
- ▶ 웹 서버와 연동하여 실행할 수 있는 자바 환경 제공
- ▶ 스프링을 이용한 웹 환경 구성시 사용

Tomcat 설치 (http://tomcat.apache.org/)





Apache Tomcat®





Save the date!

Apache Tomcat

Home Taglibs Maven Plugin

Download

Which version? Tomcat 10 Tomcat 9 Tomcat Migration Tool for Jakarta EE Tomcat Connectors Tomcat Native Taglibs Archives

Documentation

Tomcat 10.1 (alpha)

Tomcat 9.0

Apache Tomcat

The Apache Tomcat[®] software is an open source implementation of the <u>Jakarta Servlet</u>, <u>Jakarta Server Pages</u>, <u>Jakarta Expression</u> Language, Jakarta WebSocket, Jakarta Annotations and Jakarta Authentication specifications. These specifications are part of the

The Jakarta EE platform is the evolution of the Java EE platform. Tomcat 10 and Jater implement specifications developed as part of laborts EE Tomcat 0 and paying implements an

The Apache Tomcat software is develope version 2. The Apache Tomcat project is We invite you to participate in this open

of Jakarta EE. Tomcat 9 and earlier imple

Apache Tomcat software powers numer organizations. Some of these users and

Apache Tomcat, Tomcat, Apache, the Ap-Software Foundation.

Tomcat 8.5.71 Released

The Apache Tomcat Project is proud to a specifications that are part of the Java EI

- Update the internal fork of Apache FileUpload to 2.0, and Apache Con
- Update the packaged version of the Problems? · Correct parsing of Content-Range

Archives

Documentation

Tomcat 10.1 (alpha) Tomcat 10.0 Tomcat 9.0

Tomcat 8.5 Tomcat Connectors Tomcat Native

Migration Guide

Presentations Specifications

Security Reports Find help Mailing Lists **Bug Database**

Get Involved

Overview Source code Buildbot Translations Tools

Media

Twitter YouTube Blog

You are currently using https://mirror.navercorp.com/apache/. If you encour another mirror. If all mirrors are failing, there are backup mirrors (at the end of

Other mirrors: https://mirror.navercorp.com/apache/ V Change

9.0.53

Please see the **README** file for packaging information. It explains what every dis

Binary Distributions

- Core:
 - zip (pgp, sha512)
 - o tar.gz (pgp, sha512)
 - o 32-bit Windows zip (pgp, sha512)
 - 64-bit Windows zip (pgp, sha512)
 - o 32-bit/64-bit Windows Service Installer (pgp, sha512)

 - tar.gz (pgp, sha512)
- Deployer:
 - zip (pgp, sha512)
 - tar.gz (pgp, sha512)
- Embedded:
 - tar.gz (pgp, sha512)
 - o zip (pgp, sha512)

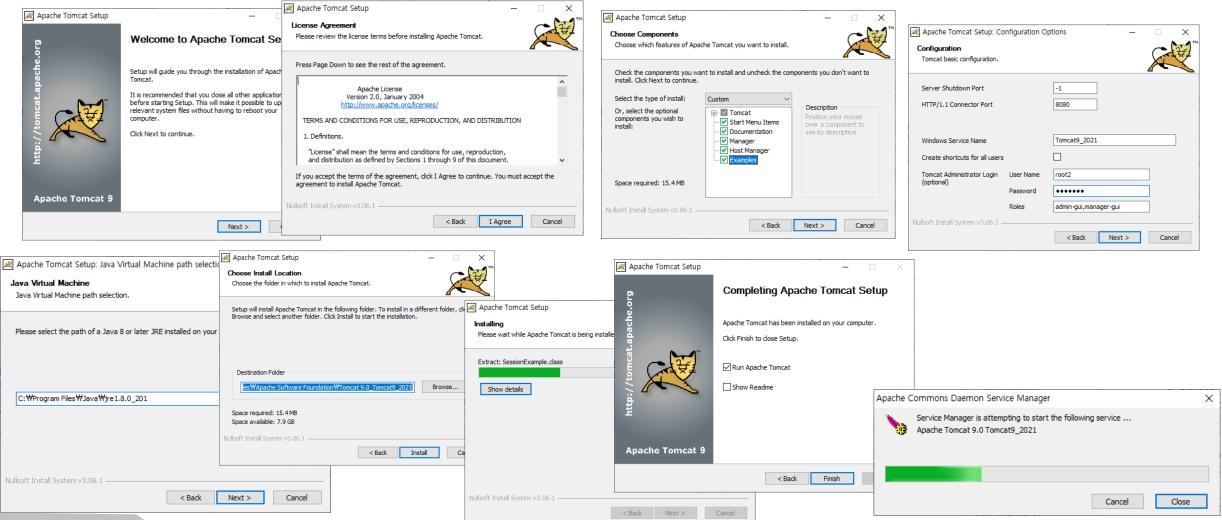
Source Code Distributions

- tar.gz (pgp, sha512)
- zip (pgp, sha512)



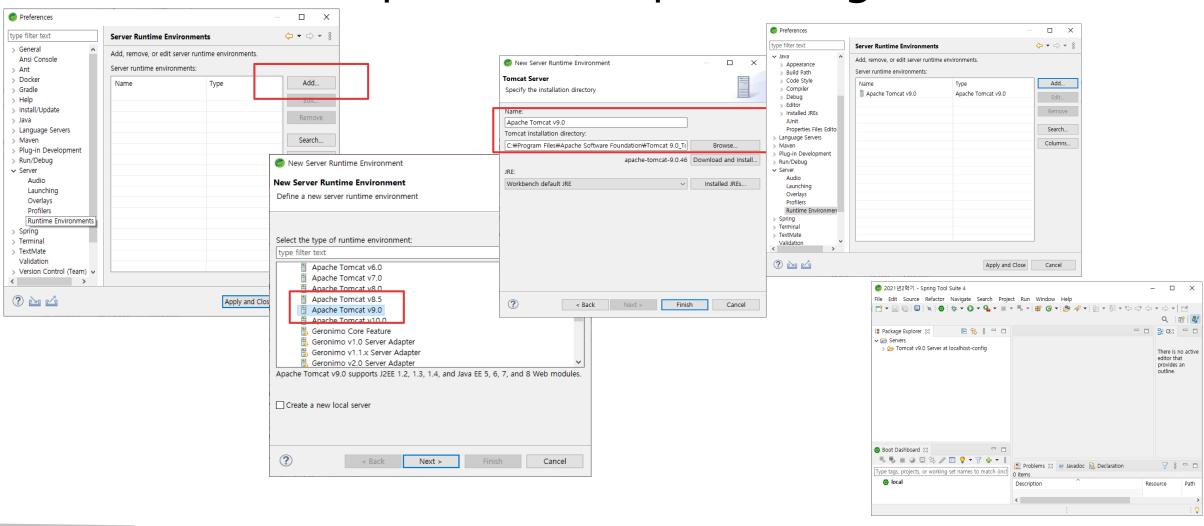
Tomcat 설치





Tomcat 설치 (http://tomcat.apache.org/)

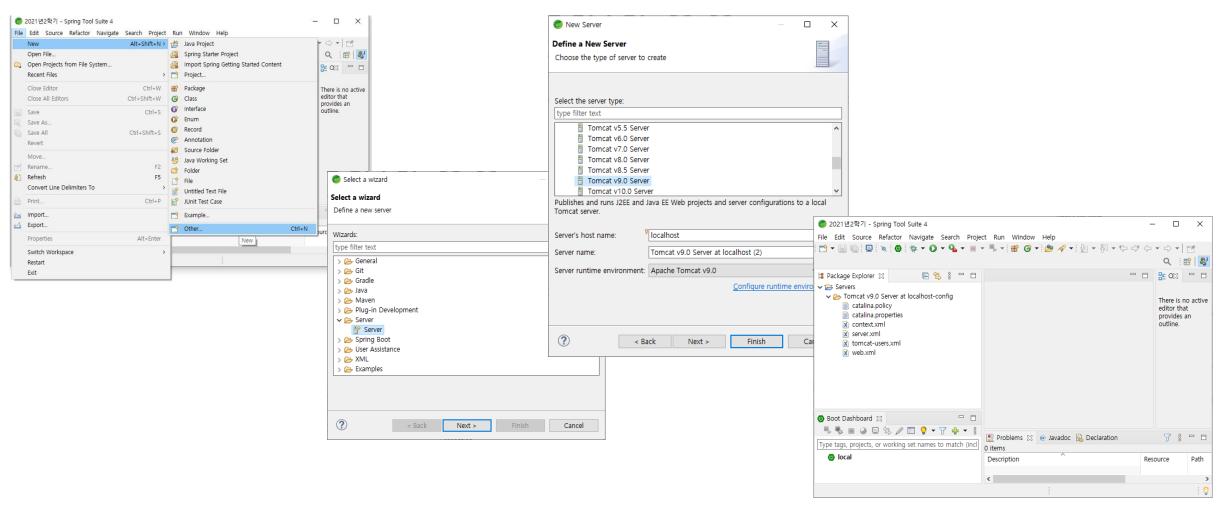






Server 생성

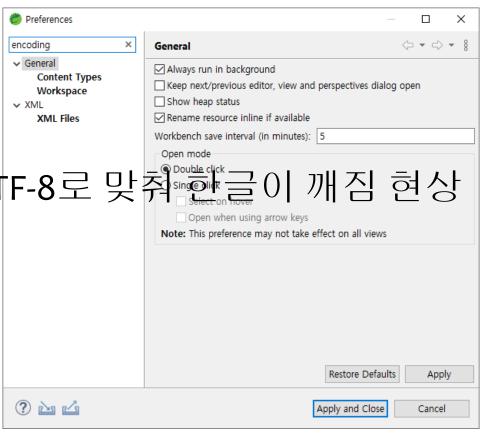




프로젝트 생성



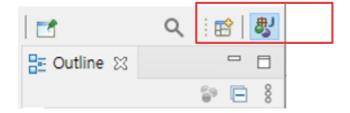
- ▶ STS 기본 설정
 - encoding: UTF-8
 - STS에서 사용되는 모든 문자셋을 UTF-8로 맞춰 Sing 이 깨짐 현상 방지
- ▶ Java 설정
- ▶ 스프링 프로젝트 생성



프로젝트 생성

000

- ▶ STS 기본 설정
 - encoding: UTF-8
 - ▶ STS에서 사용되는 모든 문자셋을 UTF-8로 맞추고 작업을 해야 추 후에 한글이 깨지는 현상을 방지할 수 있음
- ▶ Java 설정



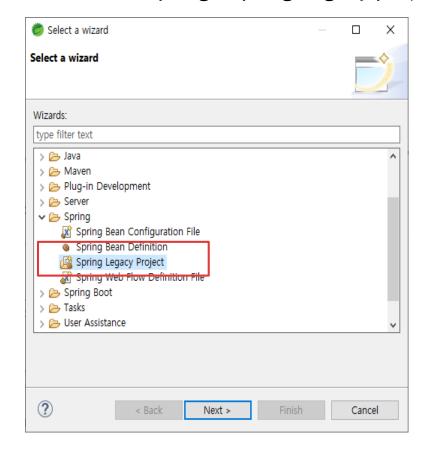
▶ 스프링 프로젝트 생성

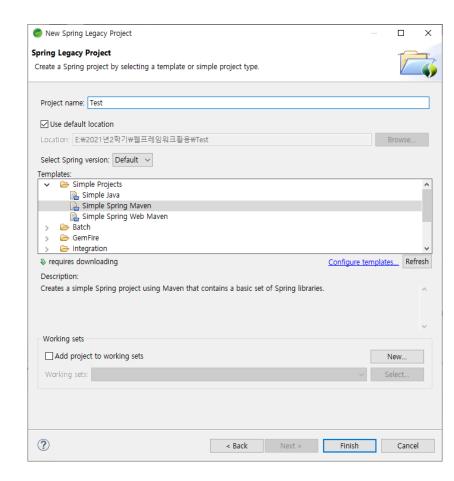


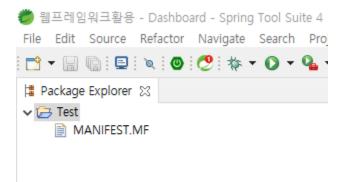
스프링 프로젝트 생성



File->other->spring->Spring Legacy project



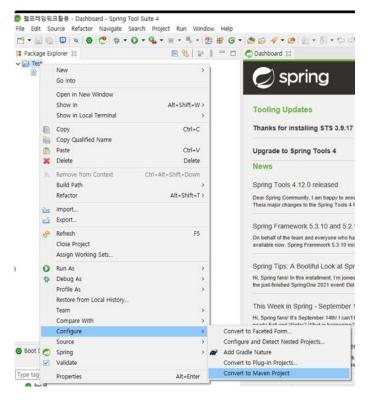


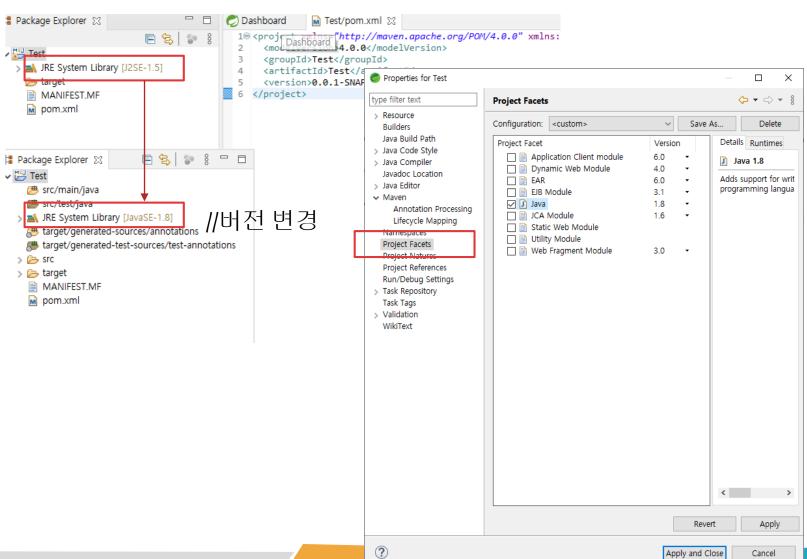




스프링 프로젝트 생성







메이븐라이브러리

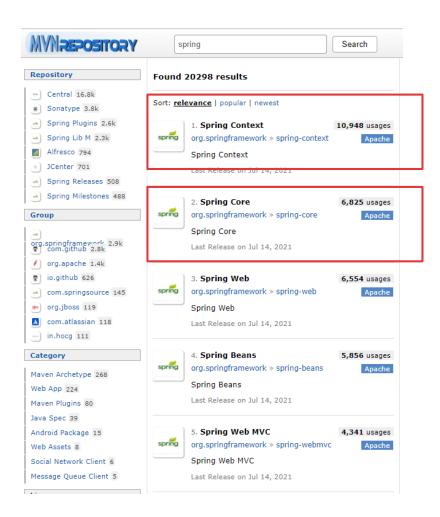


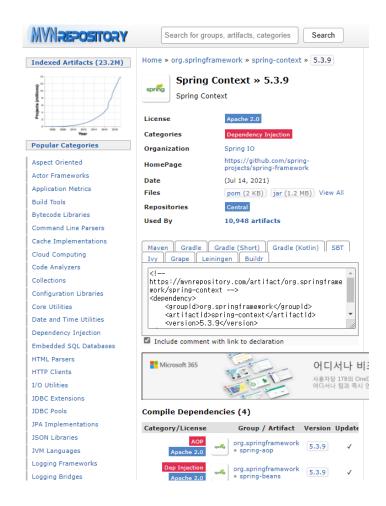
- Apache Maven
 - ▶ 자바 프로젝트 관리 도구 (자바 기본 버전1.5로 되어 있음)
 - ▶ 빌드, 문서화, 리포팅, 의존관계 관리, 소스코드 관리, 릴리즈, 배포
 - 빌드시스템을제공
 - ▶ pom.xml에 빌드에 관련된 모든 정보 기술
 - ▶ 라이브러리 의존성 관리
 - https://mvnrepository.com
 - https://maven.apache.org



https://mvnrepository.com





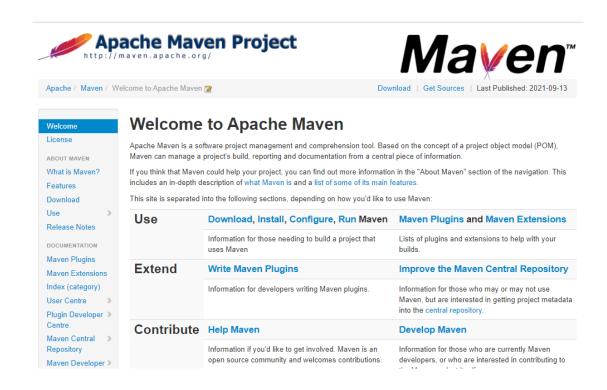


*다양한 라이브러리 제공



https://maven.apache.org





```
ct>
<build>
[...]
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-compiler-plugin</artifactId>
<version>3.8.1</version>
<configuration>
   <source>1.8</source>
   <target>1.8</target>
</configuration>
</plugin>
</plugins>
| ... |
</build>
| ... |
</project>
```

http://Apache Maven Compiler Plugin – Setting the -source and -target of the Java Compiler

pom.xml 파일의 구성(1)

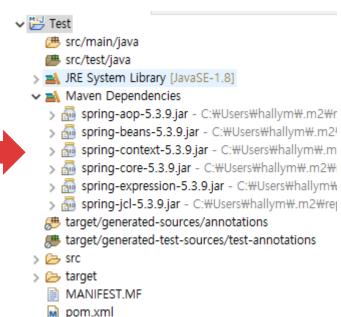
```
000
```

```
Dashboard
        ★Test/pom.xml 
 10 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http:/
     <modelVersion>4.0.0</modelVersion>
     <groupId>Test
     <artifactId>Test</artifactId>
     <version>0.0.1-SNAPSHOT</version>
     <dependencies>
       <dependency>
       </dependency>
     </dependencies>
   </project>
```

pom.xml 파일의 구성(2)



```
Dashboard
      <modelVersion>4.0.0</modelVersion>
    <groupId>Test
    <artifactId>Test</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <dependencies>
 8
 9
      <!-- https://mvnrepository.com/artifact/org.springframework/spring-context -->
      <dependency>
109
11
         <groupId>org.springframework
         <artifactId>spring-context</artifactId>
12
13
         <version>5.3.9
14
      </dependency>
      <!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->
15
16⊝
      <dependency>
         <groupId>org.springframework
17
         <artifactId>spring-core</artifactId>
18
19
         <version>5.3.9</version>
20
      </dependency>
21
    </dependencies>
22
23
24 </project>
```





pom.xml 파일의 구성(3)

```
000
```

```
project xmlns="http://maven.apache.org/POM/4.o.o"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.o.o https://maven.apache.org/xsd/maven-4.o.o.xsd">
<modelVersion>4.0.0</modelVersion>
<groupId>Test/groupId>
<artifactId>Test</artifactId>
<version>o.o.1-SNAPSHOT</version>
<dependencies>
<!-- https://mvnrepository.com/artifact/org.springframework/spring-context -->
<dependency>
                    <groupId>org.springframework
                    <artifactId>spring-context</artifactId>
                    <version>5.3.9</version>
</dependency>
<!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->
<dependency>
                    <groupId>org.springframework</groupId>
                    <artifactId>spring-core</artifactId>
                    <version>5.3.9</version>
</dependency>
</dependencies>
<bul><build>
        <plugins>
                    <plugin>
                        <groupId>org.apache.maven.plugins</groupId>
                        <artifactId>maven-compiler-plugin</artifactId>
                        <version>3.8.1</version>
                        <configuration>
                                        <source>1.8</source>
                                        <target>1.8</target>
                        </configuration>
                    </plugin>
         </plugins>
</build>
</project>
```



스프링 IoC의 구현(1)



```
1 package p01;
  public class HelloWorld {
       private String message;
       public String getMessage() {
            return message;
 9
10⊝
       public void setMessage(String message) {
11
            this.message = message;
12
13
14
                                          1 package p01;
15 }
                                            public class Main{
                                                 public static void main(String[] args) {
                                          6
                                                     HelloWorld hi = new HelloWorld();
                                                     hi.setMessage("안녕");
            🥋 Problems @ Javadoc 📵 Declaratio
                                          9
.0
.1 }
                                                     System.out.println(hi.getMessage());
            <terminated> Main [Java Application] C:\
            안녕
```



스프링 IoC의 구현(2)

```
000
```

```
1 <?xml version="1.0" encoding="UTF-8"?>
                                20 <beans xmlns="http://www.springframework.org/schema/beans"

√ Æ p01

                                      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     > II HelloWorld.java
                                      xsi:schemaLocation="http://www.springframework.org/schema/beans
     > 🞵 Main.java
                                         http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
                                 6
  > Æ p02
                                      <bean id="hi" class="p01.HelloWorld">
       beans.xml
                                8
                                          cproperty name="message" value="안녕하세요~" />
                                      </bean>
                                10
                                11 </beans>
```

```
package p01;

public class HelloWorld {
    private String message;

public String getMessage() {
    return message;
}

public void setMessage(String message) {
    this.message = message;
}

public void setMessage(String message) {
    this.message = message;
}
```

```
1 package p01;
3 import org.springframework.context.ApplicationContext;
4 import org.springframework.context.support.ClassPathXmlApplicationContext;
6 public class Main{
      public static void main(String[] args) {
          HelloWorld hi = new HelloWorld();
9
                                                                   🥋 Problems @ Javadoc 📵 Declara
                                                                  <terminated> Main [Java Application] C
          hi.setMessage("안녕");
          System.out.println(hi.getMessage());
                                                                   안녕하세요~
      */
4
5
6⊜
      public static void main(String[] args) {
          ApplicationContext con = new ClassPathXmlApplicationContext("beans.xml");
7
8
          HelloWorld h = (HelloWorld)con.getBean("hi");
          System.out.println(h.getMessage());
1 }
```

11 }

스프링 DI의 구현(1)

```
000
```

```
1 package p03;
                                   1 package p03;
                                                                                   3 public class EnPrint implements Print{

▼ 冊 p03
                                   2
                                                                                         private Name name;
                                   3 public class Name {
     In EnPrint.java
                                          private String name;
                                                                                         public EnPrint() {
     KrPrint.java
                                                                                                this.name = new Name();
                                         public Name() {
    Main.java
                                              this.setName("홍길동");
     Name.java
                                                                                  10⊝
                                                                                         @Override
     Print.java
                                   9
                                                                                 11
                                                                                         public void show() {
                                                                                 12
                                                                                            System.out.println(this.getClass().getSimpleName()+ ":"+name.getName());
                                  109
                                          public Name(String name) {
     beans.xml
                                                                                 13
                                  11
                                              this.name = name;
                                                                                 14
                                  12
                                                                                 15 }
                                  13
                                                                                                      1 package p03;
                                 14⊝
                                          public String getName() {
                                 15
                                              return name;
                                                                                                      3 public class KrPrint implements Print{
                                 16
                                                                                                           private Name name;
                                  17
                                          public void setName(String name) {
                                                                                                           public KrPrint() {
                                  18⊜
                                                                                                                  this.name = new Name();
                                  19
                                              this.name = name;
                                  20
                                                                                                      9⊝
                                                                                                           public KrPrint(Name name) { //생성자 파라미터
                                                                                                     10
                                                                                                               this.name = name;
1 package p03;
                                                                                                    11
                                                                                                    12
                                                                                                           public void setName(Name name) { // 메소드 파라미터
   public interface Print {
                                                                                                    14
                                                                                                               this.name = name;
        void show();
                                                                                                    15
5 }
                                     1 package p03;
                                                                                                    16
                                                                                                    17⊝
                                                                                                           @Override
                                                                                                    18
                                                                                                           public void show() {
                                       public class Main {
                                                                                                    19
                                                                                                               System.out.println(this.getClass().getSimpleName()+ ":"+name.getName());
                                            public static void main(String[] args) {
                                                                                                    20
                                     5
                                                 EnPrint ep = new EnPrint();
                                                                                                     21 }
                                                 ep.show();
                                     8
                                                 KrPrint kp = new KrPrint();
                                                                                             Problems @ Javadoc
                                                 kp.show();
                                                                                             <terminated> Main (2) [Java A
                                    10
                                                                                             EnPrint:홍길동
```

KrPrint:홍길동



스프링 DI의 구현(2)



```
> I EnPrint.java
> I KrPrint.java
> I Main.java
> I Name.java
> I Print.java
```

```
1 package p03;
 3 public class Main {
       public static void main(String[] args) {
 4⊖
            EnPrint ep = new EnPrint(); //기본형
 6
           ep.show();
           KrPrint kp = new KrPrint(); //기본형(변경 가능한 객체)
 8
 9
            kp.show();
10
11
           Name np = new Name("이만두"); //생성자 주입
12
            kp.setName(np);
           kp.show();
13
14
15
           np.setName("최말자"); //세터 주입
                                                      🧌 Problems @ Javadoc 😉 Declar
16
           kp.setName(np);
                                                      <terminated> Main (2) [Java Application
17
           kp.show();
                                                      EnPrint:홍길동
18
                                                      KrPrint:홍길동
19
                                                      KrPrint:이만두
20 }
                                                      KrPrint: 최말자
```



스프링 DI의 구현(3)



```
1 package p03;
 3 public class Main {
       public static void main(String[] args) {
           EnPrint ep = new EnPrint(); //기본형
 6
           ep.show();
           KrPrint kp = new KrPrint(); //기본형(변경 가능한 객체)
 9
           kp.show();
10
11
           Name np = new Name("이만두"); //생성자 주입
                                                             DI 디자인 패턴
12
           kp.setName(np);
13
           kp.show();
14
15
           np.setName("최말자"); //세터 주입
16
           kp.setName(np);
                                                           🥋 Problems @ Javadoc 😉 Declar
17
           kp.show();
                                                           <terminated> Main (2) [Java Application
18
                                                           EnPrint:홍길동
19
                                                           KrPrint:홍길동
20 }
                                                           KrPrint:이만두
                                                           KrPrint: 최말자
```



스프링 DI의 구현(4)



```
p03
DenPrint.java
```

```
1 <?xml version="1.0" encoding="UTF-8"?>
 20 <beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
          http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
       <bean id="hi" class="p01.HelloWorld">
           cproperty name="message" value="안녕하세요~" />
 9
       </bean>
10
11⊜
       <bean id="name" class="p03.Name">
           cproperty name="name" value="스프링 세터" />
12
       </bean>
13
14
       <bean id="kp" class="p03.KrPrint">
15⊜
           <constructor-arg ref="name" />
16
17
       </bean>
18
19 </beans>
                                                   Problems @ Javadoc 🗟 Declaration 🥃
```

```
에 Problems @ Javadoc 의 Declaration 의 Application 의 C:\(\mathcal{P}\mathcal{P}\)

*terminated> Main (2) [Java Application] C:\(\mathcal{P}\mathcal{P}\)

EnPrint: 홍길동

KrPrint: 이만두

KrPrint: 최말자

KrPrint: 스프링 세터
```

27 }

```
1 package p03;
 3 import org.springframework.context.ApplicationContext;
 4 import org.springframework.context.support.ClassPathXmlApplicationContext;
  public class Main {
      public static void main(String[] args) {
           EnPrint ep = new EnPrint(); //기본형
           ep.show();
10
           KrPrint kp = new KrPrint(); //기본형(변경 가능한 객체)
11
           kp.show();
12
13
           Name np = new Name("이만두"); //생성자 주입
           kp.setName(np);
15
16
           kp.show();
17
           np.setName("최말자"); //세터 주입
18
           kp.setName(np);
19
           kp.show();
20
21
          //스프링을 이용한 DI의 방법
22
          ApplicationContext cnt = new ClassPathXmlApplicationContext("beans.xml");
           KrPrint newp = (KrPrint)cnt.getBean("kp");
24
25
26
          newp.show();
```



스프링 DI의 구현(5)



```
p03
EnPrint.java
Main.java
Mame.java
Print.java
```



```
1 <?xml version="1.0" encoding="UTF-8"?>
20 < beans xmlns="http://www.springframework.org/schema/beans"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.springframework.org/schema/beans
         http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
      <bean id="hi" class="p01.HelloWorld">
          roperty name="message" value="안녕하세요~" />
      </bean>
10
[1⊖
      <bean id="name" class="p03.Name">
          cproperty name="name" value="스프링 세터" />
.3
      </bean>
      <bean id="name2" class="p03.Name">
          <constructor-arg name="name" value="스프링 생성자" />
16
1.7
      </bean>
.8
      <bean id="kp" class="p03.KrPrint">
          <constructor-arg ref="name2" />
                                                            Problems @ Javadoc Declaration
                                                            <terminated> Main (2) [Java Application]
      </bean>
22
                                                            EnPrint:홍길동
13 </beans>
                                                            KrPrint:홍길동
                                                            KrPrint:이만두
                                                            KrPrint: 최말자
                                                           KrPrint:스프링 생성자
```

```
1 package p03;
 3 import org.springframework.context.ApplicationContext;
 4 import org.springframework.context.support.ClassPathXmlApplicationContext;
   public class Main {
       public static void main(String[] args) {
           EnPrint ep = new EnPrint(); //기본형
           ep.show();
10
           KrPrint kp = new KrPrint(); //기본형(변경 가능한 객체)
11
           kp.show();
12
13
14
           Name np = new Name("이만두"); //생성자 주입
                                                            🔐 Problems @ Javadoc 📵 Declaration 튾
           kp.setName(np);
15
                                                            <terminated> Main (2) [Java Application] C:\Pr
16
           kp.show();
                                                            EnPrint:홍길동
17
                                                            KrPrint:홍길동
           np.setName("최말자"); //세터 주입
18
                                                            KrPrint:이만두
           kp.setName(np);
                                                            KrPrint: 최말자
19
                                                            KrPrint:스프링 세터
           kp.show();
20
21
22
           //스프링을 이용한 DI의 방법
           ApplicationContext cnt = new ClassPathXmlApplicationContext("beans.xml");
24
           KrPrint newp = (KrPrint)cnt.getBean("kp");
25
26
           newp.show();
27 }
```