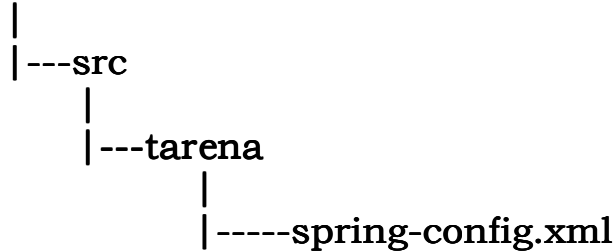


实验四.txt
实验四：IoC的实现ConstructorDI和SetDI

ConstructBeanDemo



加入Spring2的相关的JAR包

步骤一，创建HelloBean类

```
public class HelloBean {  
  
    private String name;  
    private String helloWorld;  
  
    public HelloBean(String name, String helloWorld) {  
        this.name = name;  
        this.helloWorld = helloWorld;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String name) {  
        this.name = name;  
    }  
  
    public String getHelloWorld() {  
        return helloWorld;  
    }  
  
    public void setHelloWorld(String helloWorld) {  
        this.helloWorld = helloWorld;  
    }  
}
```

实验四.txt

```
}
```

步骤二，创建spring配置文件

```
<?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">
```

```
<!--
```

方式一

```
<bean id="HelloBean" class="tarena.HelloBean">
```

```
<constructor-arg index="1">
```

```
<value>你好!</value>
```

```
</constructor-arg>
```

```
<constructor-arg index="0">
```

```
<value>Tom</value>
```

```
</constructor-arg>
```

```
</bean>
```

```
-->
```

```
<!-- 方式二 -->
```

```
<bean id="HelloBean" class="tarena.HelloBean">
```

```
<constructor-arg index="1" value="Hello"/>
```

```
<constructor-arg index="0" value="Tom"/>
```

```
</bean>
```

```
</beans>
```

步骤三，创建测试类

```
public class Test {
```

实验四.txt

```
public static void main(String[] args) {  
  
    ApplicationContext context = new  
    ClassPathXmlApplicationContext(  
        "tarena/spring-config.xml");  
  
    HelloBean helloBean = (HelloBean)  
    context.getBean("HelloBean");  
  
    System.out.println("用户名: " +  
    helloBean.getName());  
    System.out.println("话语: " +  
    helloBean.getHelloWorld());  
  
    }  
  
}
```

步骤四，结果
用户名: Tom
话语: Hello

SetBeanDemo
|
|---src

实验四.txt

```
|  
|---tarena  
|  
|-----spring-config.xml
```

加入Spring2的相关的JAR包

步骤一，创建HelloBean类

```
public class HelloBean {  
  
    private String name;  
    private String helloWorld;  
  
    public HelloBean(String name, String helloWorld) {  
        this.name = name;  
        this.helloWorld = helloWorld;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String name) {  
        this.name = name;  
    }  
  
    public String getHelloWorld() {  
        return helloWorld;  
    }  
  
    public void setHelloWorld(String helloWorld) {  
        this.helloWorld = helloWorld;  
    }  
  
}
```

步骤二，创建spring配置文件

实验四.txt

```
<?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="Time" class="java.util.Date"></bean>

    <bean id="HelloBean" class="tarena.HelloBean">
        <property name="age">
            <value>100</value>
        </property>
        <property name="time">
            <ref bean="Time" />
        </property>
    </bean>

</beans>
```

步骤三，创建测试类

```
public class Test {

    public static void main(String[] args) {

        ApplicationContext context = new
        ClassPathXmlApplicationContext(
            "tarena/spring-config.xml");

        HelloBean helloBean = (HelloBean)
        context.getBean("HelloBean");

        System.out.println("年龄： " +
        helloBean.getAge());
        System.out.println("时间： " +
        helloBean.getTime());
    }
}
```

实验四.txt

}
}

步骤四，结果

年龄：100

时间：Mon Nov 24 23:38:38 CST 2008