

# 列设置使用Java类自定义规则

- [示例说明](#)
- [实现步骤](#)

创建Excel导入模板，在进行“列设置”时我们会遇到这样的问题，数据表中某个字段无法简单地绑定到Excel的某列上，而应该绑定到多个列经过计算后的结果上。

这时我们就需要自定义一个Java类，然后在“绑定输入”下拉框中选择自定义的Java类了。

如果有类似需求，可以参考下面的示例。

## 示例说明

该示例实现为每个学校导入、更新行政单位及其负责人。Excel数据如下：

	A	B	C	D
1	行政单位名称	单位号	单位职能	单位负责人
2	党政办公室	001	其他	杨海文
3	组织部（统战部）	002	其他	赵为民
4	宣传部（新闻信息中心）	003	其他	宁习洲
5	监察处（纪委办公室）	004	其他	陈映斌

目标表结构：

```
CREATE
TABLE xxxgxzdw
(
    C_XXDM VARCHAR(200) COMMENT '学校代码',
    C_XZDWMC VARCHAR(200) COMMENT '行政单位名称',
    C_DWH VARCHAR(200) COMMENT '单位号',
    C_DWZN VARCHAR(200) COMMENT '单位职能',
    C_DWFZR VARCHAR(200) COMMENT '单位负责人',
    C_ID VARCHAR(200) NOT NULL COMMENT '唯一id',
    PRIMARY KEY (C_ID)
)
ENGINE=INNODB DEFAULT CHARSET=utf8;
```

其中“学校代码”，需要根据用户选择的值导入，“唯一id”需要根据“学校代码”和“单位号”组合而成。

## 实现步骤

1、新建扩展包“DataImporterDemo”；

1) 需要在extension.xml中设置depends=“DataImporter”，或者将本扩展包的加载优先级设置的低一些。如果是独立开发扩展包，为了正常编译，需要将DataImporter.jar放到扩展包的lib-compile目录（打包时不需要打包到扩展包）

```
<extension name="DataImporterDemo" depends="DataImporter" alias="Excel数据导入示例"/>
```

2) 需要将applicationContext.xml的内容全部替换为如下代码：

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN 2.0//EN" "http://www.springframework.org/dtd/spring-beans-2.0.
dtd">
<beans>
    <bean id="framework" class="smartbi.framework.Framework" factory-method="getInstance">
        <property name="modules">
            <map>
                <entry><key><value>DAQDemoModule</value></key><ref bean="DAQDemoModule" /><
/entry>
            </map>
        </property>
    </bean>
    <bean id="DAQDemoModule" class="smartbi.demo.DAQDemoModule" factory-method="getInstance">
    </bean>
</beans>
```

2、在扩展包中，新建一Java类：“smartbi.demo.XZDWID”，行政单位ID类。完整的代码如下：

```
package smartbi.demo;

import smartbi.daq.Excelimport.ICellDataGenerator;
import smartbi.daq.Excelimport.ImportFileConfig;
import smartbi.daq.Excelimport.RowData;
import smartbi.net.sf.json.JSONObject;
import smartbi.util.StringUtil;

public class XZDWID implements ICellDataGenerator {

    @Override
    public String getCellValue(JSONObject colConfig, RowData rowData, ImportFileConfig config) {
        String xxdm = rowData.getCell("C_XXDM", true); //
        String dwh = rowData.getCell("C_DWH"); //
        if (!StringUtil.isNullOrEmpty(xxdm) && !StringUtil.isNullOrEmpty(dwh)) {
            return xxdm + "_" + dwh;
        }
        return null;
    }
}
```

3、在扩展包中，新建Module类：“smartbi.demo.DAQDemoModule”，注册自定义JAVA类，名为“学校行政单位id”。完整的代码如下：

```
package smartbi.demo;

import smartbi.daq.DataAcquisitionModule;
import smartbi.framework.IModule;

public class DAQDemoModule implements IModule {

    private static DAQDemoModule instance = new DAQDemoModule();

    /**
     * @return DataAcquisitionModule
     *
     */
    public static DAQDemoModule getInstance() {
        return instance;
    }

    @Override
    public void activate() {
        DataAcquisitionModule.getInstance().registerExcelCellDataGenerator("id", "id", new XZDWID());
    }
}
```

4、编译打包DataImporterDemo扩展包，然后部署该扩展包。

- 5、在“Excel数据导入”界面添加导入规则。
- 6、点击“列设置”标签页，设置列映射。其中“唯一id”字段的“绑定输入”选择“使用JAVA类”的方式，并选择上面自定义的“学校行政单位id”。
- 7、点击右下角 **保存** 按钮，弹出保存对话框，输入名称“演示绑定自定义java类的导入模板”，保存Excel导入模板。
- 8、双击该模板执行导入，选择本地“表1-3学校相关行政单位.xls”文件。
- 9、到数据库中查看导入后的结果：

	C_XXDM	C_XZDWMC	C_DWH	C_DWZN	C_DWFZR	C_ID
<input type="checkbox"/>	4144010558	党政办公室	001	其他	杨海文	4144010558_001
<input type="checkbox"/>	4144010558	组织部（统战部）	002	其他	赵为民	4144010558_002
<input type="checkbox"/>	4144010558	宣传部（新闻信息中心）	003	其他	宁习洲	4144010558_003
<input type="checkbox"/>	4144010558	监察处（纪委办公室）	004	其他	陈映斌	4144010558_004
<input type="checkbox"/>	4144010558	学生工作部	005	学生管理	张超	4144010558_005
<input type="checkbox"/>	4144010558	保卫处（武装部）	006	其他	胥来昌	4144010558_006

PS: [HelloExt.rar](#) 此附件为示例的源码。