

# Spring 的第一个例子 HelloWorld

先以一个最著名的例子 HelloWorld 来看一下 Spring 如何开始。

创建 java project ， 命名为 HelloSpring。

在 src 文件夹下建一个包， 名为 com.zhangyun.First, 包中建两个 java 文件， 分别是 HelloWorld.java 和 MainAPP.java

创建如下：

HelloWorld.java

```
package com.zhangyun.First;
```

```
public class HelloWorld {
```

```
    private String message;
```

```
    public String getMessage() {  
        return message;  
    }
```

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

```
    public String toString()  
    {  
        return getClass().getName() + ": " + "message:" + message;  
    }  
}
```

MainAPP.java

```
package com.zhangyun.First;
```

```
import org.springframework.context.ApplicationContext;
```

```
import org.springframework.context.support.ClassPathXmlApplicationContext;
```

```
public class MainApp {
```

```
    public static void main(String[] args) {
```

```

        ApplicationContext context =
            new ClassPathXmlApplicationContext("Beans.xml");
        HelloWorld obj = (HelloWorld) context.getBean("helloworld");
        System.out.println(obj);
    }
}

```

现在程序是运行不了的，因为少了配置文件，Beans.xml。

在 src 目录下创建一个 xml 配置文件 Beans.xml，代码如下：

```

<?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-3.0.xsd">

    <bean id="helloworld" class="com.zhangyun.First.HelloWorld">
        <property name="message" value="HelloWorld!" />
    </bean>

</beans>

```

此时程序还是运行不了，这是为什么呢？因为我没有做好准备工作，就是导入相应的 spring 的 jar 包,我本地已经下载好 spring 相关的 jar 包，这里记录下 spring 官网：Spring 官网。

将 spring 的 jar 包加入路径后，一切就绪，通过 Myeclipse 运行 MainAPP.java 程序，结果如下：

```

七月18, 2018 6:38:29 下午 org.springframework.context.support.AbstractApplicationContext prepareRefresh
信息: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@6477463f: startup date
七月18, 2018 6:38:29 下午 org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions
信息: Loading XML bean definitions from class path resource [Beans.xml]
com.zhangyun.First.HelloWorld: message:HelloWorld!
https://blog.csdn.net/qq_36360784

```

结果如预料所示，打印出类信息以及 message 信息。

以下为总结和思考：

首先，创建一个 Spring 项目后，第一件事是将 Spring 相关的 jar 包导入路径，然后创建需要的类文件，类文件写好后，在包一级路径下创建 xml 文件，例子中我的 xml 文件名为 Beans.xml，你可以随便起名。

Beans.xml 用于给不同的 bean 分配唯一的 ID（Spring 中可称类为 bean），并且控制不

同值的对象的创建，而不会影响 Spring 的任何源文件。上例中，你可以通过 `property` 标签为 “`message`” 变量传递任何值，就可以输出信息的不同值，而不会影响的 `HelloWorld.java` 和 `MainApp.java` 文件。

```
<bean id="helloworld" class="com.zhangyun.First.HelloWorld">
    <property name="message" value="HelloWorld!" />
</bean>
```

`id` 是 `bean` 的唯一标识，`class` 标识创建的实例的类，要写全类名。`<property>` 标签中，指定 `message` 的值为“`HelloWorld`”，`<property>` 标签中，`message` 是 `key`, `value` 代表 `message` 的值（`value`），是键值对的表示。其实，`<property>` 标签是调用了 `HelloWorld` 类中的实例方法 `setMessage()`。当然，除了通过 `<property>` 指定键值对指定实例变量的值，还有其他方式，后面的文章会慢慢介绍。

关于主要程序有以下两个要点需要注意：

第一步是我们使用框架 API `ClassPathXmlApplicationContext()` 来创建应用程序的上下文。这个 API 加载 `beans` 的配置文件并最终基于所提供的 API，处理创建并初始化所有的对象，即在配置文件中提到的 `beans`。

第二步是使用已创建的上下文的 `getBean()` 方法来获得所需的 `bean`。这个方法使用 `bean` 的 `ID` 返回一个最终可以转换为实际对象的通用对象。一旦有了对象，你就可以使用这个对象调用任何类的方法。

-----  
作者：英雄策马走天涯

来源：CSDN

原文：[https://blog.csdn.net/qq\\_36360784/article/details/81103244](https://blog.csdn.net/qq_36360784/article/details/81103244)

版权声明：本文为博主原创文章，转载请附上博文链接！