# 第5章 LOOKUP 功能开发

## 关键知识点

本章通过 Lookup 开发案例，理解实体之间 N：1 关联关系，理解 Lookup 的使用。



**关键知识点：**







NUI 的基本使用

N:1 关联

Lookup 的基本使用

## 案例描述

之前的示例中，增加新的报销单信息时有一个字段为受益部门 ID，需要手工输入数字， 非常不方便。而在实际应用中，使用者会浏览到所有的部门信息，然后选择一个部门，而选择的数据做为新增报销单信息的一部分，随报销单信息存到数据库中。

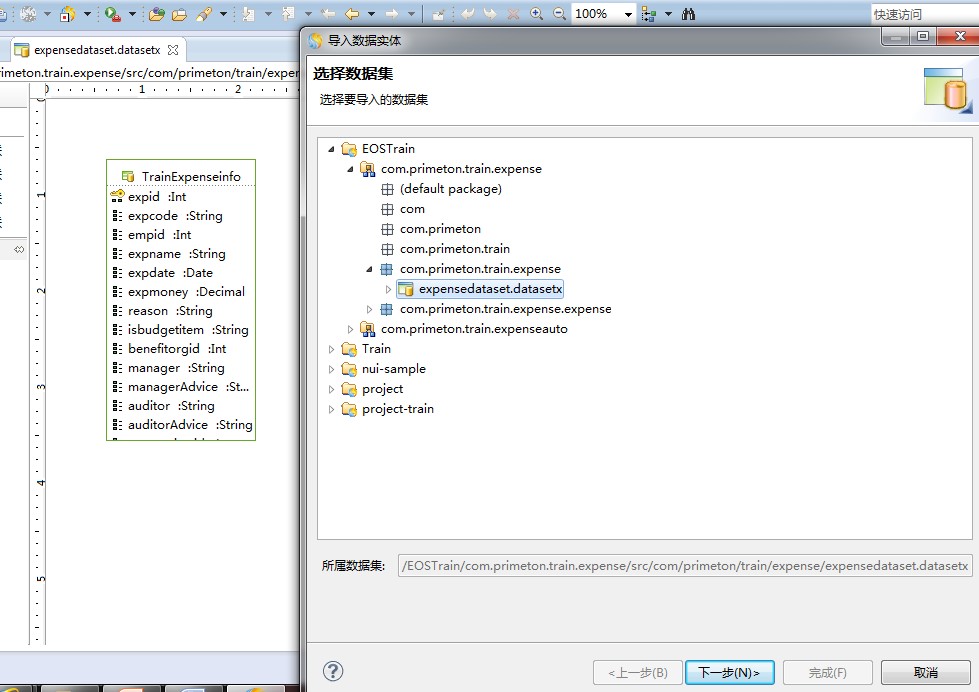
在本小节中将利用EOS 提供的 Lookup 控件功能，来实现报销单信息增加时可视地增加受益部门信息的功能。

## 案例开发

本案例不需要新建项目，直接在 EOSTrain 项目 com.primeton.train.expense 构件包的下开发。开发的详细步骤如下：

* + 1. 新建数据实体 OrgOrganization

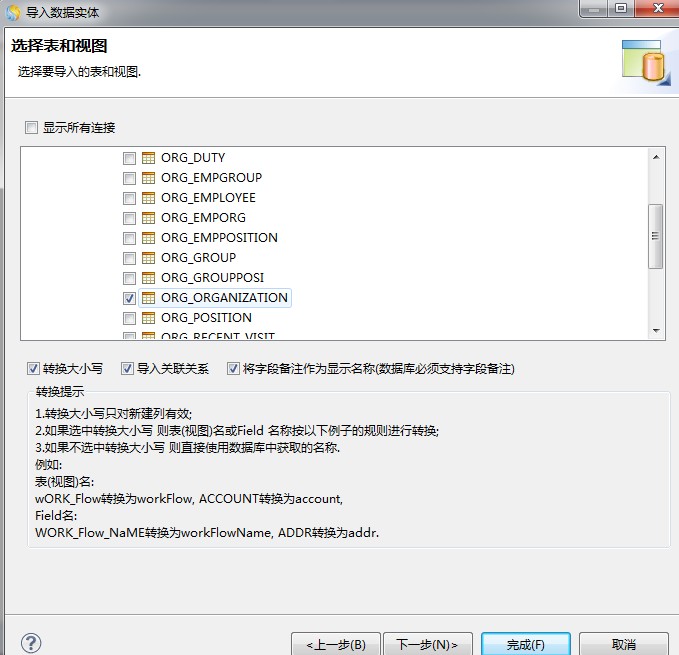
1. 本例子中受益部门借助 coframe 中组织机构的表，所以不需要单独创建机构表；
2. 打开 com.primeton.train.expense 构件包->数据–>数据模型： expensedataset.datasetx 文件，在文件上右键，点击“导入数据实体”，找到 ORG\_ORGANIZATION 表，完成导入。具体过程如下：



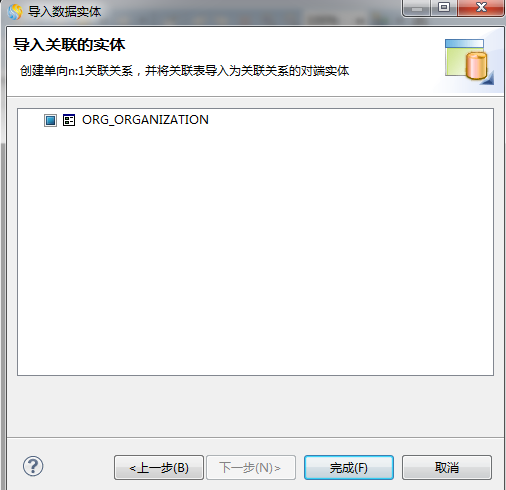
**图5-1** 导入数据实体 1



**图5-2** 导入数据实体 2

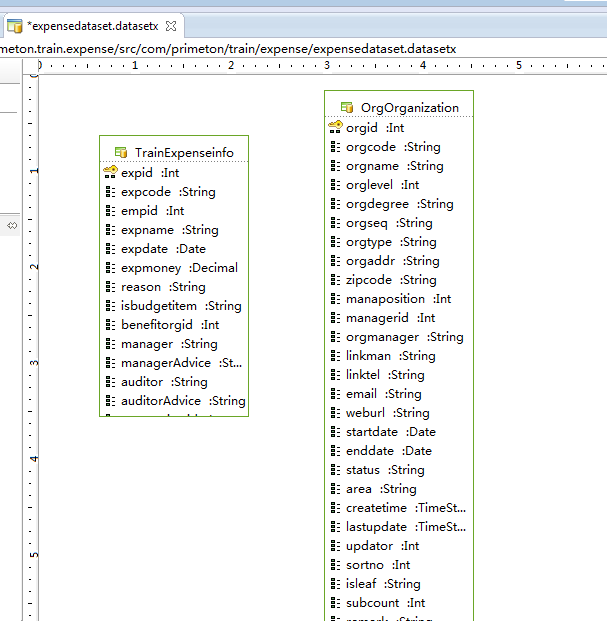


**图5-3** 导入数据实体 3



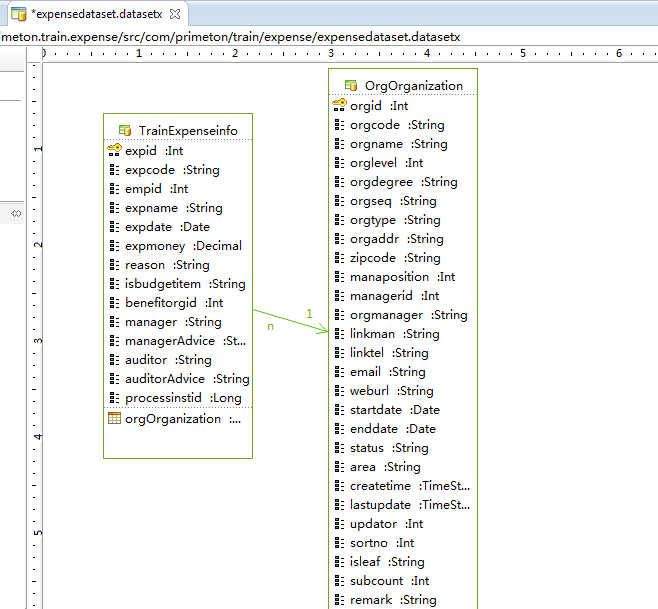
**图5-4** 导入数据实体 4

在 expensedataset.datasetx 文件中会出现 OrgOrganization 实体，如图：

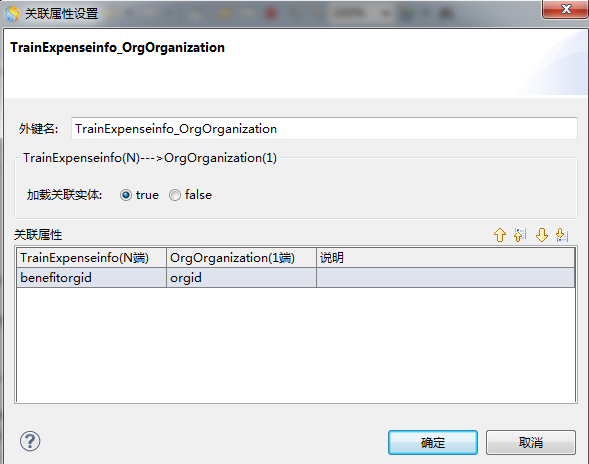


**图5-5** OrgOrganization 数据实体

1. 为两个数据实体建立单向 n:1 关系，在左侧调色板中找到“单向 n:1 关系”线，选中， 从“TrainExpenseinfo”实体向“OrgOrganization”实体拖动，会在两个实体间出现绿色单向连线。并且TrainExpenseinfo 实体中会增加一个新的属性：orgOrganization。



**图5-6** 单向 n:1 关系

双击连线，通过下拉列表选中N 端的 benefitorgid。如图；

**图5-7** 关联属性

* + 1. 借助 NUI Demo 了解 Lookup 用法

用浏览器打开 NUI Demo 首页：http://127.0.0.1:8080/default/demo/index.jsp，在左侧菜单里找到：控件列表–>表单控件 -> Lookup，右侧展开“code”,可以看到 Lookup 由3 部分组成：“nui-lookup”类型的input 控件、“nui-panel”类型的div 控件及“nui-datagrid”类型的 div 控件。

1. “nui-lookup”类型的 input 控件，代码如下：

<input id=*"lookup2"* name=*"look"* class=*"nui-lookup"* style="width:*200px*;" textField=*"name"*valueField=*"id"*popupWidth=*"auto"*

popup=*"#gridPanel"*grid=*"#datagrid1"*multiSelect=*"true"* value=*"5e9d1625-d14e-49f2-b618-efcaadeeca71"*text=*"abc"* />

其中：name 和value 组成提交时该控件的“名”与“值”；textField 与 valueField 为 下拉列表选中后传回到主页面的“显示值”与“真实值”；popup 的值为“#”加上“nui-panel”类型的 div 控件的 id 值;grid 的值为“#”加上“nui-datagrid”类型的 div 控件的 id。

1. “nui-panel”类型的 div 控件

<div id=*"gridPanel"* class=*"nui-panel"* title=*"header"* iconCls=*"icon-add"* style="width: *450px*; height: *250px*;"

showToolbar=*"true"* showCloseButton=*"true"* showHeader=*"false"* bodyStyle=*"padding:0"* borderStyle=*"border:0"*>

<div property=*"toolbar"* style="padding: *5px*; padding-left: *8px*; text-align:

*center*;">

<div style="float: *left*; padding-bottom: *2px*;">

<span>姓名：</span><input id=*"keyText"* class=*"nui-textbox"* style="width: *160px*;" onenter=*"onSearchClick"* /><a

class=*"nui-button"* onclick="onSearchClick">查询</a><a class=*"nui-button"* onclick="onClearClick">清除</a>

</div>

<div style="float: *right*; padding-bottom: *2px*;">

<a class=*"nui-button"* onclick="onCloseClick">关闭</a>

</div>

<div style="clear: *both*;"></div>

</div>

以上代码实现了用“姓名”进行搜索得到结果功能的页面部分。

1. “nui-datagrid”类型的 div 控件，这部分代码含在“nui-panel”类型的 div 控件内：

<div id=*"gridPanel"* class=*"nui-panel"* title=*"header"* iconCls=*"icon-add"* style="width: *450px*; height: *250px*;"

showToolbar=*"true"* showCloseButton=*"true"* showHeader=*"false"* bodyStyle=*"padding:0"* borderStyle=*"border:0"*>

<div property=*"toolbar"* style="padding: *5px*; padding-left: *8px*; text-align:

*center*;">

<div style="float: *left*; padding-bottom: *2px*;">

<span>姓名：</span><input id=*"keyText"* class=*"nui-textbox"*

style="width: *160px*;" onenter=*"onSearchClick"* /><a

class=*"nui-button"* onclick="onSearchClick">查询</a><a class=*"nui-button"* onclick="onClearClick">清除</a>

</div>

<div style="float: *right*; padding-bottom: *2px*;">

<a class=*"nui-button"* onclick="onCloseClick">关闭</a>

</div>

<div style="clear: *both*;"></div>

</div>

<div id=*"datagrid1"* class=*"nui-datagrid"* style="width: *100%*; height:

*100%*;" borderStyle=*"border:0"*

showPageSize=*"false"* showPageIndex=*"false"*

url=*"org.gocom.components.nui.demo.newdataset.impl.TEmployee.queryEmployee.bi z.ext"*>

<div property=*"columns"*>

<div type=*"checkcolumn"*></div>

<div field=*"loginname"* width=*"120"* headerAlign=*"center"* allowSort=*"true"*>员工帐号</div>

<div field=*"name"* width=*"120"* headerAlign=*"center"*

allowSort=*"true"*>姓名</div>

<div field=*"createtime"* width=*"100"* headerAlign=*"center"* dateFormat=*"yyyy-MM-dd"* allowSort=*"true"*>创建日期</div>

</div>

</div>

</div>

这部分代码实现了结果展示的页面。

* + 1. 页面修改

1. 打开 com.primeton.train.expense 构件包->展现–>页面资源：expenseForm.jsp 文件， 此页面实现报销单信息增加和修改功能。找到页面中显示 benefitorgid 的输入框：

<input class="nui-textbox" name="trainexpenseinfo.benefitorgid"/>

去掉这个 input ，将 demo 页面中的第一部分“nui-lookup”类型的 input 控件代码拷

贝到这个位置。并进行对应修改：

<input id="lookup2" name="trainexpenseinfo.orgOrganization.orgid"

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| class="nui-lookup" style="width: 200px;" | | | | |
| textField=" | orgname" | valueField=" | orgid | " popupWidth="auto" |
| popup="#gridPanel" grid="#datagrid1" multiSelect="true" /> | | | | |

将 demo 页面中的“nui-panel”类型的 div 控件代码拷贝到<script>标签的上面。并进行对应修改：

<div id=*"gridPanel"* class=*"nui-panel"* title=*"header"* iconCls=*"icon-add"* style="width:*450px*;height:*250px*;"

showToolbar=*"true"* showCloseButton=*"true"* showHeader=*"false"* bodyStyle=*"padding:0"*borderStyle=*"border:0"*>

<div property=*"toolbar"* style="padding:*5px*;padding-left:*8px*;text-align:*center*;">

<div style="float:*left*;padding-bottom:*2px*;">

<span>部门名称：</span>

<input id=*"keyText"* class=*"nui-textbox"* style="width:*160px*;" onenter=*"onSearchClick"*/>

<a class=*"nui-button"* onclick="onSearchClick">查询</a>

<a class=*"nui-button"* onclick="onClearClick">清除</a>

</div>

<div style="float:*right*;padding-bottom:*2px*;">

<a class=*"nui-button"* onclick="onCloseClick">关闭</a>

</div>

<div style="clear:*both*;"></div>

</div>

<div id=*"datagrid1"* class=*"nui-datagrid"* style="width:*100%*;height:*100%*;" borderStyle=*"border:0"* showPageSize=*"false"*

showPageIndex=*"false"* dataField=*"orgs"* url=*" com.primeton.train.expense.expense.queryOrg.biz.ext"*>

<div property=*"columns"*>

<div type=*"checkcolumn"*></div>

<div field=*"orgcode"* width=*"120"* headerAlign=*"center"* allowSort=*"true"*>机构编码</div>

<div field=*"orgname"* width=*"120"* headerAlign=*"center"* allowSort=*"true"*>机构名称</div>

<div field=*"parentorgid"* width=*"120"* headerAlign=*"center"* allowSort=*"true"*>

上级机构编号</div>

</div>

</div>

</div>

1. 将 demo 中<script>标签内的代码拷贝到页面中的<script>中:

**var** grid = nui.get("datagrid1");

**var** keyText = nui.get("keyText"); grid.load();

**function** onSearchClick(e) { grid.load({

orgname : keyText.value

});

}

**function** onCloseClick(e) {

**var** lookup2 = nui.get("lookup2"); lookup2.hidePopup();

}

**function** onClearClick(e) {

**var** lookup2 = nui.get("lookup2"); lookup2.deselectAll();

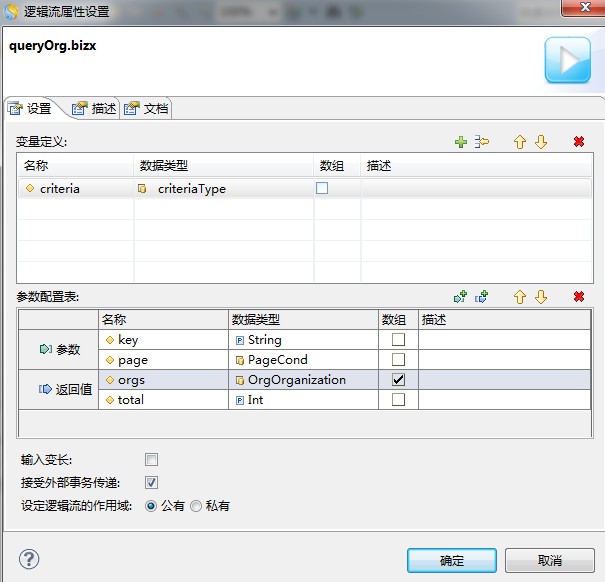
}

* + 1. 逻辑流开发

1. 在 com.primeton.train.expense 构件包->构件-> expense.componentType 右键，新建逻辑流 queryOrg.bizx。并拖入 2 个赋值图元，1 个“queryEntitiesByCriteriaEntityWithPage”图元。

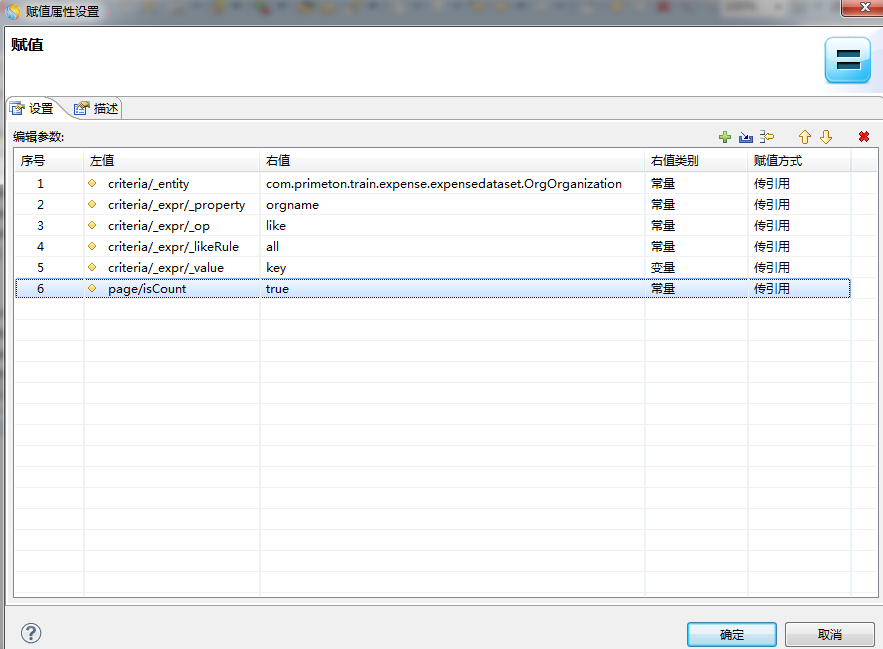


**图5-8** 关联属性

1. 双击逻辑流空白处，设置逻辑流相关参数：

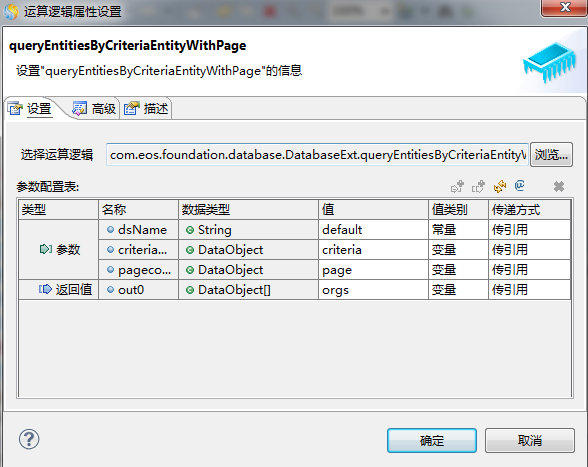
**图5-9** 逻辑流属性

1. 双击“赋值”，设置如下：



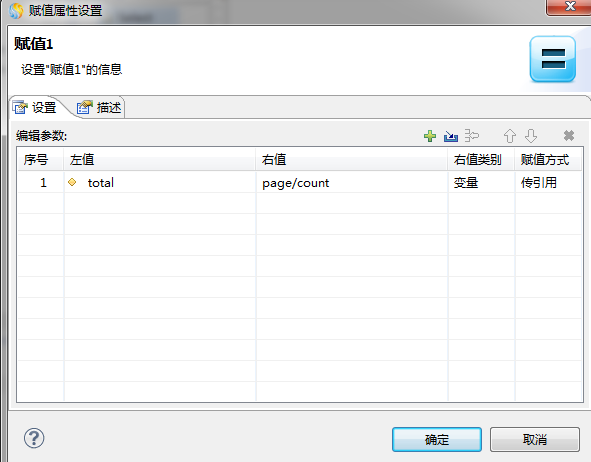
**图5-10** 赋值属性

1. 双击“queryEntitiesByCriteriaEntityWithPage”，设置如下：



**图5-11** queryEntitiesByCriteriaEntityWithPage 属性

1. 双击“赋值 1”，设置如下：



**图5-12** 赋值 1 属性

1. 完善页面

修改 JSP 文件中的 setFormData() 方法：

**function** setFormData(data) {

//跨页面传递的数据对象，克隆后才可以安全使用

**var** infos = nui.clone(data);

//保存 list 页面传递过来的页面类型：add 表示新增、edit 表示编辑

nui.getbyName("pageType").setValue(infos.pageType);

//如果是点击编辑类型页面

**if** (infos.pageType == "edit") {

**var** json = infos.record;

**var** form = **new** nui.Form("#form1");//将普通 form 转为 nui 的

form

form.setData(json); nui.get(); form.setChanged(**true**);

nui.get("lookup2").setText(json.trainexpenseinfo.orgOrganization.orgname);

}

}

1. 完善页面expenseList.jsp

找到显示机构 ID 的部分：

<div field=*"benifitorgid"* headerAlign=*"center"* allowSort=*"true"*>

受益部门

</div>

将上面代码改为：

<div field=*"orgOrganization.orgname"* headerAlign=*"center"* allowSort=*"true"*>

受益部门

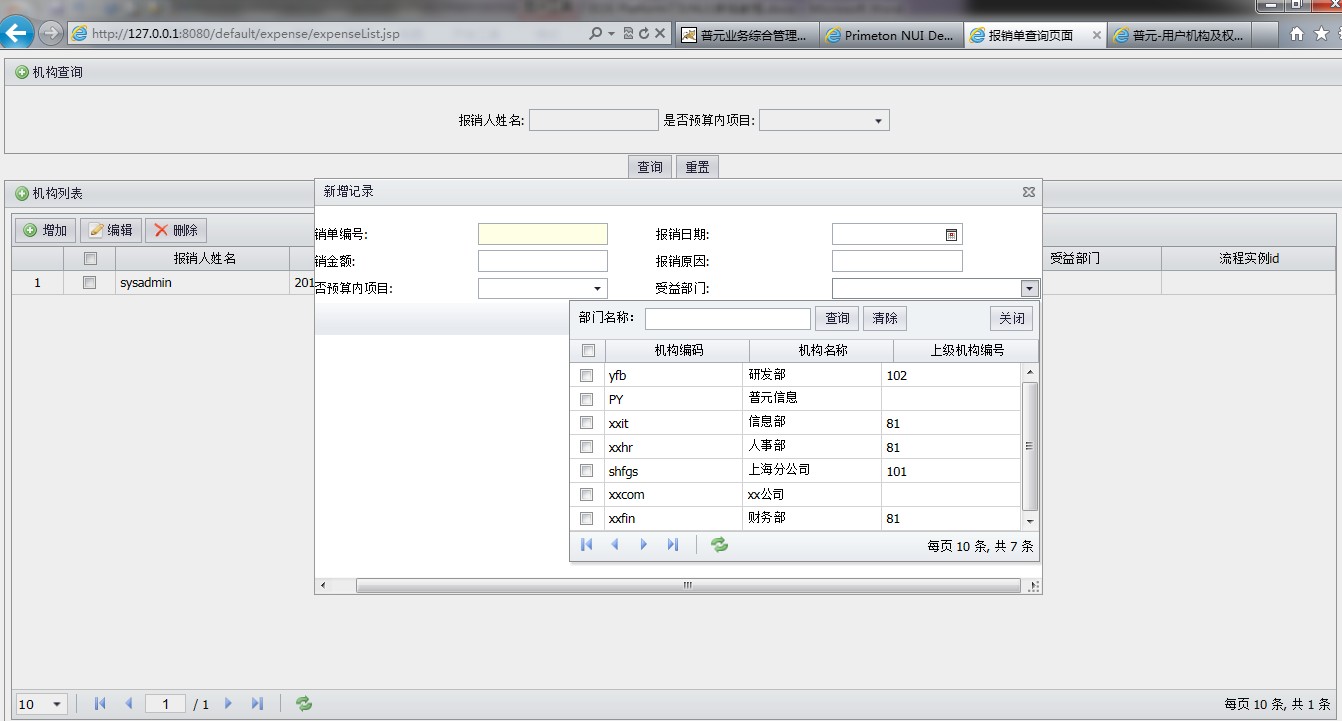
</div>

注释 add() 和 edit () 中的长宽设置：

//width: 600, height: 300

开发完成。

## 测试运行

1. 访问 http://127.0.0.1:8080/default/expense/expenseList.jsp，点击“增加”按钮，在打开的增加页面点击机构对应的输入框，出现 lookup 页面。

**图5-13** 测试页面 1

1. 访问 http://127.0.0.1:8080/default/expense/expenseList.jsp，选中一条记录，点击“修改”按钮，在打开的修改页面 lookup 输入框内显示出当前的机构信息，点击机构对应的输入框后方按键，出现 lookup 页面，进行修改机构信息。



**图5-14** 测试页面 2

**知识点解读：**

**对于 N：1 关联关系，有如下几点注意事项：**

* **N 端实体中用于关联的属性可以是主键也可以是非主键，1 端的关联字段必须是主键(可以是单主键也可以是复合主键)。**
* **当在 N 端选择了用于关联的属性，那么这些属性在 N 端实体中是不存在的，且不可访问， 会用 1 个 1 端实体来代替。当要访问 N 端这些属性的值的时候，需要操作 1 端实体的主键属性才可以。（通过 Xpath 获取关联属性）**
* **单向 N:1 关联是支持延迟加载的。如果设置加载关联实体为 true，表示不支持延迟加载，那么会在查询 N 端实体的时候同时查询 1 端实体，查询时是用一句 join sql 查询；如果设置加载关联实体为 false，表示支持延迟加载，则先是查出所有的 N 端实体，然后根据 N 端的关联属性去查询 1 端实体。**