使用IDEA开发Spring入门程序

Original 泰斗贤若如 泰斗贤若如 2019-12-09

使用IDEA开发Spring入门程序

前言

本文通过一个简单的入门程序向读者演示Spring框架的使用过程,具体如下:

一、使用IDEA创建Maven应用项目并导入相应模块

使用IDEA创建一个名为newspringdemo的项目,并将Spring的4个基础模块和第三方依赖模块commons-logging-1.2导入到Maven项目中,如下图所示:

```
🌉 newspringdemo [D:\MyNewWorld\Study_JAVA\newspringdemo] - newspringdemo - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
□ □ □ ← → □ √ M Tomcat 8 ∨ ▶ # □ □ □ ↓ / □ □ Q 10
newspringdemo m pom.xml
   ■ Project ▼
                                     ⊕ 😤 🌣 — mnewspringdemo
                                                            version>1.0-SNAPSHOT
    newspringdemo D:\MyNewWorld\Study JAVA\newspringd
                                                           <packaging>war</packaging>
       idea.idea
                                                           <name>newspringdemo Maven Webapp</name>
     ∨ III src
                                                           <url>http://www.example.com</url>
       ∨ main

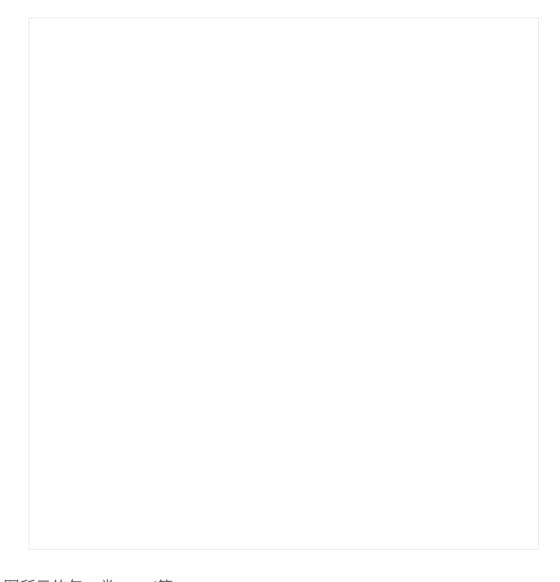
▼ Im webapp

                                                           (properties)
                                                             cproject.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

✓ III WEB-INF

                                                             <maven.compiler.source>1.7</maven.compiler.source>
<maven.compiler.target>1.7</maven.compiler.target>
                  web.xml
                                                           properties>
               index.jsp
                                                           <dependencies>
     > target
                                                             <dependency>
                                                              <groupId>junit</groupId>
        newspringdemo.iml
                                                              <artifactId>junit</artifactId>
<version>4.11</version>
       m pom.xml
                                                             <scope>test</scope>
</dependency>
     III External Libraries
     Scratches and Consoles
                                                             <dependency>
                                                               <groupId>commons-logging</groupId>
                                                              <artifactId>commons-logging</artifactId>
                                                               version>1.2</version>
                                                             </dependency>
</--Spring核心依赖模块 --
                                                   36
                                                             <dependency>
                                                              <groupId>org.springframework</groupId>
                                                              <artifactId>spring-core</artifactId>
<version>5.0.2.RELEASE</version>
                                                             </dependency>
. Z: Structure
                                                             <dependency>
.<groupId>org.springframework</groupId>
                                                              <artifactId>spring-context</artifactId>
                                                               <version>5.0.2.RELEASE
                                                             </dependency>
<dependency>
                                                   47
                                                              <groupId>org.springframework</groupId>
                                                              <artifactId>spring-beans</artifactId>
                                                               <version>5.0.2.RELEASE
                                                              /dependency>
                                                              <groupId>org.springframework</groupId>
                                                              <artifactId>spring-expression</artifac >
                                                   54
55
                                                               <version>5.0.2.RELEASE/version>
                                                             dependencies
 <dependencies>
     <dependency>
        <groupId>junit
        <artifactId>junit</artifactId>
        <version>4.11
        <scope>test</scope>
     </dependency>
     <!-- 第三方依赖commons-logging -->
      <dependency>
        <groupId>commons-logging
        <artifactId>commons-logging</artifactId>
        <version>1.2</version>
     </dependency>
     <!--Spring核心依赖模块 -->
      <dependency>
        <groupId>org.springframework
        <artifactId>spring-core</artifactId>
        <version>5.0.2.RELEASE
     </dependency>
     <dependency>
        <groupId>org.springframework
        <artifactId>spring-context</artifactId>
        <version>5.0.2.RELEASE
     </dependency>
     <dependency>
        <groupId>org.springframework
        <artifactId>spring-beans</artifactId>
```

2、项目案例



创建如上图所示的包、类、xml等

dao

TestDao代码如下:

```
package dao;
public interface TestDao {
    public void sayHello();
}
```

TestDaoImpl代码如下:

```
package dao;
import dao.TestDao;

public class TestDaoImpl implements TestDao {
    @Override
    public void sayHello() {
        System.out.println("Hello,Spring!!!");
    }
}
```

applicationContext.xml

代码如下:

注:配置文件的名称可以自定义,但习惯上命名为applicationContext.xml,有关Bean的创建我在后面会详细讲解,这里读者只需了解即可。

test

Test代码如下:

```
package test;
import dao.TestDao;
```

```
import dao.TestDaoImpl;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Test {
    public static void main(String[] args) {
        //初始化Spring容器ApplicationContext, 加载配置文件
        ApplicationContext applicationContext = new ClassPathXmlApplicationContext
        //通过容器获取test实例
        TestDao testDao =(TestDao)applicationContext.getBean("test");
        testDao.sayHello();
    }
}
```

运行结果

总结

以上内容就是一个简单的入门Spring程序,可以看出在main方法中并没有使用new创建TestDao-Impl类的对象,而是通过Spring容器来获取实现类对象,这就是Spring IoC的工作机制,在后面我会详细讲Spring IoC的工作机制。

往期推荐

- 《为什么很多程序员没有升级到架构师?》
- 《什么是数据结构》
- 《什么是抽象数据类型》
- 《算法核心——空间复杂度和时间复杂度超详细解析》
- 《学网络编程前的计算机知识普及》

- 《如何进行网络通信》
- 《计算机网络重磅来袭——一文让你拨开迷雾,直击网络原理》
- 《天降大祸于斯人也——幸捡一命,深思我的程序员生涯》
- 《在你迷茫时不如学好一门语言》
- 《流行框架知多少》
- 《致Java程序员兄弟们的一封信》