

Spring经过大神们的构思、编码，日积月累而来，所以，对其代码的理解也不是一朝一夕就能快速完成的。源码学习是枯燥的，需要坚持！坚持！坚持！当然也需要技巧，第一遍学习的时候，不用关注全部细节，不重要的代码可以先忽略掉，达到理解大体的架子及流程，避免第一次就陷入某个坑里出不来。第二遍针对某个流程更深入的、有针对性的去分析学习，当然遇到某个实在过不去的坎可以标记，后面再思考，毕竟是别人设计的，有些不是那么容易理解的。次数多了，总会有收获！

首先，看一下Spring的最基本使用方式，直接看代码，

```
public class LoginService {

    public void login() {
        System.out.println("execute LoginService");
    }
}

public class LoginResource {

    private LoginService loginService;

    public LoginService getLoginService() {
        return loginService;
    }

    public void setLoginService(LoginService loginService) {
        this.loginService = loginService;
    }

    public void login() {
        loginService.login();
    }
}
```

applicationContext.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<beans xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.springframework.org/schema/beans"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans.xsd
">

    <bean id="loginService" name="loginService" class="spring.LoginService"/>
    <bean id="loginResource" name="loginResource" class="spring.LoginResource">
        <property name="loginService" ref="loginService"/>
    </bean>

</beans>
```

```
public class TestClient {

    @Test
    public void test() {
        ApplicationContext applicationContext = new
ClassPathXmlApplicationContext("applicationContext.xml");
        LoginResource loginResource = (LoginResource)
applicationContext.getBean("loginResource");
        loginResource.login();
    }
}
```

概括的描述一下Spring背后的操作，解析applicationContext.xml，将xml中定义的bean(如loginService和loginResource)解析成Spring内部的BeanDefinition，并以beanName(如loginService)为key，BeanDefinition(如loginService相应的BeanDefinition)为value存储到DefaultListableBeanFactory中的beanDefinitionMap(其实就是一个ConcurrentHashMap)中，同时将beanName存入beanDefinitionNames(List类型)中，然后遍历beanDefinitionNames中的beanName，进行bean的实例化并填充属性，在实例化的过程中，如果有依赖没有被实例化将先实例化其依赖，然后实例化本身，实例化完成后将实例存入单例bean的缓存中，当调用getBean方法时，到单例bean的缓存中查找，如果找到并经过转换后返回这个实例(如LoginResource的实例)，之后就可以直接使用了。

上面只是一个简单的描述，其中的实现过程还是很复杂的，后续篇幅将深入代码分析具体的实现过程，分析的内容不局限于IOC、AOP、事务处理、SpringMVC，内容来自于笔者的理解及分析，如有错误之处还请鞭策(吐槽)。