# 概念

Spring是一个基于IOC和AOP的结构J2EE系统的框架

IOC 反转控制 是Spring的基础，Inversion Of Control ，简单说就是创建对象由以前的程序员自己new 构造方法来调用，变成了交由Spring创建对象

DI 依赖注入 Dependency Inject. 简单地说就是拿到的对象的属性，已经被注入好相关值了，直接使用即可。

# 实践

1. 基本步骤

（1）新建项目

（2）下载相关jar包

（3）导入jar包：右键 project->properties->java build path->libaries->add external jars

（4）准备pojo Category

package com.how2java.pojo;

public class Category {

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

private int id;

private String name;

}

（5）编辑applicationContext.xml

在src目录下新建applicationContext.xml文件，applicationContext.xml是Spring的核心配置文件，通过关键字c即可获取Category对象，该对象获取的时候，即被注入了字符串"category 1“到name属性中：

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:aop="http://www.springframework.org/schema/aop"

xmlns:tx="http://www.springframework.org/schema/tx"

xmlns:context="http://www.springframework.org/schema/context"

xsi:schemaLocation="

http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-3.0.xsd

http://www.springframework.org/schema/aop

http://www.springframework.org/schema/aop/spring-aop-3.0.xsd

http://www.springframework.org/schema/tx

http://www.springframework.org/schema/tx/spring-tx-3.0.xsd

http://www.springframework.org/schema/context

http://www.springframework.org/schema/context/spring-context-3.0.xsd">

<bean name="c" class="com.how2java.pojo.Category">

<property name="name" value="category 1" />

</bean>

</beans>

（6）测试spring

package com.how2java.test;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.how2java.pojo.Category;

public class TestSpring {

public static void main(String[] args) {

ApplicationContext context = new ClassPathXmlApplicationContext(

new String[] { "applicationContext.xml" });

Category c = (Category) context.getBean("c");

System.out.println(c.getName());

}

}

2. 原理

获取对象的方式来进行比较

传统的方式：

通过new 关键字主动创建一个对象

IOC方式：

对象的生命周期由Spring来管理，直接从Spring那里去获取一个对象。 IOC是反转控制 (Inversion Of Control)的缩写，就像控制权从本来在自己手里，交给了Spring。

打个比喻：

传统方式：相当于你自己去菜市场new 了一只鸡，不过是生鸡，要自己拔毛，去内脏，再上花椒，酱油，烤制，经过各种工序之后，才可以食用。

用 IOC：相当于去馆子(Spring)点了一只鸡，交到你手上的时候，已经五味俱全，你就只管吃就行了。