1. 创建maven时的GroupId 和ArtifactId是什么意思？

GroupID是项目组织唯一的标识符，实际对应[Java](http://lib.csdn.net/base/javase)的包的结构，是main目录里java的目录结构。

ArtifactID就是项目的唯一的标识符，实际对应项目的名称，就是项目根目录的名称。

1. [context:exclude-filter 与 context:include-filter](http://blog.csdn.net/liuwenbo0920/article/details/7260013)

**(1) 在主容器中（applicationContext.xml），将Controller的注解打消掉**

1. **<context:component-scan** base-package="com"**>**
2. **<context:exclude-filter** type="annotation" expression="org.springframework.stereotype.Controller" **/>**
3. **</context:component-scan>**

**(2) 而在springMVC配置文件中将Service注解给去掉**

1. **<context:component-scan** base-package="com"**>**
2. **<context:include-filter** type="annotation" expression="org.springframework.stereotype.Controller" **/>**
3. **<context:exclude-filter** type="annotation" expression="org.springframework.stereotype.Service" **/>**
4. **</context:component-scan>**

因为spring的context是父子容器，所以会产生冲突，Controller会先被扫描装配，而此时的Service还没有进行事务的加强处理惩罚，获得的将是原样的Service（没有经过事务加强处理惩罚，故而没有事务处理惩罚才能） ，最后才是applicationContext.xml中的扫描设备进行事务处理惩罚

1. 在实体类映射中为什么用set而不是list？

防止保存重复对象。

1. mapper.xml中的resultMap 和resultType

MyBatis中在查询进行select映射的时候，返回类型可以用resultType，也可以用resultMap，resultType是直接表示返回类型的，而resultMap则是对外部ResultMap的引用，但是resultType跟resultMap不能同时存在。

在MyBatis进行查询映射的时候，其实查询出来的每一个属性都是放在一个对应的Map里面的，其中键是属性名，值则是其对应的值。当提供的返回类型属性是resultType的时候，MyBatis会将Map里面的键值对取出赋给resultType所指定的对象对应的属性。所以其实MyBatis的每一个查询映射的返回类型都是ResultMap，只是当我们提供的返回类型属性是resultType的时候，MyBatis对自动的给我们把对应的值赋给resultType所指定对象的属性，而当我们提供的返回类型是resultMap的时候，因为M

1. 为何<insert><delete><update> 语句的返回值类型为什么是int?

增删改操作实际上返回的是操作的条数。而Mybatis框架本身是基于JDBC的，所以此处也沿袭这种返回值类型。

1. #{} 和 ${}

#{value jdbcType = valuetype}：jdbcType 表示该属性的数据类型在[**数据库**](http://lib.csdn.net/base/mysql)中对应的类型，如 #{user jdbcType=varchar} 等价于 String username；

${value } : **这种方式不建议大量使用**，可能会发送sql注入而导致安全性问题。一般该取值方式可用在非经常变化的值上，如orderby ${columnName}；

1. controller之间的参数传递

public String con1(){

String message = message;

return “forword：con2.do？message=”message；

}

public String con2(String message){

message=message;

return “路径”；

}

即控制器之间通过url传值

1. 如何从子窗口执行父窗口的操作

**parent**.*showFull*();

表示调用父类的这个方法

1. 获取当前点击事件的数据

标签的点击事件

$(pro).click(function(){

获取当前标签的属性名为name的值

$(this).attr(name);

})

1. spring mvc如何从页面获取对象？

在页面上只需要提供对象的属性名称，如：

<input name=”properties”>

就会自动把对应的属性封装进对象中。

springmvc不支持 name=”bean.properties”

1. mybatis在执行insert操作时返回插入值生成的主键id值

<**insert id="addOneStudent" parameterType="com.ssm.student.entity.StudentEntity" useGeneratedKeys="true" keyProperty="studentEntity.studentEntityId" keyColumn="t\_student\_id"** >

关键词说明：

|  |  |
| --- | --- |
| id | 对应mapper接口中的方法，名称必须一致 |
| parameterType | 入参的全限定类名或类型别名 |
| keyColumn | 设置数据表自动生成的主键名。对特定数据库（如PostgreSQL），若自动生成的主键不是第一个字段则必须设置 |
| keyProperty | 默认值unset，用于设置getGeneratedKeys方法或selectKey子元素返回值将赋值到领域模型的哪个属性中 |
| useGeneratedKeys | 取值范围true|false(默认值)，设置是否使用JDBC的getGenereatedKeys方法获取主键并赋值到keyProperty设置的领域模型属性中。[MySQL](http://lib.csdn.net/base/mysql)和SQLServer执行auto-generated key field，因此当数据库设置好自增长主键后，可通过JDBC的getGeneratedKeys方法获取。但像Oralce等不支持auto-generated key field的数据库就不能用这种方法获取主键了 |

返回主键值已经赋值在对象中，而不是赋值在方法返回值中。

1. 修改select下拉菜单默认值

$(**"#Id option[value="**+value+**"]"**).attr(**"selected"**,**true**);

选定id为Id的下拉多选框，其值为value的option为默认选项

 设置下拉框text属性为5的选项 选中  
 $("#selectID option[text=5]").attr("selected", "selected");

1. MyBatis中的动态sql语句

<**select id="queryStuPage" resultMap="stuEntity"**>  
 SELECT  
 s.identity\_Card  
 FROM t\_student s where 1=1

<**if test="****param!=null and param!='' and param!=1000 "**>  
 and s.major\_id= #{ **param** }  
 </**if**>  
 </**select**>

示例在select语句中：

test中为条件；param为方法的参数