第四章 代码生成器-使用规则

[一].建表规范

[1].表必须有唯一主键: OBID（字符类型 36位）
备注: 主键采用UUID方式生成

[2].表必需字段（创建人，创建时间等..）

[3].表字段必须有注释
备注: JSP页面字段文本，是根据表字段注释来生成

[4]. 主表和子表的外键字段名字，必须相同（除主键OBID外）

[5]. 子表引用主表主键OBID作为外键，外键字段必须以OBID结尾

注释：请按照建表模板来创建新表，模板表中原有的字段，生成器会过滤不在页面生成

建表模板：

字段名	类型	长度	备注	主键
OBID	varchar	36	主键	TURE
其他字段..				
CRTUSER	varchar	36	创建人	
CRTUSER_NAME	varchar	32	创建人名字	
CREATE_DT	datetime	0	创建时间	
MODIFIER	varchar	36	修改人	
MODIFIER_NAME	varchar	32	修改人名字	
MODIFY_DT	datetime	0	修改时间	
DELFLAG	int	2	删除标记	
DEL_DT	datetime	0	删除时间	

[二].页面生成规则

说明：JSP页面字段的文本内容，取表字段的注释前6位字符(如果建表字段注释为空，则页面字段文本会为空)

A.默认生成的JSP页面前五个字段为必须项，其他字段为非必须输入（需要自己手工加）

B.数据库字段类型为：datetime -->对应页面字段，会自动追加[年月日-时分秒]时间控件

C.数据库字段类型为：date -->对应页面会字段，自动追加[年月日]时间控件

D.数据库字段类型为：Int/Number-->对应页面字段，会自动追加数字校验（不允许输入小数）

E.数据库字段类型为：float/double/decimal-->对应页面页面字段，会自动追加数字校验（允许输入小数）

F.如果表字段为字符类型，并且设置了长度，页面输入框会自动设置maxlength对应表字段长度

[三].代码生成器工具类

A.单表模型工具类: com.test.CodeUtil

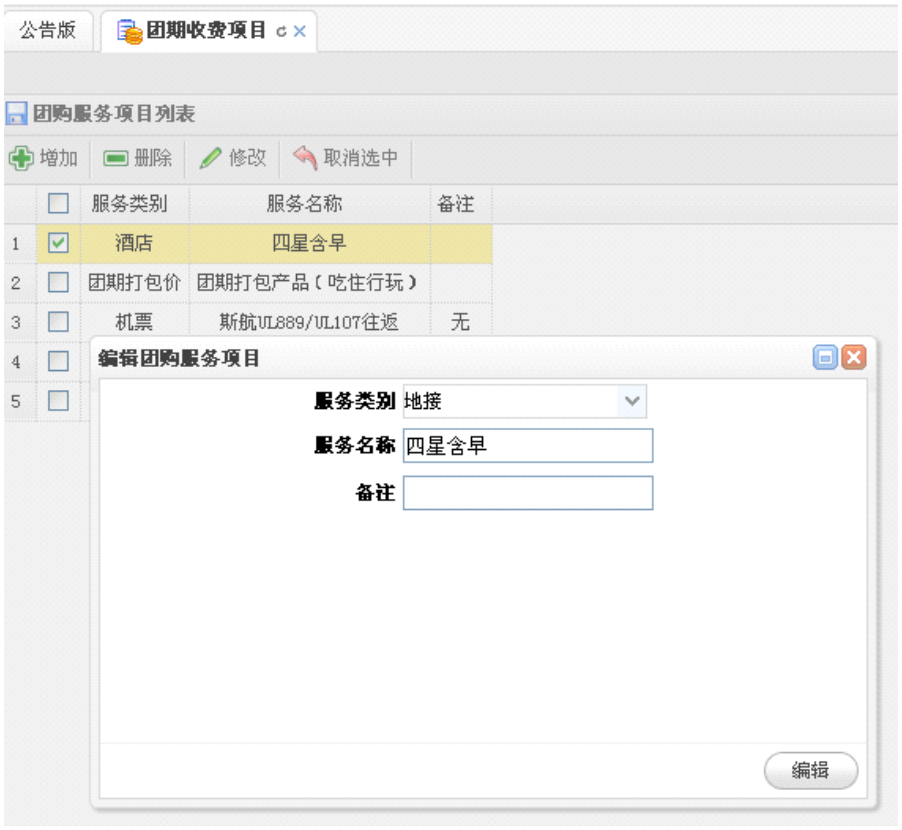
代码生成器界面：

A.动态选择需要生成文件(ServiceI\ServiceImpl\Jsp\Action\Entity\Page)

B.动态选择JSP模板（两种页面风格：行编辑/详细页编辑）



A-1.详细页编辑页面风格：



A-2.行编辑页面风格



B.一对多模型(父子表)工具类：生成的页面明细可以动态添加行

com.test.CodeOneToMainUtil[一对多的增删改查]

第一步:设置一对多代码生成需要参数，执行方法生成代码

```
public static void main(String[] args) {
    //--------------------------------------------------------
    //第一步：设置主表
    CodeParamEntity codeParamEntityIn = new CodeParamEntity();
    codeParamEntityIn.setTableName("t60_gbuy_order");//主表[表名]
    codeParamEntityIn.setEntityName("GbuyOrder");     //主表[实体名]
    codeParamEntityIn.setEntityPackage("order");      //主表[包名]
    codeParamEntityIn.setFtlDescription("订单抬头");   //主表[描述]
    codeParamEntityIn.setFtl_mode(CodeGenerateOneToMany.FTL_MODE_B);//主表[模板 A:明细单页布局显示 B:明细采用Tab布局展现]

    //第二步：设置子表集合
    List<SubTableEntity> subTabParamIn = new ArrayList<SubTableEntity>();
    //[1].子表一
    SubTableEntity po = new SubTableEntity();
    po.setTableName("t60_gbuy_order_custom");//子表[表名]
    po.setEntityName("GbuyOrderCustom");      //子表[实体名]
    po.setEntityPackage("order");             //子表[包]
    po.setFtlDescription("订单客户明细");       //子表[描述]
    po.setForeignKeys(new String[]{"GORDER_OBID","GO_ORDER_CODE"});//子表[外键:与主表关联外键]
    subTabParamIn.add(po);
    //[2].子表二
    SubTableEntity po2 = new SubTableEntity();
    po2.setTableName("t60_gbuy_order_product");      //子表[表名]
    po2.setEntityName("GbuyOrderProduct");           //子表[实体名]
    po2.setEntityPackage("order");                   //子表[包]
    po2.setFtlDescription("订单产品明细");             //子表[描述]
    po2.setForeignKeys(new String[]{"GORDER_OBID","GO_ORDER_CODE"});//子表[外键:与主表关联外键]
    subTabParamIn.add(po2);
    codeParamEntityIn.setSubTabParam(subTabParamIn);

    //第三步：一对多(父子表)数据模型,代码生成
    CodeGenerateOneToMany.oneToManyCreate(subTabParamIn, codeParamEntityIn);
}
```

第二步:配置菜单，查看生成功能

