

Spring Framework

스프링 이해 실습

2008.9

ygkim

zeroad@korea.com

Spring 이해

- 개발환경 구성
- BeanFactory
- ApplicationContext
- 빈 생성과 선언

실습 - 개발환경 구성

- JDK1.5 설치
- Tomcat 5.0 설치
- Eclipse3.3 설치
- Spring Framework 설치

실습 – Eclipse Workspace 구성

- Workspace 생성
 - c:\spring\workspace 생성
 - 공유 폴더로부터 Spring-App를 복사
- Spring-App Project 생성
 - File>import 실행
 - General > Existing Projects into Workspace
 - root directory 지정 : c:\spring\workspace
 - Spring-App import
- Project Property 설정
 - Property > libraries > Add Class Folder 실행
 - Spring-App>xml 추가
 - Property > libraries > Add Library > User library > User libraries 실행
 - New > spring-lib 추가
 - Add JARS > Spring-App\lib 하위 jar 파일 추가

- HelloService 인터페이스
- HelloServiceImpl 클래스 작성
 - HelloService 인터페이스 구현
- hello.xml 작성
 - HelloServiceImpl 타입의 helloService 빈으로 등록
- HelloClient 작성
 - hello.xml을 참조하는 ClassPathResource를 이용하여 XmlBeanFactory 생성
 - helloService 빈 참조하여 실행

- HelloService.java

```
package com.zeroad.spring.hello;  
  
public interface HelloService {  
    public void sayHello();  
}
```

- HelloServiceImpl.java

```
package com.zeroad.spring.hello;

public class HelloServiceImpl implements HelloService {
    private String name;

    public HelloServiceImpl() {}

    public void sayHello() {
        System.out.println("hello! " + getName());
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

- hello.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<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-2.0.xsd">

  <bean id="helloService"
    class="com.zeroad.spring.hello.HelloServiceImpl">
    <property name="name" value="ygkim" />
  </bean>

</beans>
```

- HelloClient.java

```
package com.zeroad.spring.client;

import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.ClassPathResource;

public class HelloClient {
    public static void main(String[] args) throws Exception {
        // 1. BeanFactory를 이용한 빈 참조
        BeanFactory factory = new XmlBeanFactory(new ClassPathResource("hello.xml"));
        HelloService service = (HelloService) factory.getBean("helloService");
        service.sayHello();
    }
}
```

- HelloContextClient 작성
 - hello.xml 파일을 참조하여 FileSystemXmlApplicationContext를 이용한 ApplicationContext 생성
 - ClassPathXmlApplicationContext 를 이용한 ApplicationContext 생성
 - helloService 빈 참조하여 실행

- HelloContextClient.java

```
package com.zeroad.spring.client;

import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.ClassPathResource;

public class HelloClient {
    public static void main(String[] args) throws Exception {
        // 1. BeanFactory를 이용한 빈 참조
        ...

        // 2. FileSystemXmlApplicationContext를 이용한 빈 참조
        String xmlFile = "D:\\spring\\workspace\\Spring01-Intro\\xml\\hello.xml";
        ApplicationContext fileCtx = new FileSystemXmlApplicationContext(xmlFile);
        HelloService service1 = (HelloService) fileCtx.getBean("helloService");
        service1.sayHello();

        // 3. ClassPathXmlApplicationContext를 이용한 빈 참조
        ApplicationContext classCtx = new ClassPathXmlApplicationContext("hello.xml");
        HelloService service2 = (HelloService) classCtx.getBean("helloService");
        service2.sayHello();
    }
}
```

- HelloServiceManager 작성
 - static HelloService getInstance() 제공
- spring.xml 작성
 - hello.xml 파일 import
 - helloService빈에 대한 alias 등록
 - HelloServiceManager 타입의 helloBean 등록
 - factory-method 속성에 getInstance 메소드 지정
- SpringClient 작성
 - spring.xml 파일을 참조하는 ClassPathXmlApplicationContext 생성
 - alias를 이용하여 helloService 빈 참조
 - HelloServiceManager를 이용하여 HelloService 인스턴스 생성

- HelloServiceManager.java

```
package com.zeroad.spring.hello;

public class HelloServiceManager {

    public static HelloService getInstance(){
        return new HelloServiceImpl();
    }

}
```

- spring.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN 2.0//EN"
"http://www.springframework.org/dtd/spring-beans-2.0.dtd">

<beans>

    <import resource="hello.xml" />

    <alias name="helloService" alias="hello"/>

    <bean id="helloBean"
        class="com.zeroad.spring.hello.HelloServiceManager"
        factory-method="getInstance" />

</beans>
```