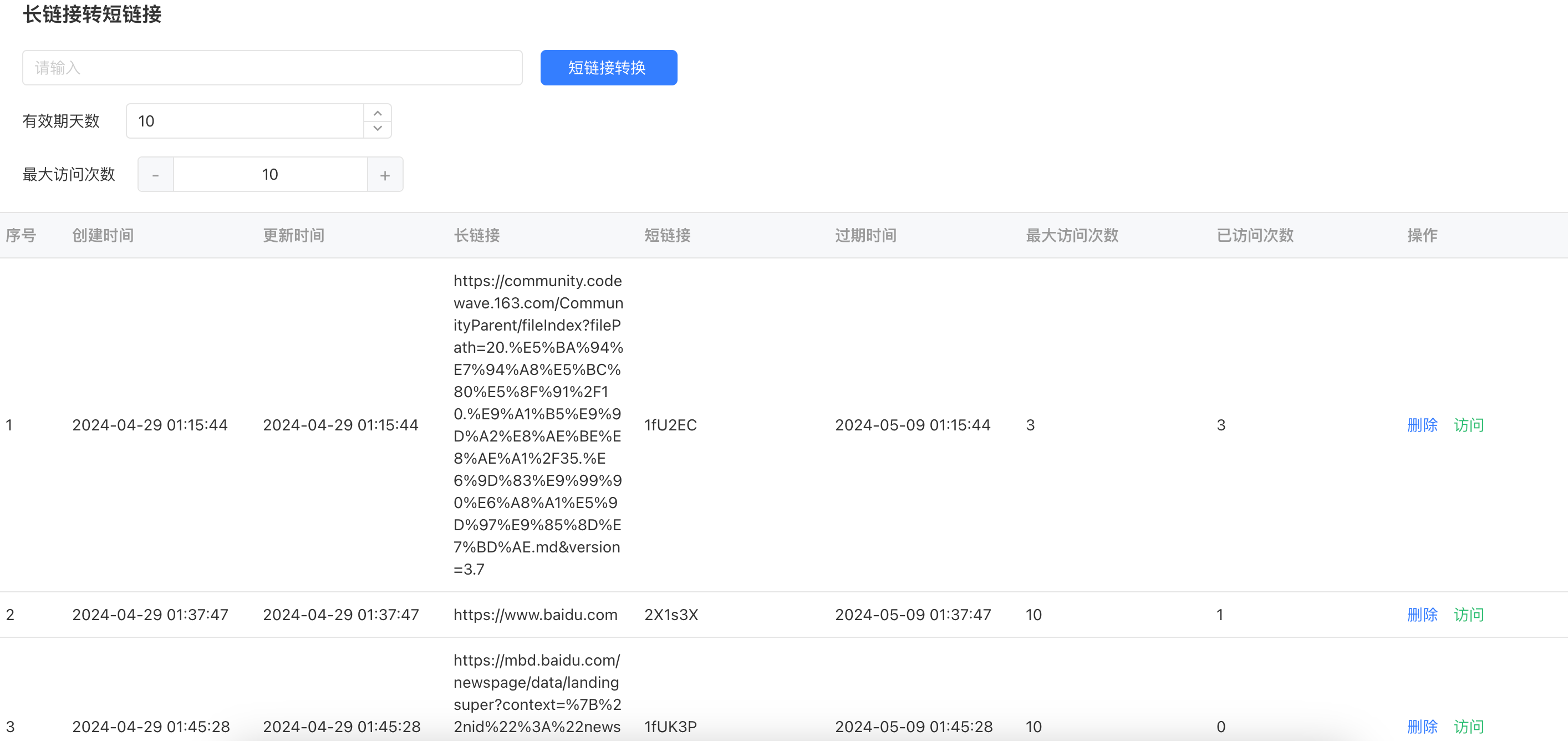
依赖库ShortUrl使用文档说明

## 1. 背景

ShortUrl是一个长链接转短链接的依赖库，提供了短链接生成方法，可以限制短链接有效期以及访问次数。

示例效果如下：



## 2. 依赖库方法

### saveUrlMap

生成短链接哈希值  
入参：

* ShortUrlRequest request
* Function<Link, Link> saveShortUrlLogic 入库保存逻辑
* Function<String, Boolean> checkLongUrlExistLogic 检查长链接是否存在逻辑

出参： Link

### getLongUrl

获取长链接

入参：

* String shortCode
* Function<String, Link> findLinkByShortCode 根据短链接码查询短链接逻辑
* Function<Link, Boolean> updateLink 更新短链接访问次数逻辑

出参： String longUrl

## 3. 示例

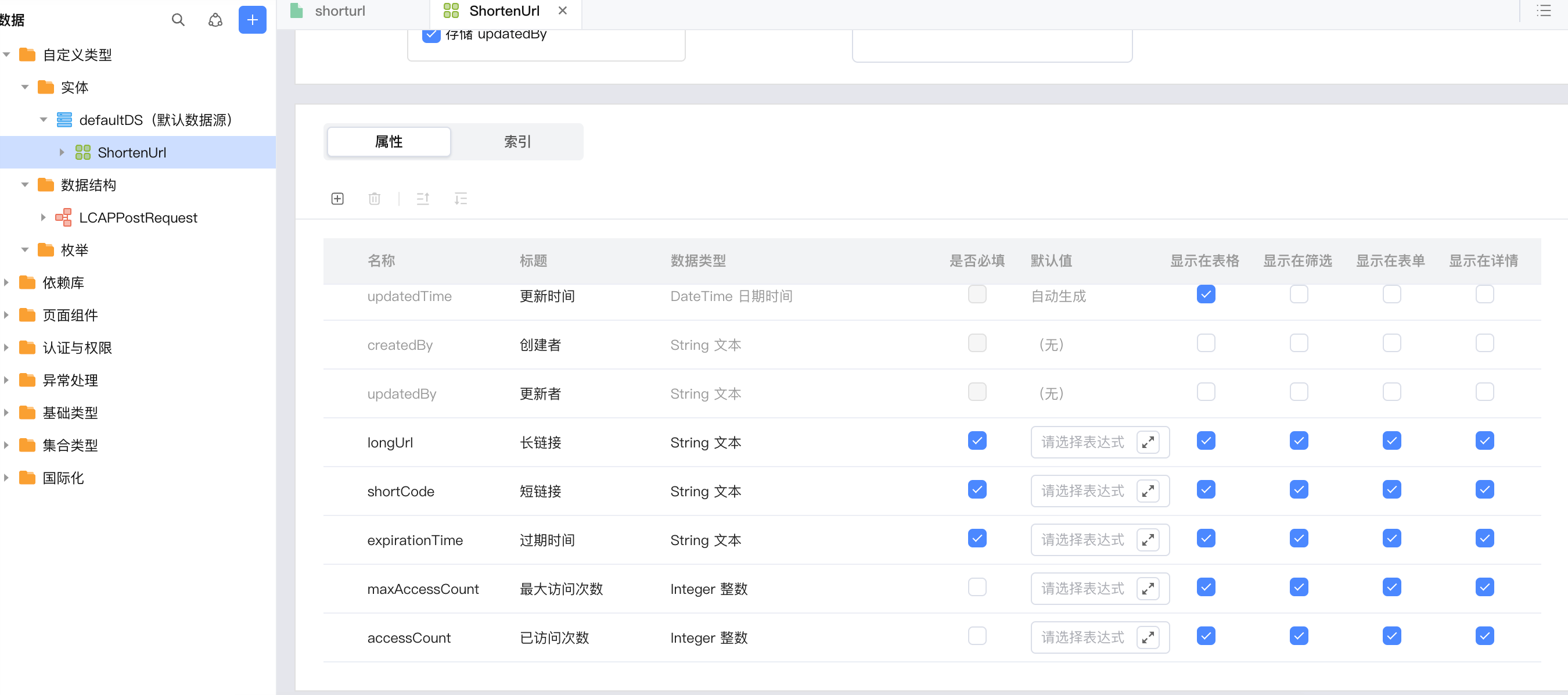
### 3.1 引入依赖库

在依赖库管理导入shorturl依赖库

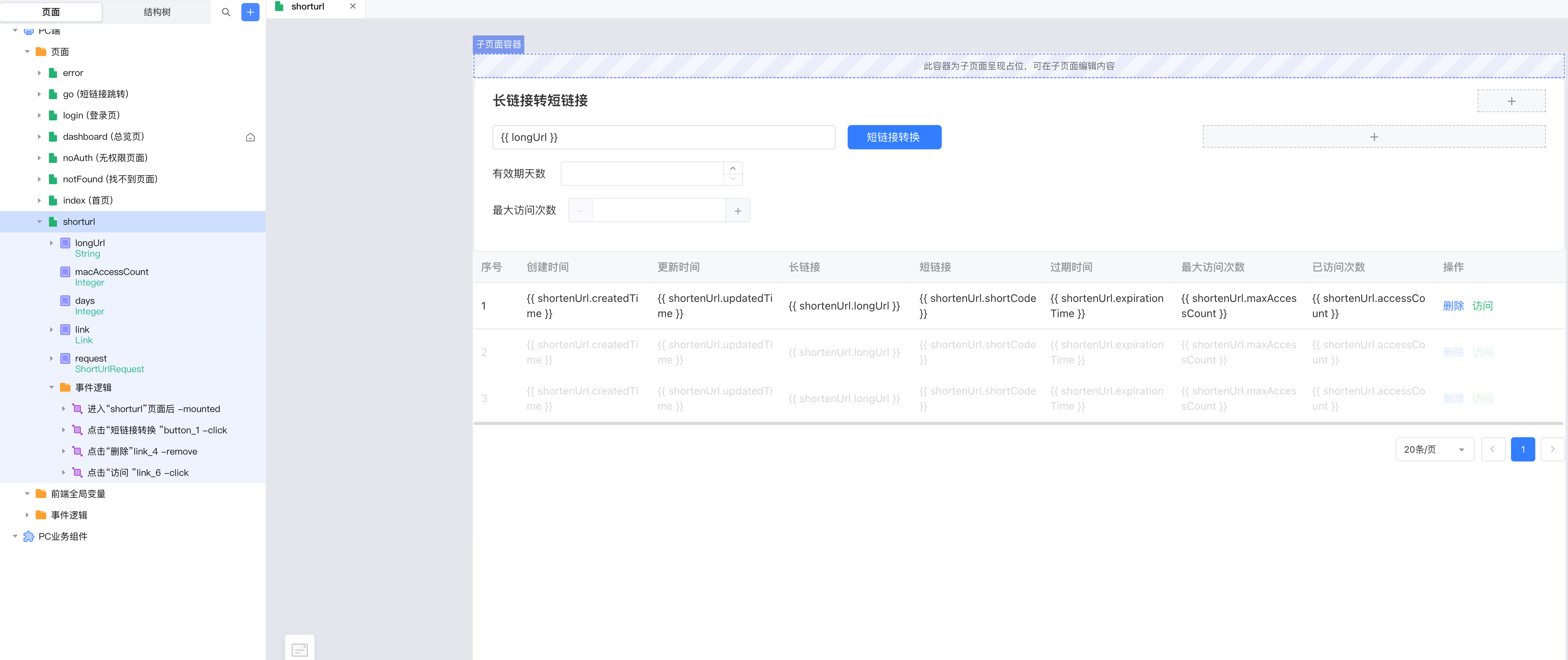


### 3.2 页面绑定及调用依赖库示例

添加数据库实体ShortenUrl, 字段如图



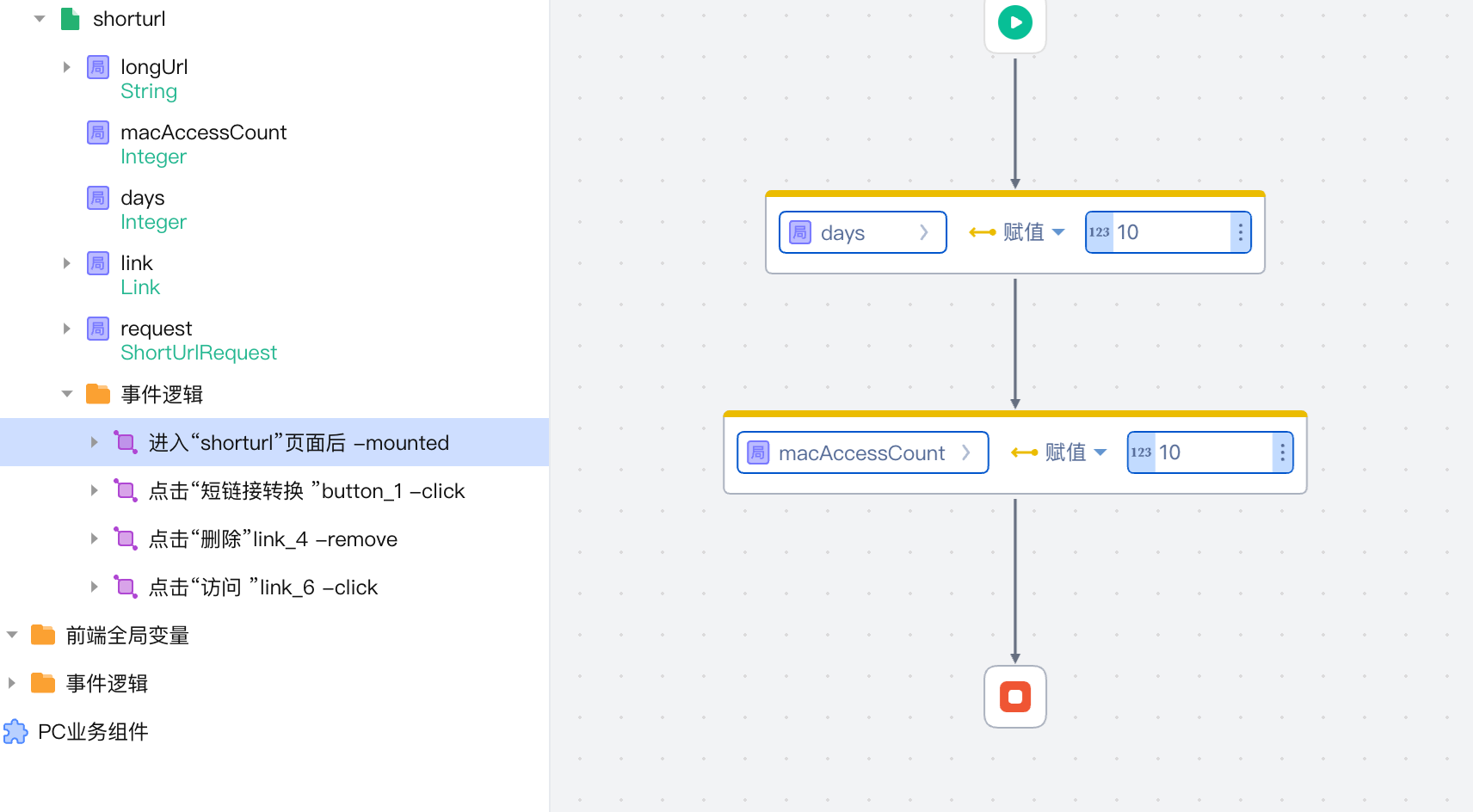
页面上新建表单如图， 并新建局部变量并绑定到表单输入项中。



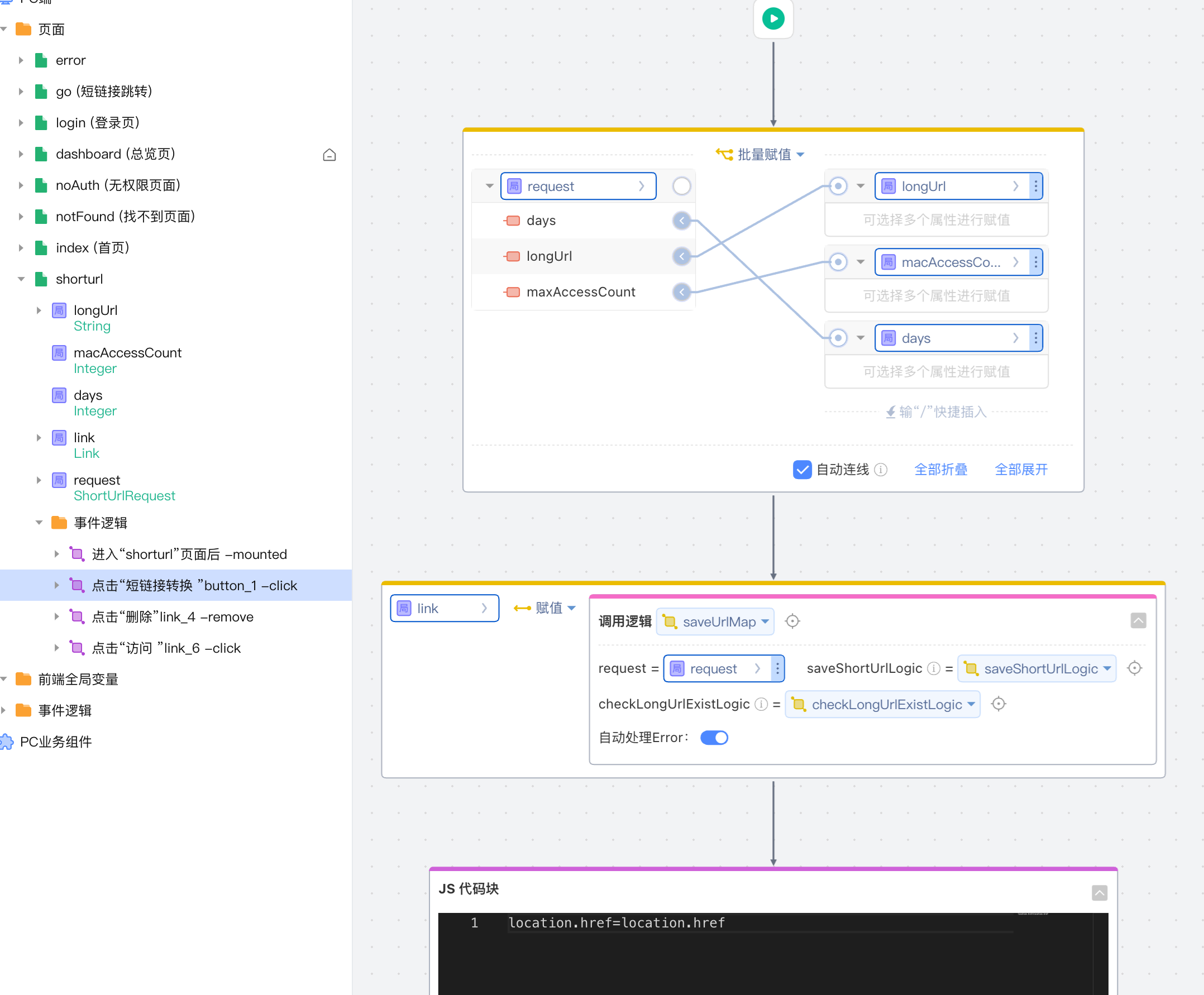
将数据源拖入页面用来展示已添加的数据。示例使用简单视图。

该页面逻辑有：

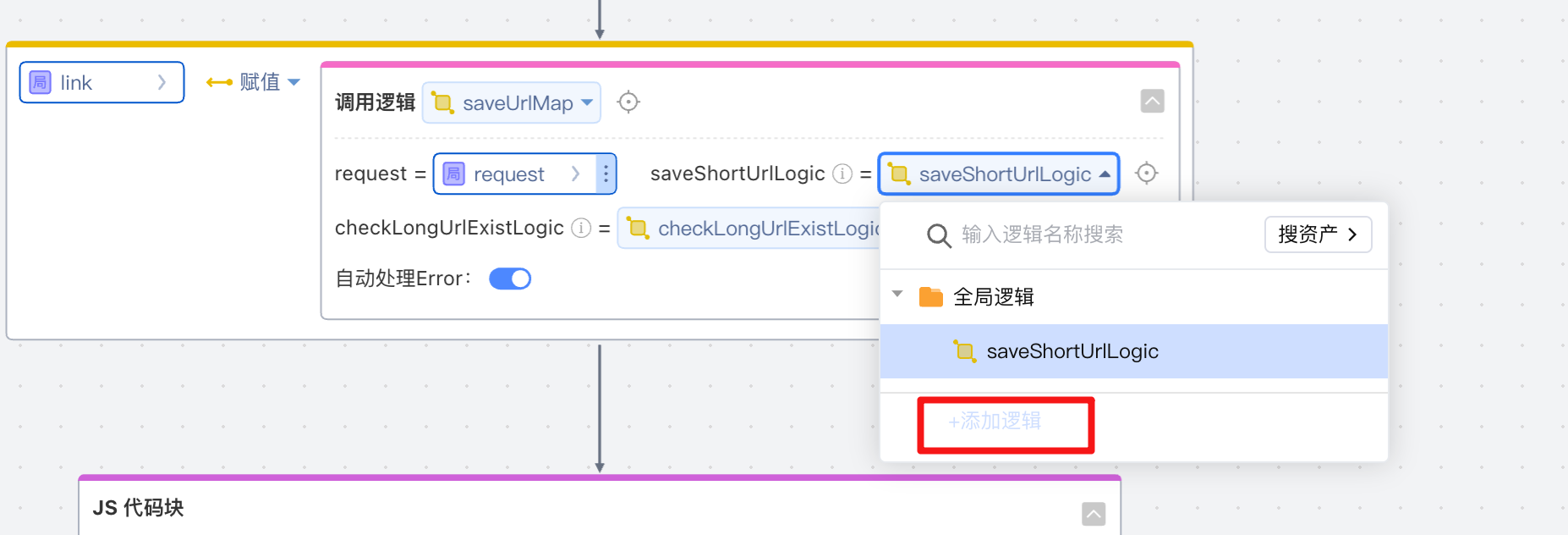
初始化表单的值：



点击短链接转换按钮后的事件：



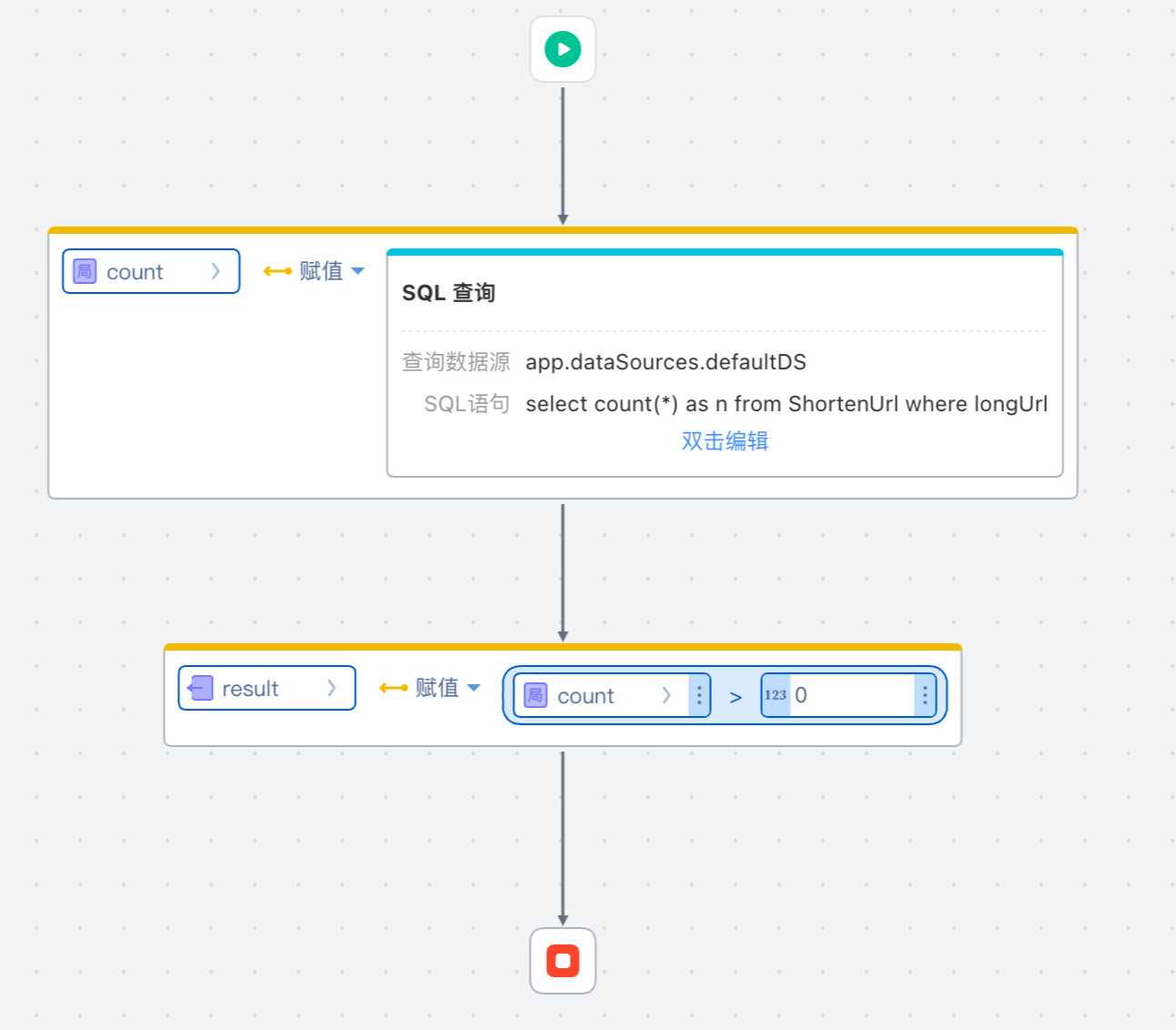
其中调用了依赖库的逻辑saveUrlMap，此时按提示新建逻辑，即可自动建立逻辑带入入参和出参。



其逻辑如图：

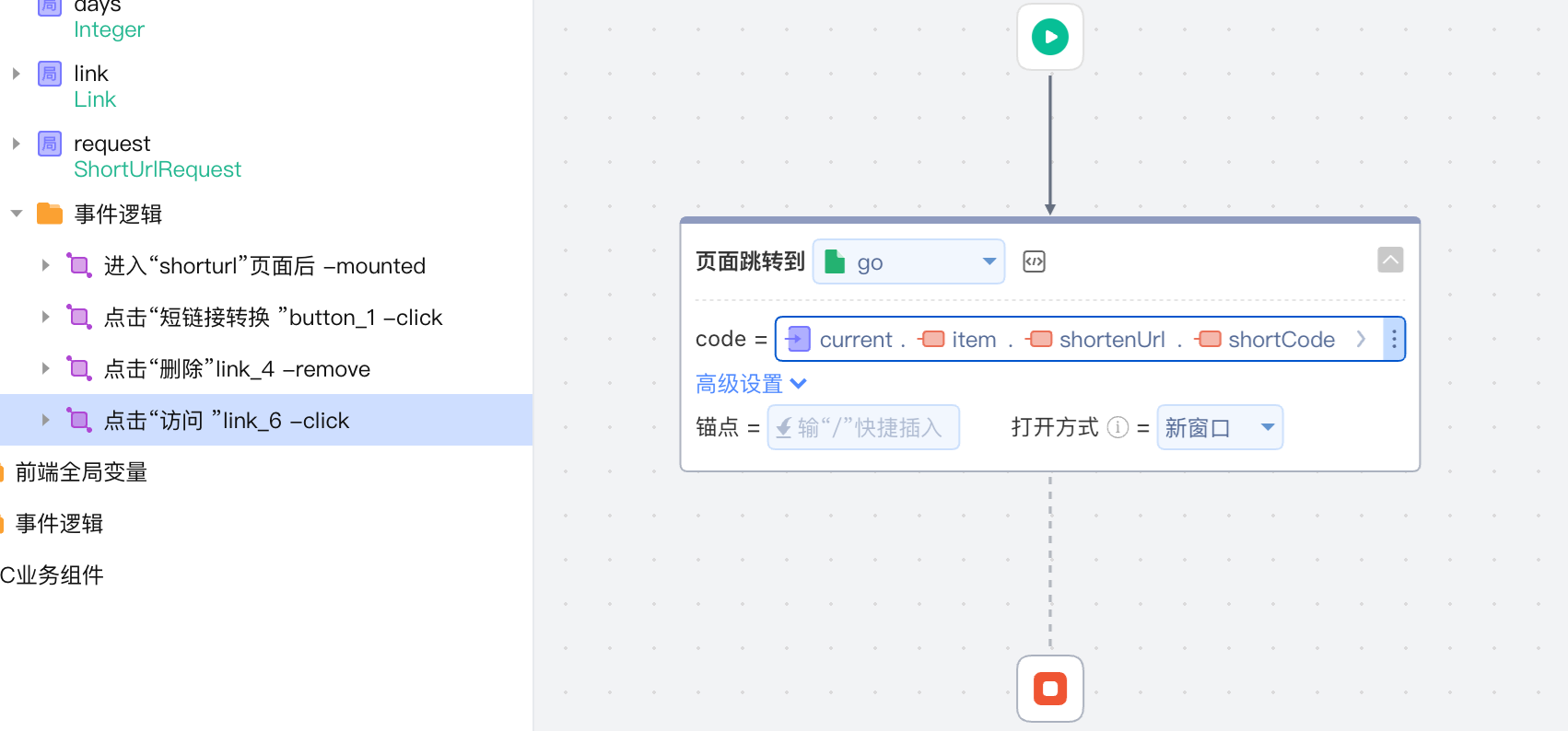


checkLongUrlExist逻辑如图：



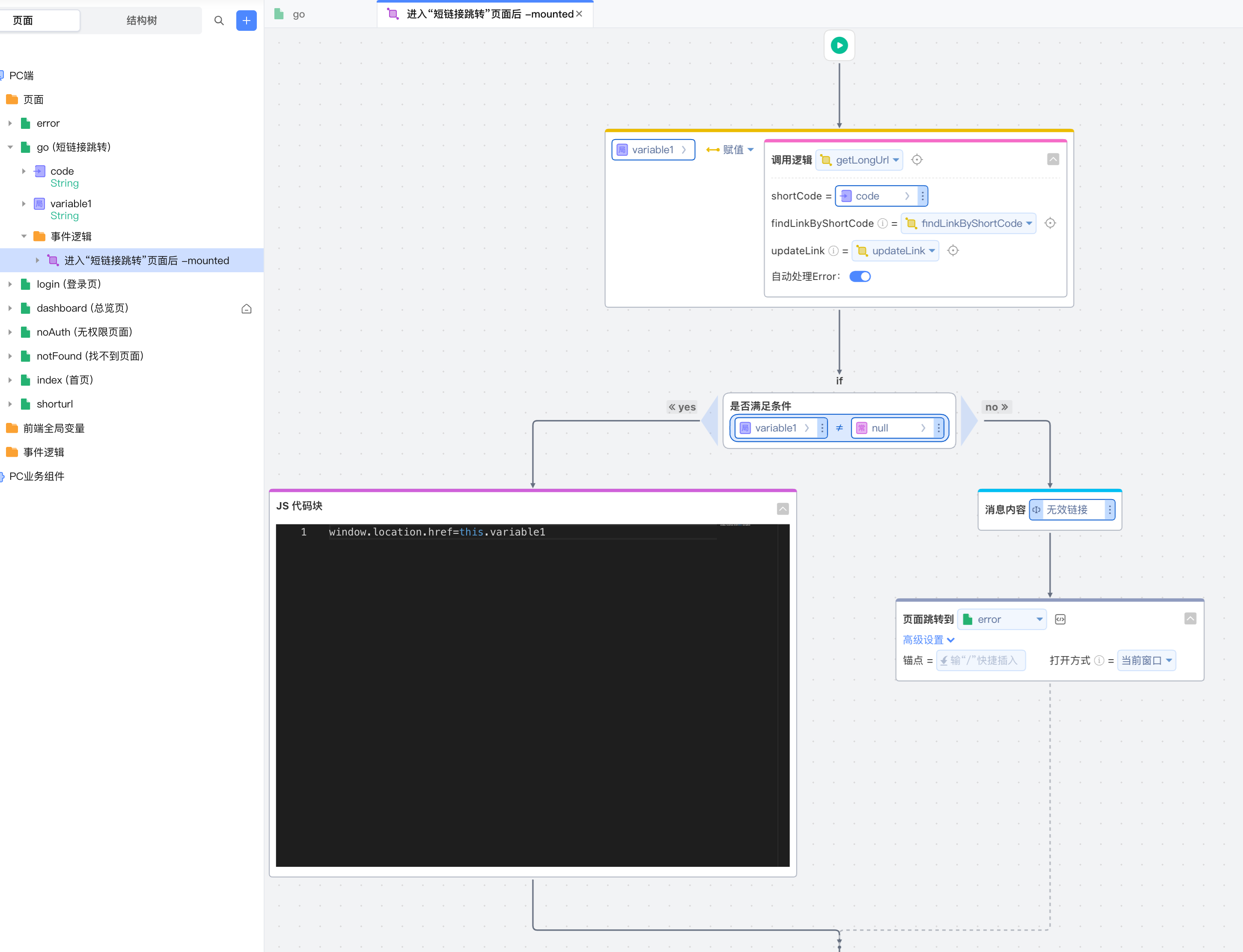
点击删除事件逻辑为表单自动生成；

点击访问事件逻辑如图：



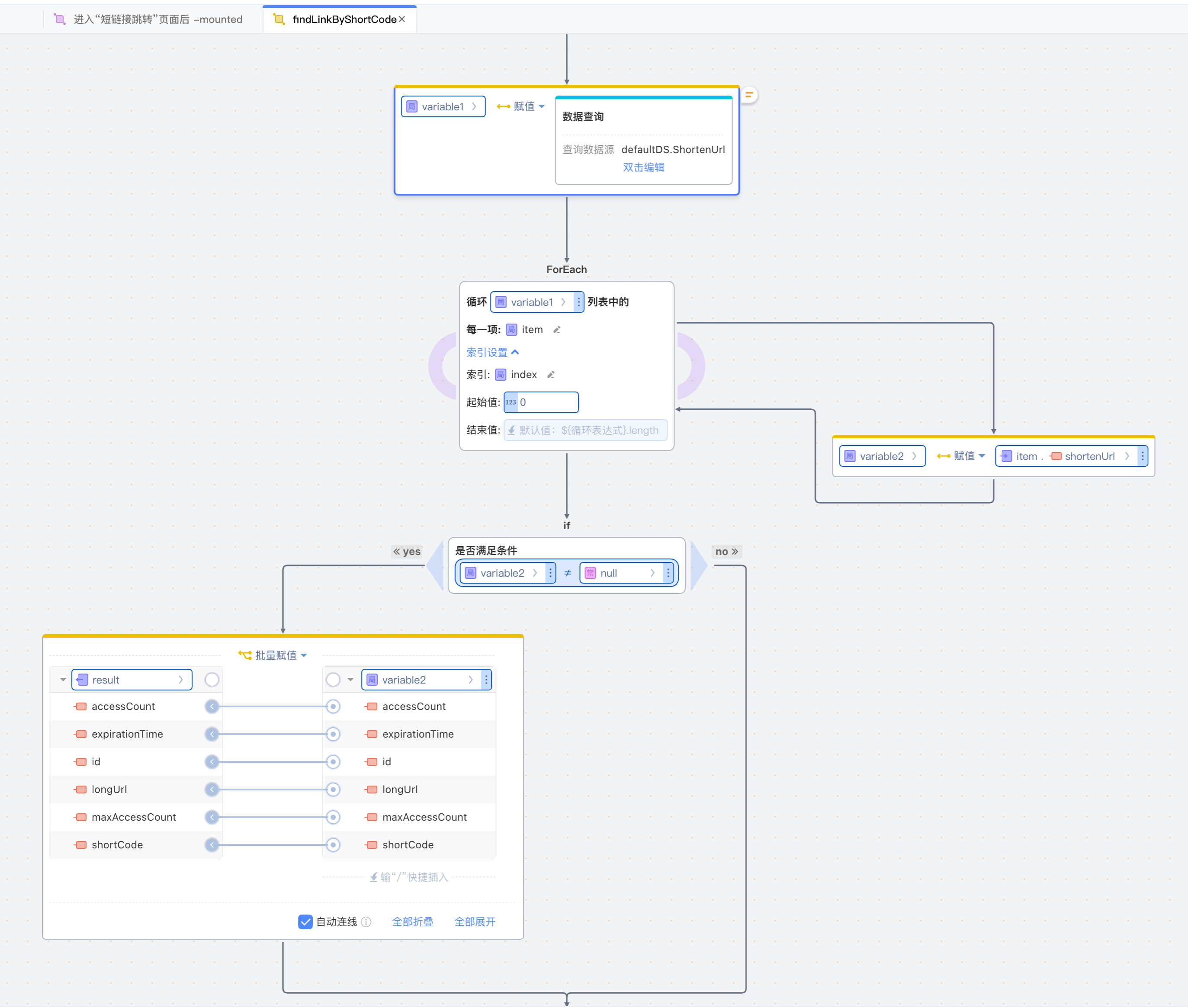
打开自定义的go页面，入参为code=xxxxx， 参数为当前行对象的shortCode

页面go无需设计 ，添加进入页面的事件：



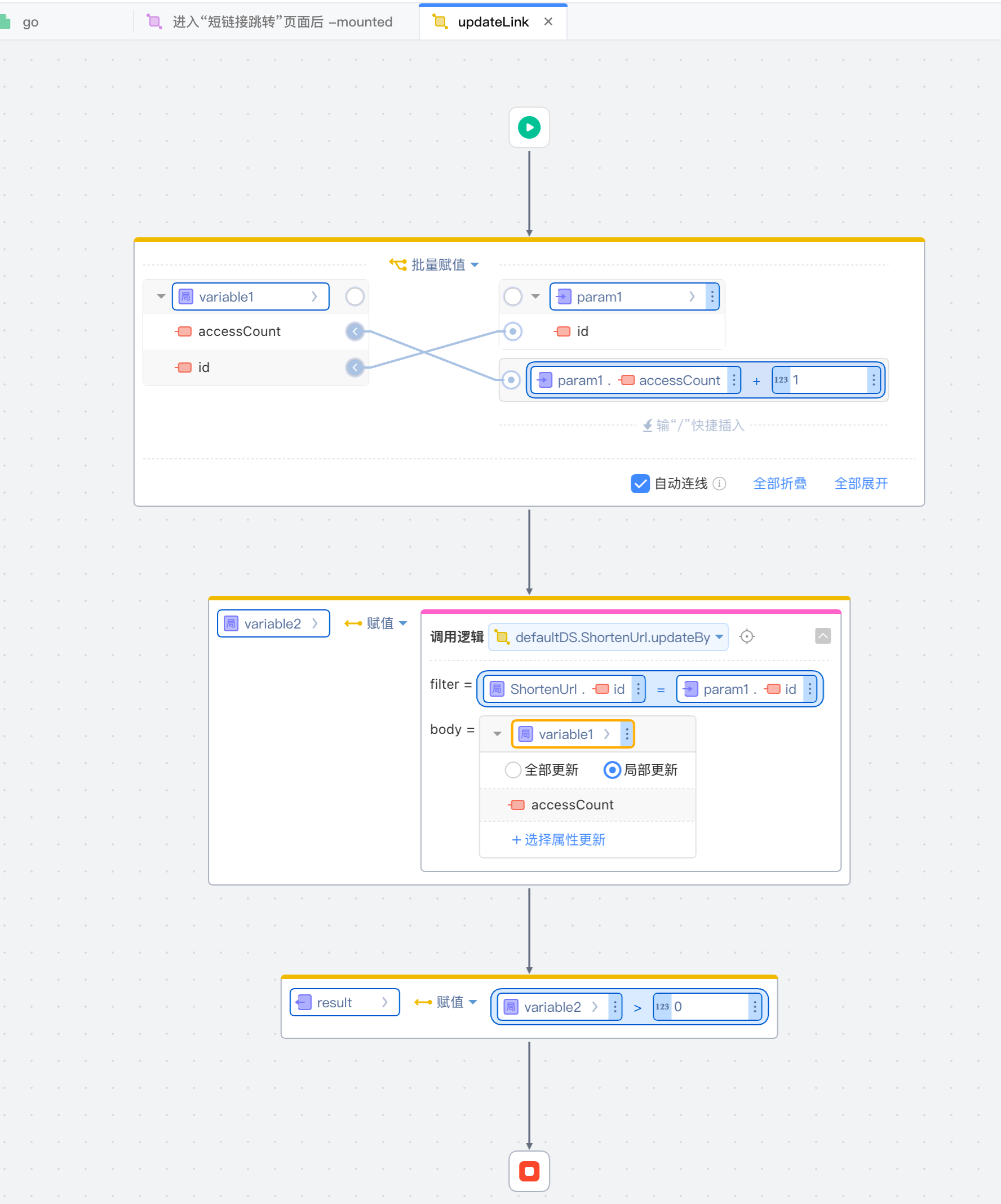
其中调用依赖库逻辑getLongUrl，传入从url传入的code值，和平台逻辑findLinkByShortCode，及updateLink

findLinkByShortCode逻辑如图：



根据shortCode查找到ShortUrl对象，并转换成Link对象传出。

updateLink逻辑如图：



调用数据源的逻辑updateBy，将ShortenUrl的accessCount加1，并返回影响行数的布尔值。

另外需要一个error页面展示链接失效或异常页面。



此页面无逻辑。需要在go页面逻辑中对无效链接跳转处理。

短链接使用时，可拼装应用地址加/go?code=<shortCode>值。

如  [https://dev-testshorturl-qa.app.codewave.163.com/go?code=2X1s3X](%20https://dev-testshorturl-qa.app.codewave.163.com/go?code=2X1s3X) 即可跳转。