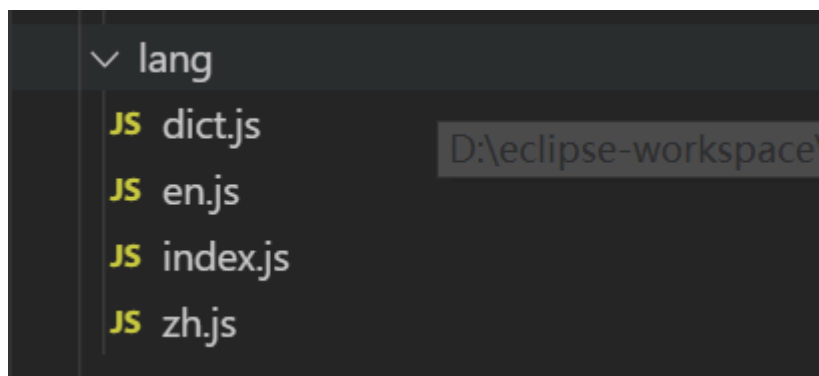


1 国际化使用

1.1 资源维护



在 src/lang 文件夹中存放着国际化相关的文件

en.js 英文

zh.js 中文

dict.js veeValidate 验证框架的国际化资源（中英文）

index.js vue-i18n 的初始化实例

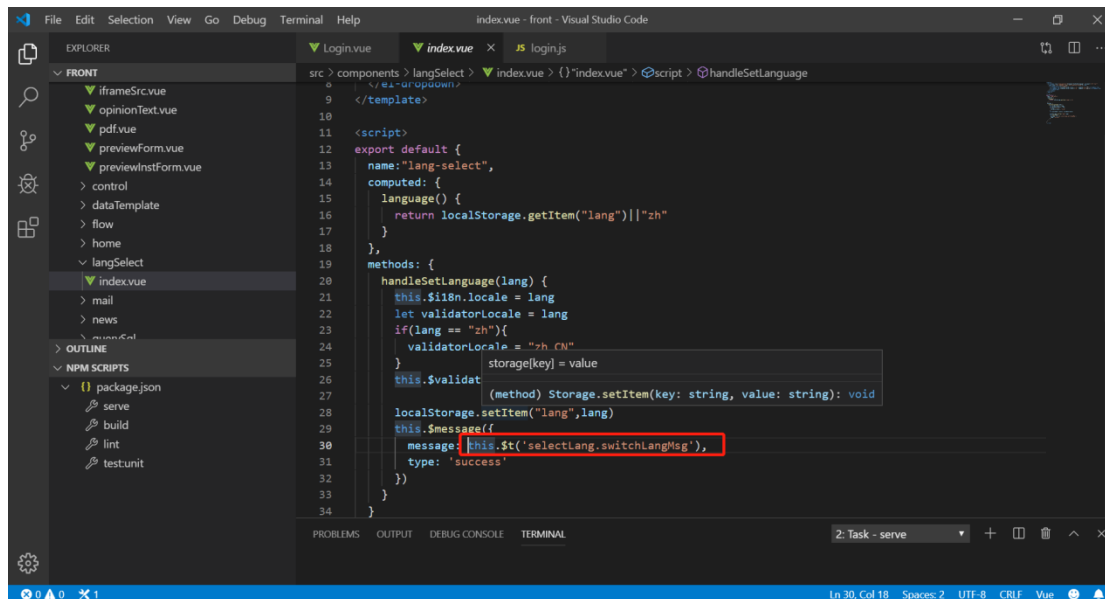
1.2 国际化资源使用示例

1.2.1 Dom 中 使用

```
{{${t('login.welcome2Login')}}}
```

1.2.2 Script 中的使用

```
this.$t('selectLang.switchLangMsg')
```



1.2.3 Js 中使用

```
导入 import i18n from '@/lang/index.js';

console.log(i18n.t("login.login"));
```

1.2.4 表单验证提示信息使用国际化

可以参考 approvalItemAdd.vue 页面

```
<el-form-item label="常用语:" prop="expression" :class="{ 'is-error': errors.has('expression') }">
  <el-input v-model="form.expression" name="expression" v-validate="'required'"></el-input>
  <span v-if="errors.has('expression')" class="el-input__suffix"><span class="el-input__suffix-inner"><!--></span><i
class="el-input__icon el-input__validateIcon el-icon-circle-close"></i></span>
  <div class="el-form-item__error" v-if="errors.has('expression')">{{errors.first('expression')}}</div>
</el-form-item>
```

```
this.$validator.validateAll().then((result)=>{
  if(result){
    this.form.type = 4;
    this.$store
```

```

        .dispatch("approvalItem/approvalItemSave", this.form)
        .then(res => {
            this.$router.go(-1);
        });
    }else {
        return false;
    }
});

```

1.3 Vue-i18n 安装

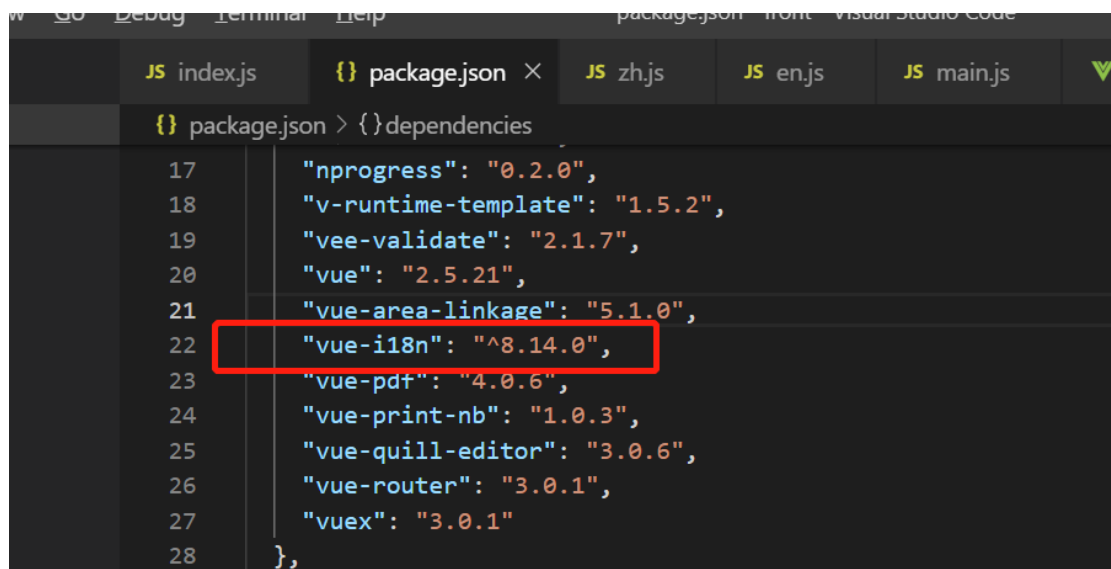
1.3.1 安装 vue-i18n

cnpm install --save vue-i18n

```

PS D:\eclipse-workspace\shenjiaosuo_dev2\shenjiaosuo\web\src\main\webapp\front> cnpm install --save vue-i18n
√ Installed 1 packages
√ Linked 0 latest versions
√ Run 0 scripts
√ All packages installed (1 packages installed from npm registry, used 736ms(network 732ms), speed 17.44kB/s, jsd
tarball 0B)
PS D:\eclipse-workspace\shenjiaosuo_dev2\shenjiaosuo\web\src\main\webapp\front> cnpm install

```



1.3.2 cnpm run build 编译

打包完成后会生成 dist 的文件夹，里面存的是编译文件

```
PS D:\eclipse-workspace\shenjiaosuo_dev2\shenjiaosuo\web\src\main\webapp\front> cnpm run build

> dh@0.1.0 build D:\eclipse-workspace\shenjiaosuo_dev2\shenjiaosuo\web\src\main\webapp\front
> vue-cli-service build

- Building for production...
```

1.3.3 启动服务命令： cnpm run serve

```
PS D:\eclipse-workspace\shenjiaosuo_dev2\shenjiaosuo\web\src\main\webapp\front> cnpm run serve

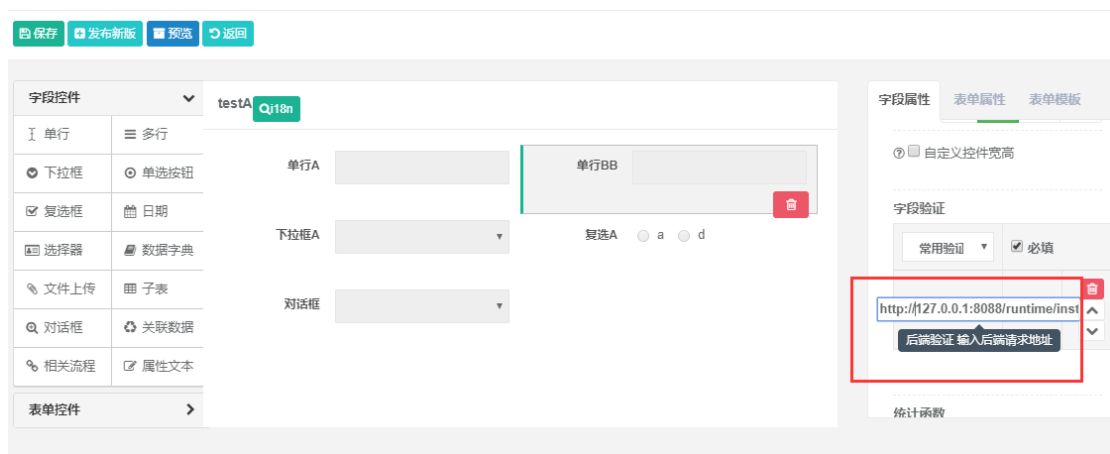
> dh@0.1.0 serve D:\eclipse-workspace\shenjiaosuo_dev2\shenjiaosuo\web\src\main\webapp\front
> vue-cli-service serve

INFO Starting development server...
```

2 后端校验

可以参考 InstanceController.java 类

[http://127.0.0.1:8088/runtime/instance/v1/isBackendValidate?param=](http://127.0.0.1:8088/runtime/instance/v1/isBackendValidate?param=param)
param 为控件输入控件的值 提交给后台接收
[http://127.0.0.1:8088/runtime/instance/v1/isBackendValidate?param=\\${form}](http://127.0.0.1:8088/runtime/instance/v1/isBackendValidate?param=${form})



param= 参数为控件的输入值

2.1 后端接口返回值 true 或者 false

```
@RequestMapping(value="isBackendValidate",method=RequestMethod.POST, produces = { "application/json; charset=utf-8" })
@ApiOperation(value = "后端验证", httpMethod = "POST", notes = "后端验证")
public boolean isBackendValidate(@ApiParam(name="param", required = false)@RequestParam String param) throws Exception {
    if(StringUtil.isEmpty(param) && "test".equals(param)) {
        return true;
    }else {
        return false;
    }
}
```

2.2 后端接口返回自定义消息

➤ 校验成功

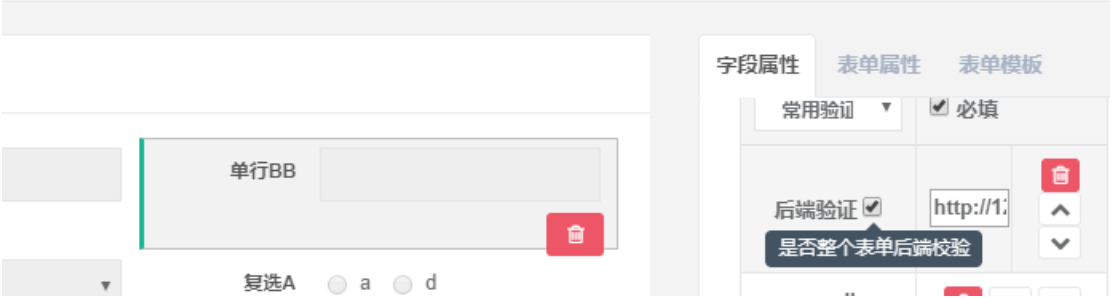
```
{
    "valid": true
}
```

➤ 校验失败

```
{
    "valid": false,
    "data": {
        "message": "已存在."
    }
}
```

```
1732 @RequestMapping(value="isBackendValidateReturnObject",method=RequestMethod.POST, produces = { "application/json; charset=utf-8" })
1733 @ApiOperation(value = "后端验证", httpMethod = "POST", notes = "后端验证")
1734 public Object isBackendValidateReturnObject(@ApiParam(name="param", required = false)@RequestParam String param) throws Exception {
1735     @RequestBody ObjectNode formData) throws Exception {
1736         System.out.println(formData);
1737         ObjectNode createObjectNode = JsonUtil.getMapper().createObjectNode();
1738         if(StringUtil.isEmpty(param) && "test".equals(param)) {
1739             createObjectNode.put("valid", true);
1740             return createObjectNode;
1741         }else {
1742             createObjectNode.put("valid", false);
1743             ObjectNode data = JsonUtil.getMapper().createObjectNode();
1744             data.put("message", "已经存在");
1745             createObjectNode.set("data",data );
1746             return createObjectNode;
1747         }
1748     }
}
```

2.3 是否整个表单后端校验

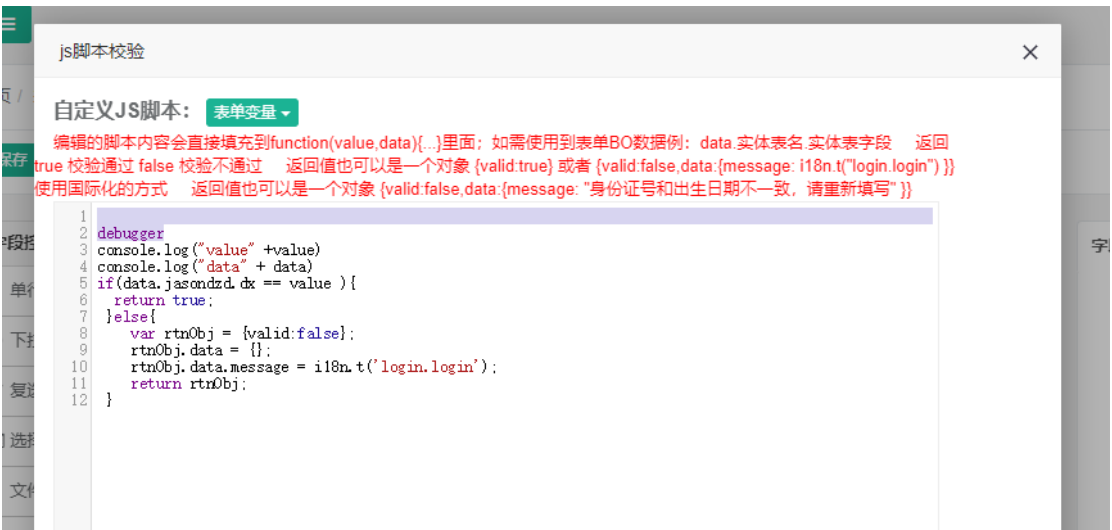


后台方法 formData 是表单数据

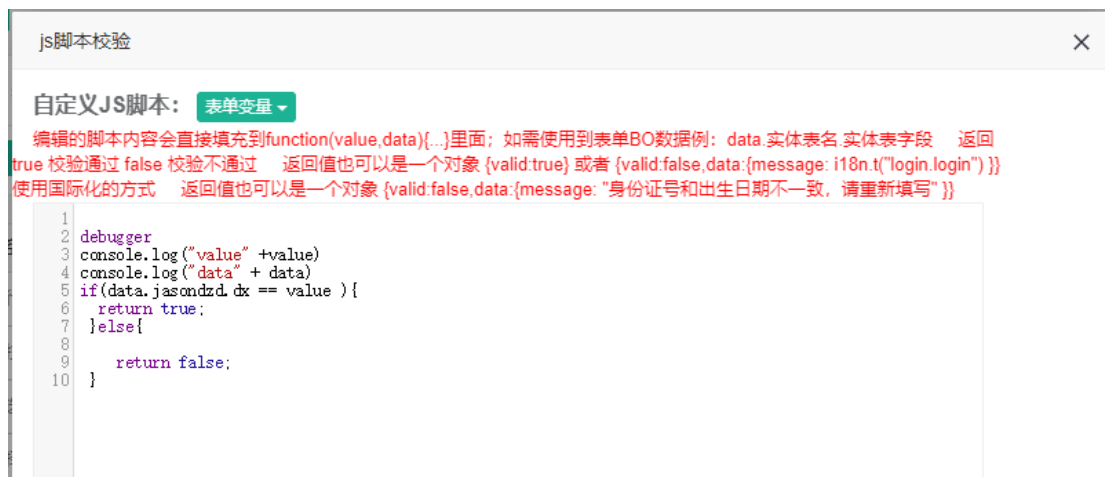
```
1732 @RequestMapping(value="isBackendValidateReturnObject",method=RequestMethod.POST, produces = { "application
1733 @ApiOperation(value = "后端验证", httpMethod = "POST", notes = "后端验证")
1734 public Object isBackendValidateReturnObject(@ApiParam(name="param", required = false)@RequestParam String
1735 @RequestBody ObjectNode formData) throws Exception {
1736     System.out.println(formData);
1737     ObjectNode createObjectNode = JsonUtil.getMapper().createObjectNode();
1738     if(StringUtil.isEmpty(param) && "test".equals(param)) {
1739         createObjectNode.put("valid", true);
1740         return createObjectNode;
1741     }else {
1742         createObjectNode.put("valid", false);
1743         ObjectNode data = JsonUtil.getMapper().createObjectNode();
1744         data.put("message", "已经存在");
1745         createObjectNode.set("data",data );
1746         return createObjectNode;
1747     }
1748 }
```

3 前端 js 校验

3.1 Demo1 返回国际化后的提示信息



3.2 Demo2 使用默认的提示信息 只要返回 true 或者 false 即可



4 多表头操作方式

4.1 先保存表单内容 （子表使用的模板是 拖拽表单子表模板）

字段属性

表单属性

表单模板

宏模板

PC端宏模板

表单模板

多列表格主模板

子实体模板

拖拽表单子表模板

4.1.1 预览效果

审批内容

testMainSub

姓名:

请输入内容

年龄:

请输入内容

学生

姓名	年龄	住址	语文	数学	英语
请输入内容	请输入内容	请输入内容	请输入内容	请输入内容	请输入内容

当前表单的表头是没有多表头的 需要自己配置表头的代码

4.2 在引入脚本功能中拷贝表头

保存 发布新版 预览 返回

字段控件

单行 多行

下拉框 单选按钮

复选框 日期

选择器 数据字典

文件上传 子表

对话框 关联数据

相关流程 属性文本

表单控件

testMainSub

Q18n

姓名 年龄

学生

姓名 年龄

住址 语文

数学 英语

字段属性 表单属性 表单模板

表单名称 testMainSub

表单标识 testMainSub

引入脚本

布局 单列 两列 三列 四列

超级管理员

业务

系统

流程

用户

系统

引入脚本

```
37 <tr @click="transitionIndex = -1" v-if="!permission.table.xs.hidden">
38 <th class="group-th" colspan="4" sub-relation="onetone">
39 学生
40 </th>
41 </tr>
42 <tr v-if="!permission.table.xs.hidden">
43 <td colspan="4">
44 <table class="form-table" cellspacing="0" cellpadding="0" border="0">
45 <thead>
46 <tr class="sub-table-header" @click="transitionIndex = -1">
47 <th v-if="!permission.fields.xs.xn!='n'" style="font-
48 weight:normal;"><span v-if="!permission.fields.xs.xn!='b'">*</span>
49 姓名
50 </th>
51 <th v-if="!permission.fields.xs.ni!='n'" style="font-
52 weight:normal;"><span v-if="!permission.fields.xs.ni!='b'">*</span>
53 年龄
54 </th>
55 <th v-if="!permission.fields.xs.zs!='n'" style="font-
56 weight:normal;"><span v-if="!permission.fields.xs.zs!='b'">*</span>
57 住址
58 </th>
59 <th v-if="!permission.fields.xs.yw!='n'" style="font-
60 weight:normal;"><span v-if="!permission.fields.xs.yw!='b'">*</span>
61 语文
62 </th>
63 <th v-if="!permission.fields.xs.sx!='n'" style="font-
64 weight:normal;"><span v-if="!permission.fields.xs.sx!='b'">*</span>
65 数学
66 </th>
67 <th v-if="!permission.fields.xs.yy!='n'" style="font-
68 weight:normal;"><span v-if="!permission.fields.xs.yy!='b'">*</span>
69 英语
70 </th>
71 </tr>
72 </thead>
73 </table>
74 </td>
75 </tr>
76 </tbody>
77 </table>
78 <tr v-for="(item, index) in data.zbbtbb.sub_xs" key="index" :data-
79 index="index" :class="['transition', index==transitionIndex]" :data-
80 subname="data.zbbtbb.sub_xs">
```

自定义脚本: 常用变量

编辑的脚本内容会直接填充到【script】里面; 如需使用到表单BO数据例: data 实体表名 实体表字段

1

4

4.2.1 需要将thead 标签元素中的内容复制出来改造

4.3 表头配置示例



4.4 预览效果



```
<tr class="sub-table-header" @click="transitionIndex = -1">

    <th colspan="3" v-if="permission.fields.xs.xml!='n'"
    style="font-weight:normal;";"><span
    if="permission.fields.xs.xml=='b'">*</span>
    学生信息
    </th>
    <th colspan="3" v-if="permission.fields.xs.nl!='n'"
    style="font-weight:normal;";"><span
    v-
```

```
if="permission.fields.xs.nl=='b'">*</span>
    成绩信息
</th>

</tr>

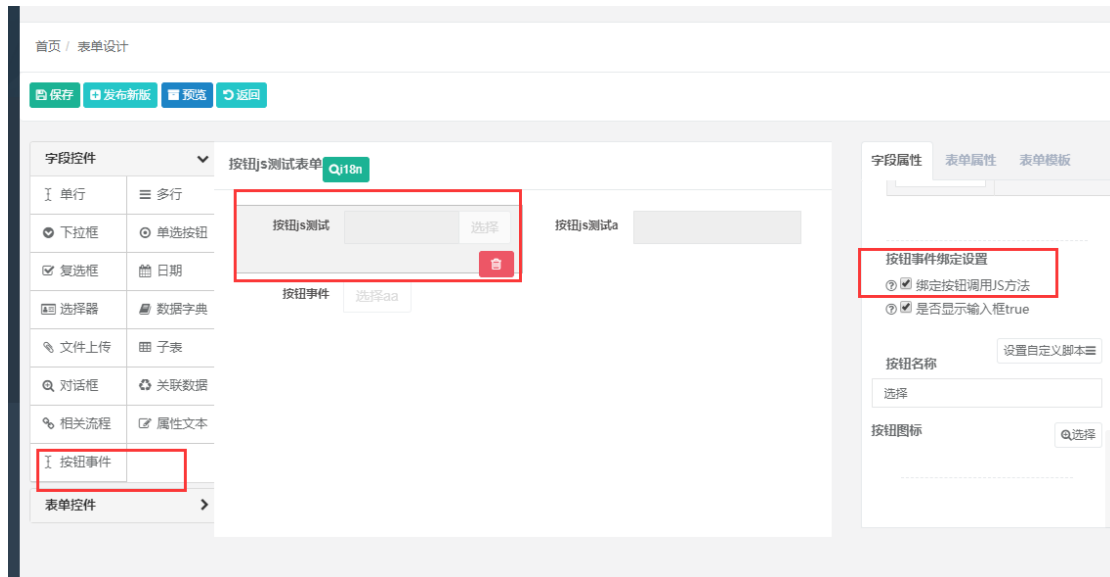
<tr class="sub-table-header" @click="transitionIndex = -1">

    <th    v-if="permission.fields.xs.xm!='n'"    style="font-
weight:normal;;;"><span    v-if="permission.fields.xs.xm=='b'">*</span>
        姓名
    </th>
    <th    v-if="permission.fields.xs.nl!='n'"    style="font-
weight:normal;;;"><span    v-if="permission.fields.xs.nl=='b'">*</span>
        年龄
    </th>
    <th    v-if="permission.fields.xs.zz!='n'"    style="font-
weight:normal;;;"><span    v-if="permission.fields.xs.zz=='b'">*</span>
        住址
    </th>
    <th    v-if="permission.fields.xs.yw!='n'"    style="font-
weight:normal;;;"><span    v-if="permission.fields.xs.yw=='b'">*</span>
        语文
    </th>
    <th    v-if="permission.fields.xs.sx!='n'"    style="font-
weight:normal;;;"><span    v-if="permission.fields.xs.sx=='b'">*</span>
        数学
    </th>
    <th    v-if="permission.fields.xs.yy!='n'"    style="font-
weight:normal;;;"><span    v-if="permission.fields.xs.yy=='b'">*</span>
        英语
    </th>

</tr>
```

5 自定义按钮 js

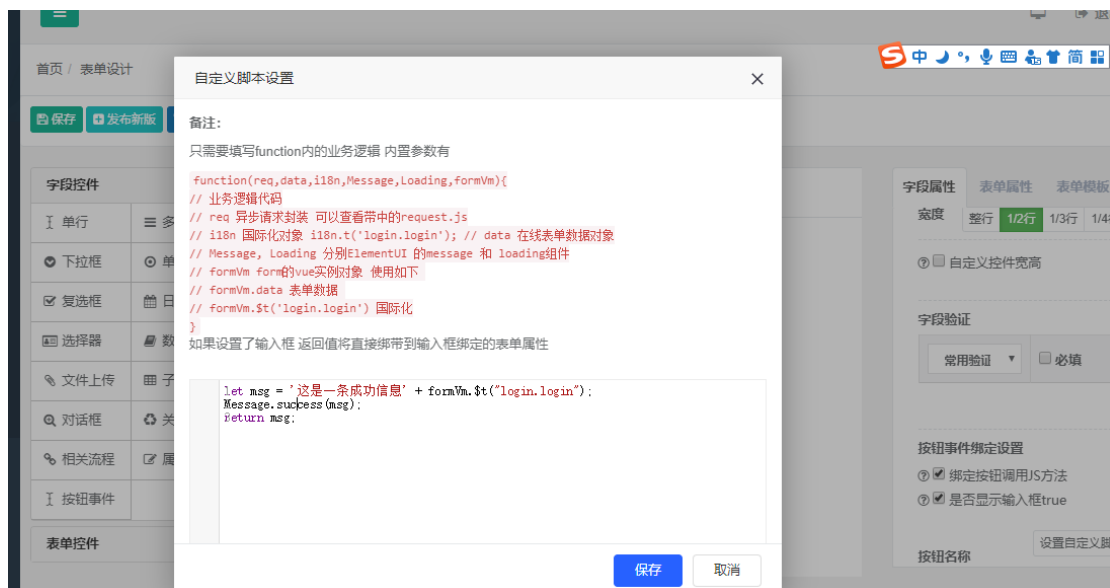
5.1 页面配置



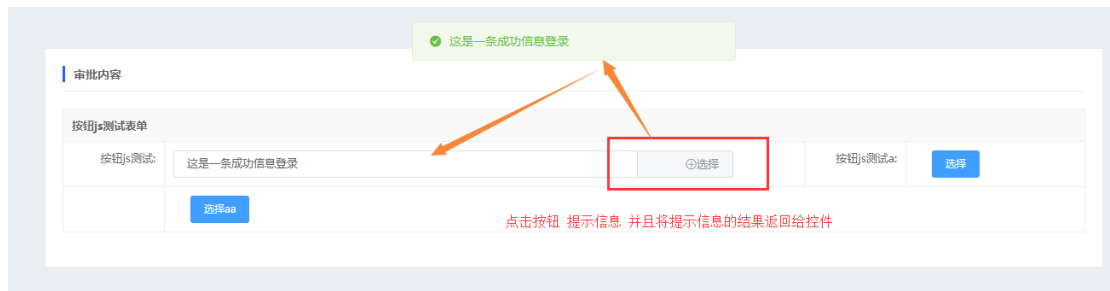
可以选择按钮事件控件类型

也可以是单行文本，但是需要勾选 绑定按钮调用 JS 方法 会自动切换为按钮事件控件类型

5.2 js 编写



5.3 预览效果

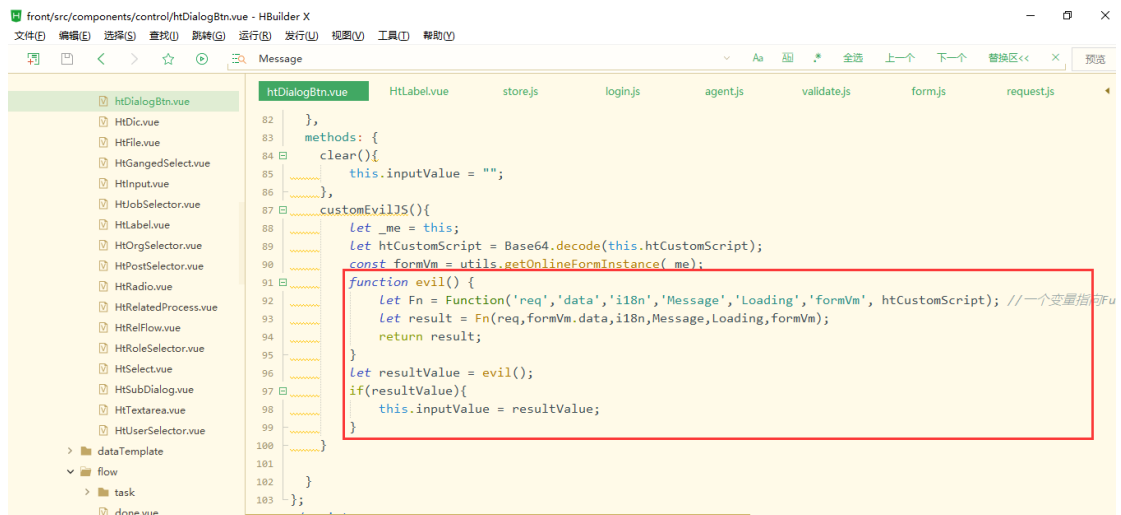


5.4 自定义按钮 js 扩展

可以进行自定义扩展， 修改 front 文件夹

\src\components\control\htDialogBtn.vue 文件

目前内置的参数有 req data i18n Message Loading formVm 可以根据情况扩展



6 数字控件

6.1 添加模板

给 `$scope.templates` 添加
`digitalControl` 数字控件模板

```
Java EE - shenjiaosuo/web/src/main/webapp/manage/js/formDesign/formDesignController.js - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

formDesignCo... formDesign.html customScript... formDesignLi... customValid.js FreeMakerUtil... DelegatingMe... dlbgzmbwbk.ftl x2

632 + </div>
633 + <div ng-if="editingField===child" class="tooltip-container">
634 + <a class="btn btn-danger fa fa-trash-o" title="移除" ng-click="removeField">
635 + </div>
636 + </div>
637 + </div>
638 + <div class="sub-footer" style="height:30px;" ng-click="selectField(field)">
639 + </div>
640 + </div>
641 'customQuery': '<select class="form-control"></select>',
642 'relFlow': '<object svg-data="../../css/svg/flowList.svg" type="image/svg+xml" style="width:{{field
643 'text': '<font ng-bind-html="field.textValue"></font>',
644 'textFixed': '<font style="font-weight:{{field.textFixedBoldText?\'bold\':\'normal\'}};font-size:{{
645 'tabs': '<div><div class="xh-hr"><hr><label ng-bind="field.desc"></label></div></div>',
646 'tabCheck': '<div><div class="xh-hr"><hr><label ng-bind="field.desc"></label></div></div>',
647 'collapseEnd': '<div><div class="xh-hr"><hr><label ng-bind="field.desc"></label></div></div>',
648 'collapse': '<div><div class="xh-hr"><hr><label ng-bind="field.desc"></label></div></div>',
649 'digitalControl': '<input type="number" class="form-control" >',
650 };
651
652 if(formType == 'mobile'){
653     $scope.columnTemplates = [
```

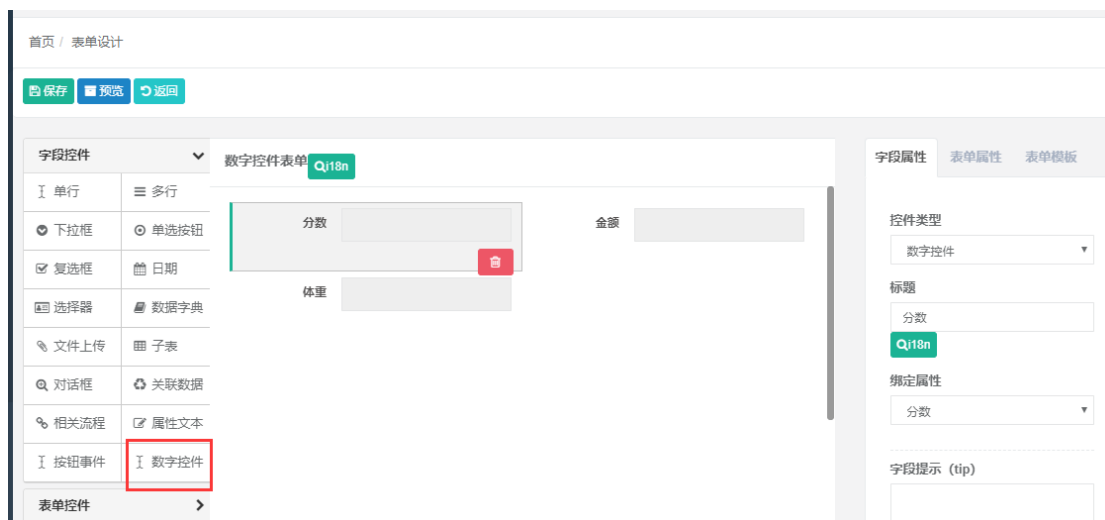
6.2 设置\$scope.fieldLib

```
Java EE - shenjiaosuo/web/src/main/webapp/manage/js/formDesign/formDesignController.js - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

*formDesignCo... formDesign.html customScript... formDesignLi... customValid.js FreeMakerUtil... DelegatingMe... dlbgzmbwbk.ftl x2

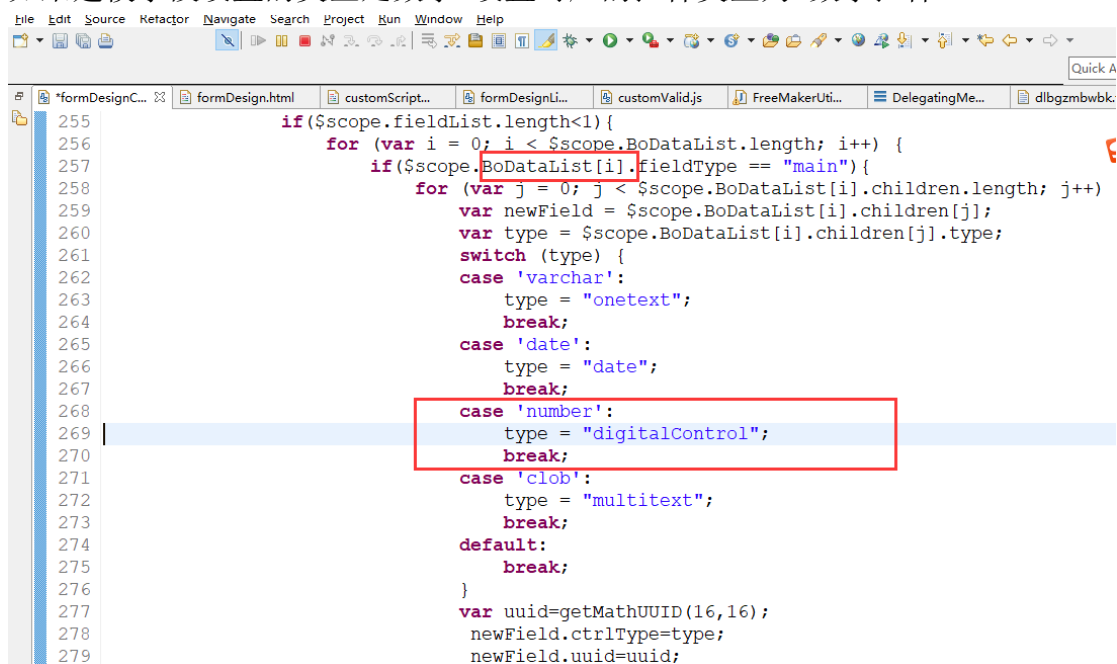
701 }
702 };
703 }else{
704     $scope.fieldLib = {
705         basic: [
706             {name:'单行', ctrlType:'onetext', icon: 'i-cursor'},
707             {name:'多行', ctrlType:'multitext', icon: 'navicon'},
708             {name:'下拉框', ctrlType:'select', icon: 'chevron-circle-down'},
709             {name:'单选按钮', ctrlType:'radio', icon: 'dot-circle-o'},
710             {name:'复选框', ctrlType:'checkbox', icon: 'check-square-o'},
711             {name:'日期', ctrlType:'date', icon: 'calendar'},
712             {name:'选择器', ctrlType:'selector', icon: 'drivers-license-o'},
713             {name:'数据字典', ctrlType:'dic', icon: 'book'},
714             {name:'文件上传', ctrlType:'fileupload', icon: 'paperclip'},
715             {name:'子表', ctrlType:'sub', icon: 'table', notSupportSub: true},
716             {name:'对话框', ctrlType:'dialog', icon: 'search-plus'},
717             {name:'关联数据', ctrlType:'customQuery', icon: 'recycle'},
718             {name:'相关流程', ctrlType:'relFlow', icon: 'chain'},
719             {name:'属性文本', ctrlType:'text', icon: 'edit'},
720             {name:'按钮事件', ctrlType:'onetextBtn', icon: 'i-cursor'},
721             {name:'数字控件', ctrlType:'digitalControl', icon: 'i-cursor'}
722         ],
723         // flow: [
724         //     {name:'流程图', ctrlType:'htbpmflowimage', icon: 'photo', notSupportSub:
725         //     {name:'意见', ctrlType:'opinion', icon: 'commenting-o', notSupportSub: t
```

设置完可以查看页面效果



6.3 根据建模字段的类型设置控件的默认类型

如果建模字段设置的类型是数字 设置对应的控件类型为 数字控件



表单设计界面

控件类型 中添加 数字控件供用户切换控件类型

```
Java EE - sherjiaosuo/web/src/main/webapp/manage/views/formDesign/formDesign.html - Eclipse
File Edit Source Navigate Search Project Run Window Help
Quick Access Java EE Del

dlbgzmb.ftl formDesignCo... formDesign.html customScript... formDesignLi... customValidjs FreeMakerUtil... DelegatingMe... dlbgzmbwbk.ftl
98<h3 heading="字典管理" ng-show="editingField.ctrlType=='collapseEND'">
99<div class="panel-body" full-scroll ng-model="contentHeight">
100<div class="ibox-content no-top-border">
101<div name="formDef" class="form-horizontal" ng-if="editingField">
102<div class="form-group" ng-if="isHideDIVByRule('sub,opinion,htbpmflowimage,history,iframe,img,collapseEND')">
103<label>控件类型</label>
104<select class="form-control input-sm" ng-change="chooseType(this)" desc="控件类型">
105ht-validate-design="{required:true}" ng-model="editingField.ctrlType">
106<option value="onetext" ng-if="editingField.type=='number' || editingField.type=='varchar'">
107<option value="onetextBtn" ng-if="editingField.type=='varchar'">按钮事件</option>
108<option value="multitext" ng-if="editingField.type=='clob'">多行</option>
109<option value="select" ng-if="editingField.type=='varchar'">下拉框</option>
110<option value="radio" ng-if="editingField.type=='varchar'">单选按钮</option>
111<option value="checkbox" ng-if="editingField.type=='varchar'">复选框</option>
112<option value="date" ng-if="editingField.type=='date'">日期</option>
113<option value="selector" ng-if="editingField.type=='varchar'">选择器</option>
114<option value="dic" ng-if="editingField.type=='varchar'">数据字典</option>
115<option value="fileupload" ng-if="editingField.type=='varchar'">文件上传</option>
116<option value="dialog" ng-if="editingField.type=='varchar'">对话框</option>
117<option value="customQuery" ng-if="editingField.type=='varchar'">关联数据</option>
118<option value="relFlow" ng-if="editingField.type=='varchar'">相关流程</option>
119<option value="text" ng-if="editingField.type=='varchar'">属性文本</option>
120<option value="digitalControl" ng-if="editingField.type=='number'">数字控件</option>
121</select>
122</div>
123<div class="form-group" ng-if="!isShowDIVByRule('textFixed')">
```

6.4 添加数字控件类型特有的属性

7 自动完成组件

7.1 input 带输入建议

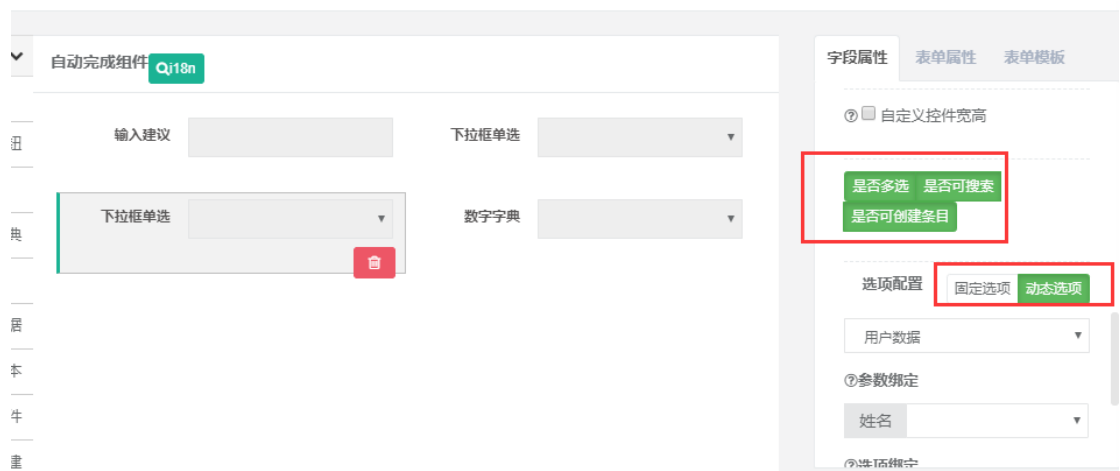
动态数据设置



也可以设置固定选项



7.2 下拉单选，下拉多选



支持固定选项和动态选项

- 可以设置是否多选 （选择多个备选项）
- 是否可搜索 （过滤选项）
- 是否可创建条目 （可通过在输入框中输入文字来创建新的条目。注意此时 filterable （是否可搜索）必须为真）

7.3 数字字典

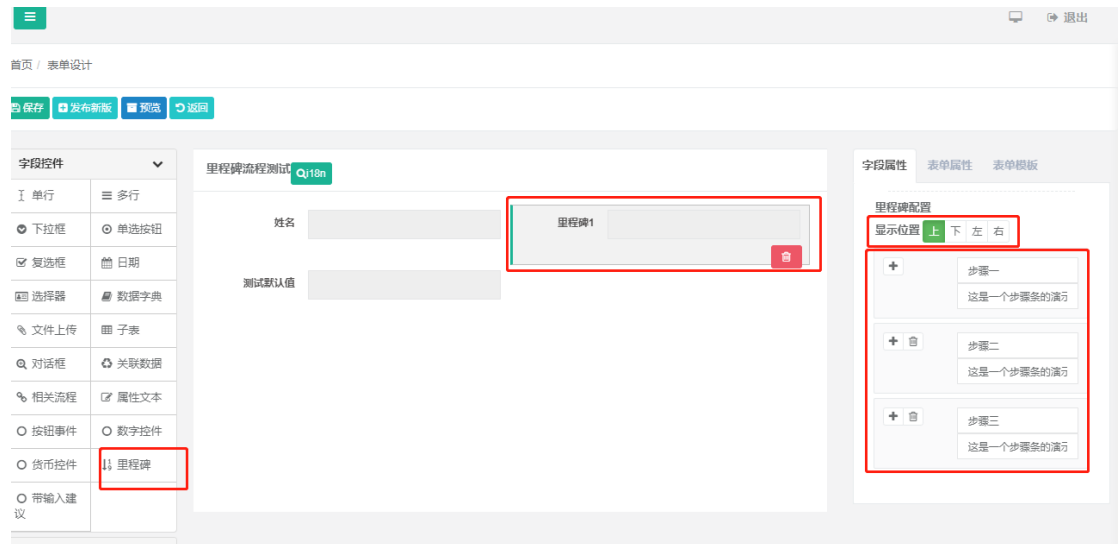


当设置字典可以搜索时，可以过滤字典项的值

8 里程碑控件

8.1 表单设置

大文本的字段才支持里程碑控件



显示位置：指的是该里程碑在表单的显示位置。
下方的各项：分别是里程碑各个节点的名称、描述。

配置完后，表单预览如下图所示：



8.2 具体代码实现：

Manage 端：

在 formDesign.html 页面添加一个里程碑控件下拉选择

```

<select class="form-control input-sm" ng-change="chooseType(this)" desc="控件类型"
  ht-validate-design="{required:true}" ng-model="editingField.ctrlType">
  <option value="onetext" ng-if="editingField.type=='number' || editingField.type=='varchar'">单行</option>
  <option value="onetextBtn" ng-if="editingField.type=='varchar'">按钮事件</option>
  <option value="multitext" ng-if="editingField.type=='clob'">多行</option>
  <option value="select" ng-if="editingField.type=='varchar'">下拉框</option>
  <option value="radio" ng-if="editingField.type=='varchar'">单选按钮</option>
  <option value="checkbox" ng-if="editingField.type=='varchar'">复选框</option>
  <option value="date" ng-if="editingField.type=='date'">日期</option>
  <option value="selector" ng-if="editingField.type=='varchar'">选择器</option>
  <option value="dic" ng-if="editingField.type=='varchar'">数据字典</option>
  <option value="fileupload" ng-if="editingField.type=='varchar'">文件上传</option>
  <option value="dialog" ng-if="editingField.type=='varchar'">对话框</option>
  <option value="customQuery" ng-if="editingField.type=='varchar'">关联数据</option>
  <option value="relFlow" ng-if="editingField.type=='varchar'">相关流程</option>
  <option value="text" ng-if="editingField.type=='varchar'">属性文本</option>
  <option value="digitalControl" ng-if="editingField.type=='number'">数字控件</option>
  <option value="currencyControl" ng-if="editingField.type=='number'">货币控件</option>
  <option value="realtimeInputCtrl">实时单行</option>
  <option value="realtimeMultitext">实时多行</option>
  <option value="onetextautocomplete" ng-if="editingField.type=='varchar'">带输入建议</option>
  <option value="stepControl" ng-if="editingField.type=='clob'">里程碑</option>
</select>
</div>

```

formDesignController.js 里添加一项

```

}else{
  $scope.fieldLib = {
    basic: [
      {name:'单行', ctrlType:'onetext', icon: 'i-cursor'},
      {name:'多行', ctrlType:'multitext', icon: 'navicon'},
      {name:'下拉框', ctrlType:'select', icon: 'chevron-circle-down'},
      {name:'单选按钮', ctrlType:'radio', icon: 'dot-circle-o'},
      {name:'复选框', ctrlType:'checkbox', icon: 'check-square-o'},
      {name:'日期', ctrlType:'date', icon: 'calendar'},
      {name:'选择器', ctrlType:'selector', icon: 'drivers-license-o'},
      {name:'数据字典', ctrlType:'dic', icon: 'book'},
      {name:'文件上传', ctrlType:'fileupload', icon: 'paperclip'},
      {name:'子表', ctrlType:'sub', icon: 'table', notSupportSub: true},
      {name:'对话框', ctrlType:'dialog', icon: 'search-plus'},
      {name:'关联数据', ctrlType:'customQuery', icon: 'recycle'},
      {name:'相关流程', ctrlType:'relFlow', icon: 'chain'},
      {name:'属性文本', ctrlType:'text', icon: 'edit'},
      {name:'按钮事件', ctrlType:'onetextBtn', icon: 'circle-o'},
      {name:'数字控件', ctrlType:'digitalControl', icon: 'circle-o'},
      {name:'货币控件', ctrlType:'currencyControl', icon: 'circle-o'},
      {name:'里程碑', ctrlType:'stepControl', icon: 'sort-numeric-asc'},
      {name:'带输入建议', ctrlType:'onetextautocomplete', icon: 'circle-o'}
    ],
    // flow: [

```

Freemarker 模板修改

```

<#if hasStepControl>
  <#assign stepPosition = util.getAttrText(stepAttr,"stepPosition")>
  <#if (stepPosition=='left' || stepPosition=='right')>
    <div style="display:flex;flex-direction:row;">
  <#else>
    <div style="display:flex;flex-direction:column;">
  </#if>
  <#if stepPosition=='left'>
    <ht-step direction='vertical' v-model="${getNgModel(stepAttr,isSub)}" content='${util.toJsonByPath(stepAttr,'steps')}' :
  </#if>
  <#if stepPosition=='top'>
    <ht-step direction='horizontal' v-model="${getNgModel(stepAttr,isSub)}" content='${util.toJsonByPath(stepAttr,'steps')}'
  </#if>
</#if>

```

前端添加 vue 组件: HtStep.vue

```
previewForm.vue OnlineForm.vue HtStep.vue HtSelect.vue HtLabel.vue HtFile.vue

1 <template>
2   <div class="inputs">
3     <el-steps :active="value" :direction="direction" align-center finish-status="success">
4       <el-step v-for="item in steplist" :title="item.title" :description="item.description"></el-step>
5     </el-steps>
6   </div>
7 </template>
8 <script>
9   import utils from "@/utils.js";
10  export default {
11    name: "ht-step",
12    props: [
13      'value',
14      'direction',
15      'content'
16    ],
17    data() {
18      return {
19        active:1,
20        steplist:[]
21      };
22    },
23    mounted() {},
24    created() {
25      var contentjson = JSON.parse(this.content);
26      this.steplist = contentjson;
27      if(!this.value){
28        //默认值
29        this.value = 1;
30      }
31    }
32  }
33 </script>
```

组件中的三个属性分别是：

value: 用途是当前里程碑（步骤条）正处于哪个节点，高亮该节点。值来源于流程设置-BO 字段设置-该字段的设置，如 `pq.get("data").put("lichengbei",2)` 表示当前第二个节点高亮。默认值为：1

direction: 表示里程碑的方向，取值有两个：vertical/horizontal，横或竖。如果表单配置左或右，则竖直显示。如果表单配置了上下，则横向显示。

content: 该值来自于里程碑控件中的各个节点信息。

8.3 如何改变里程碑当前的节点



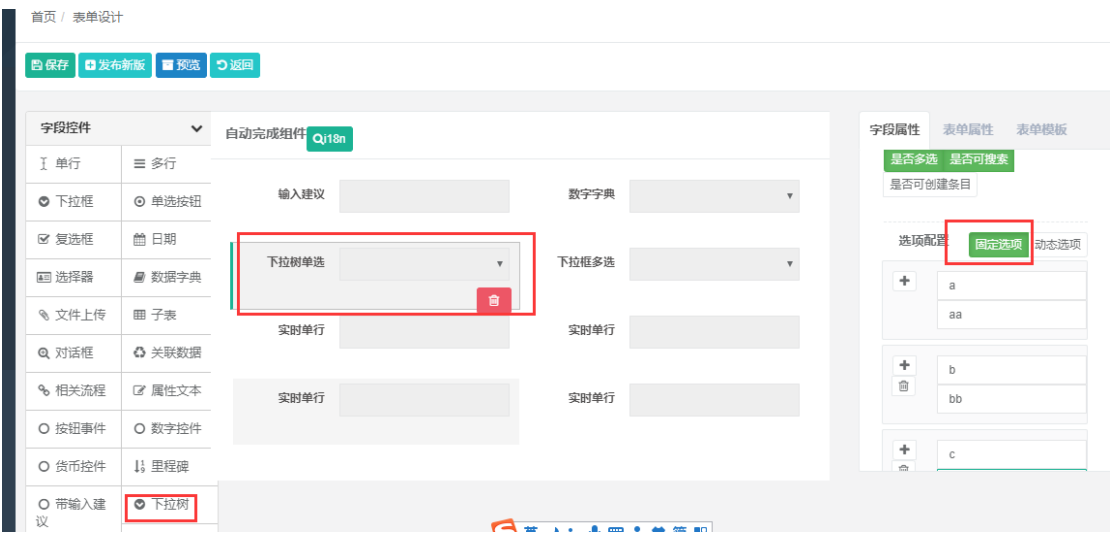
在流程的节点配置 BO，该字段的值，1 表示里程碑正处于第一个节点（高亮），

2 表示里程碑正处于第二个节点（高亮），以此类推.....

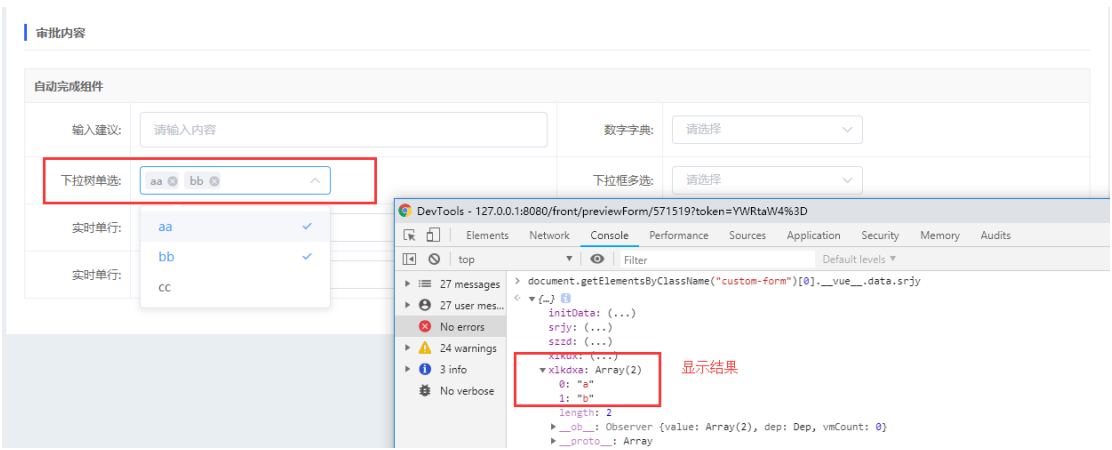
9 下拉树控件使用方法

9.1 使用固定选型（仅仅支持平铺结构）

配置页面

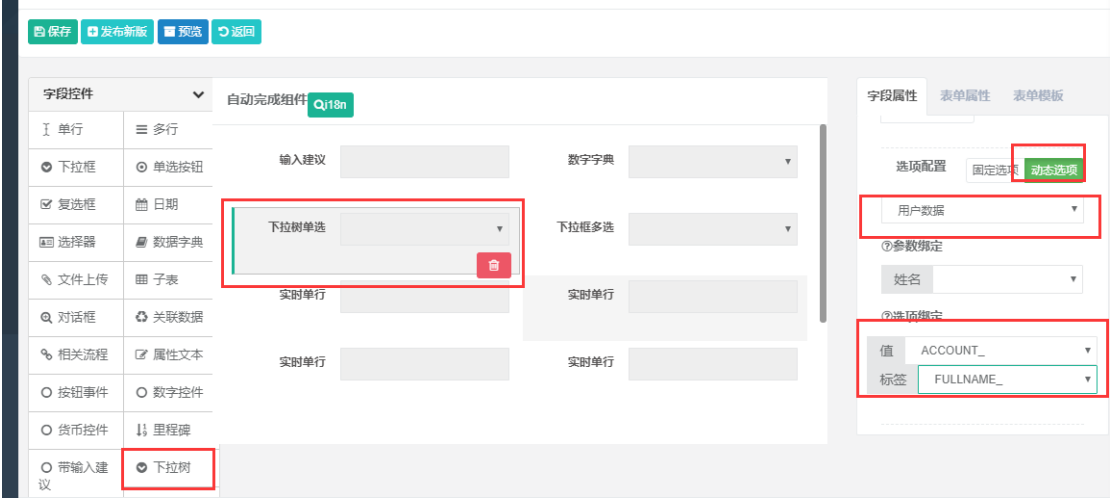


预览页面

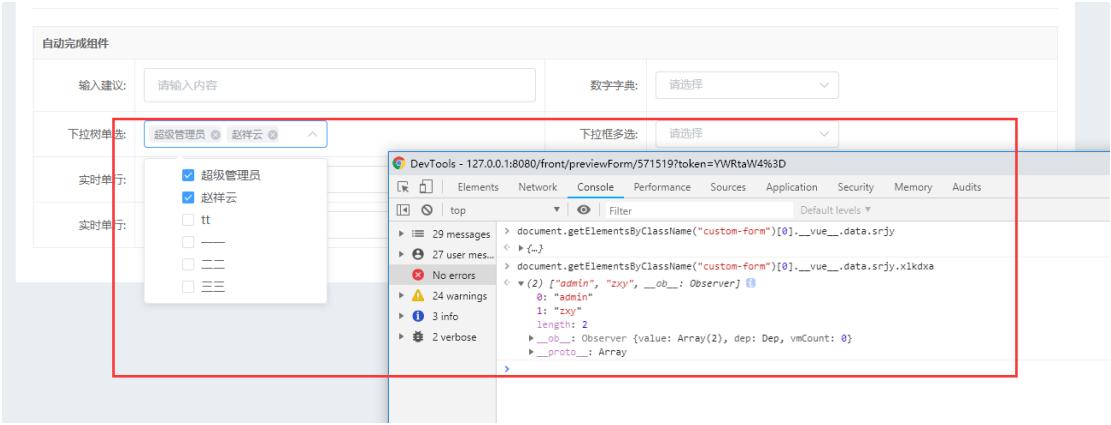


9.2 使用系统中的关联数据管理

9.2.1 数据源查询方式



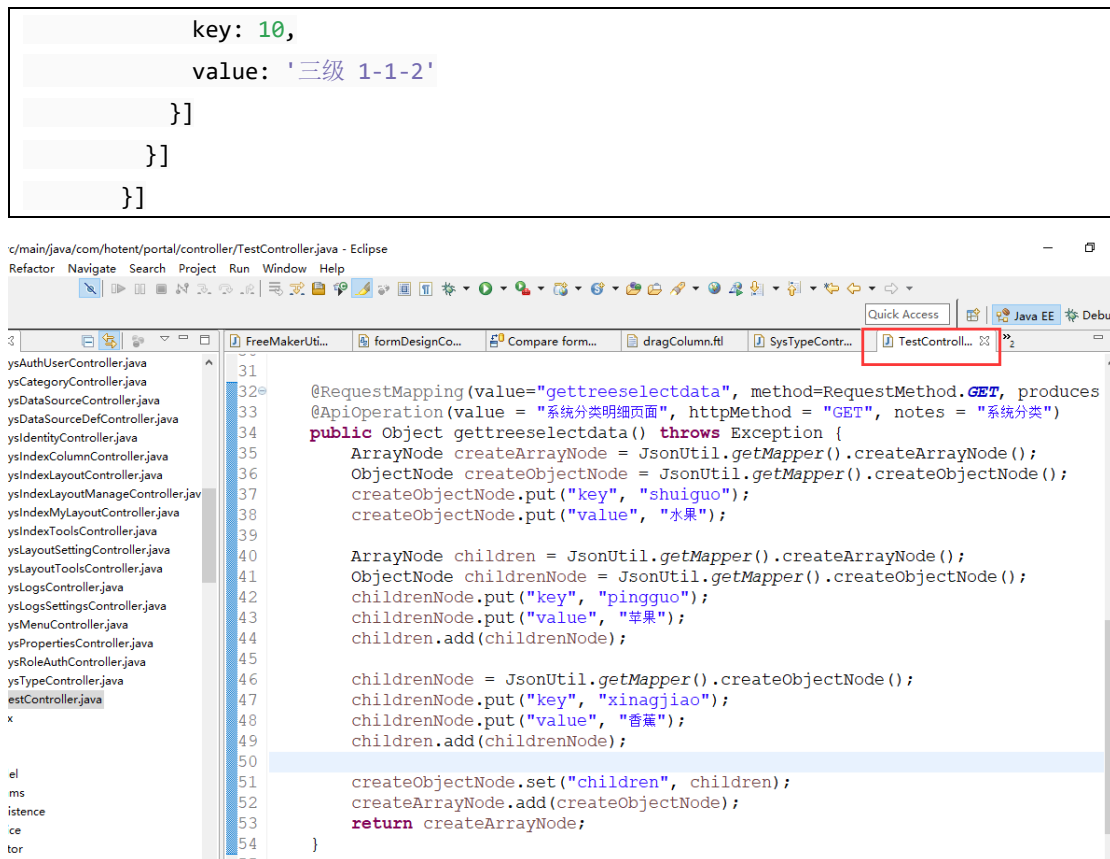
查询系统用户数据 将 accout 和 fullname 进行绑定



9.2.2 restful 接口方式

结构返回结果 示例接口可以参考 TestController.java gettreeselectdata() 方法

```
[{
  key: 1,
  value: '一级 1',
  children: [{
    key: 4,
    value: '二级 1-1',
    children: [{
      key: 9,
      value: '三级 1-1-1'
    }, {
```

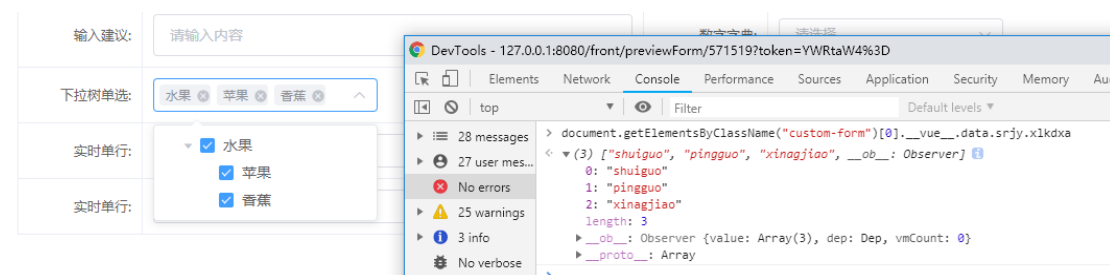


接口添加方法

首页 / 关联数据管理编辑

名称:	下拉树测试	别名:	xlscs
分页:	☐ 是 ☒ 否		
数据来源:	☐ 数据源 ☒ REST接口		
请求地址:	\${portal}/sys/test/v1/gettreeselectdata	请求类型:	☐ POST ☒ GET
数据源:	☒ 设置列		

不用设置列



10 级联树控件

绑定值格式 {value:['1','2','3'],label:['广东','广州','海珠']}

10.1 数据源方式

10.1.1测试数据准备：

执行《\doc\项目开发文档\级联测试数据.sql》文件中的 sql

此电脑 > 软件 (D:) > workspace > sjs_wp > shenjiaosuo > doc > 项目开发文档

名称	修改日期	类型	大小
国际化使用说明VUE.docx	2019/9/4 14:13	Microsoft Word ...	394 KB
级联测试数据.sql	2019/9/25 9:45	SQL 文件	244 KB
新功能开发使用手册.docx	2019/9/25 9:47	Microsoft Word ...	3,304 KB

10.1.2添加关联数据管理

首页 / 关联数据管理编辑

名称*:

别名*:

分页: ☒ 是 ☐ 否

分页大小:

数据来源: ☒ 数据源 ☐ REST接口

选择表或视图:

对象名称: china所属数据源: LOCAL

设置列

保存

返回

10.1.2.1 设置返回值

首页保存关闭新增

获取字段信息

字段	注解
☐ Id	Id
☐ Name	Name
☐ Pid	Pid
☐ leaf	是否叶子 0 叶

条件字段

返回字段

排序字段

自定义SQL

字段名	显示名	条件	值来源	默认值
Pid	Pid	等于	参数传入	

10.1.2.2 设置自定义 sql

```
if(!"".equals(Pid)){
    return "SELECT a.Id,a.Name,a.Pid, CASE (select count(*) from china b
where b.Pid=a.Id )" +
        "WHEN 0 THEN" +
        "    true " +
        "ELSE " +
        "    false " +
        "END as \"leaf\"  FROM china a WHERE a.Pid="+Pid;
}else{
    return "SELECT a.Id,a.Name,a.Pid, CASE (select count(*) from china b
where b.Pid=a.Id )" +
        "WHEN 0 THEN" +
        "    true " +
        "ELSE " +
        "    false " +
        "END as \"leaf\"  FROM china a WHERE a.Pid=0";
}
```

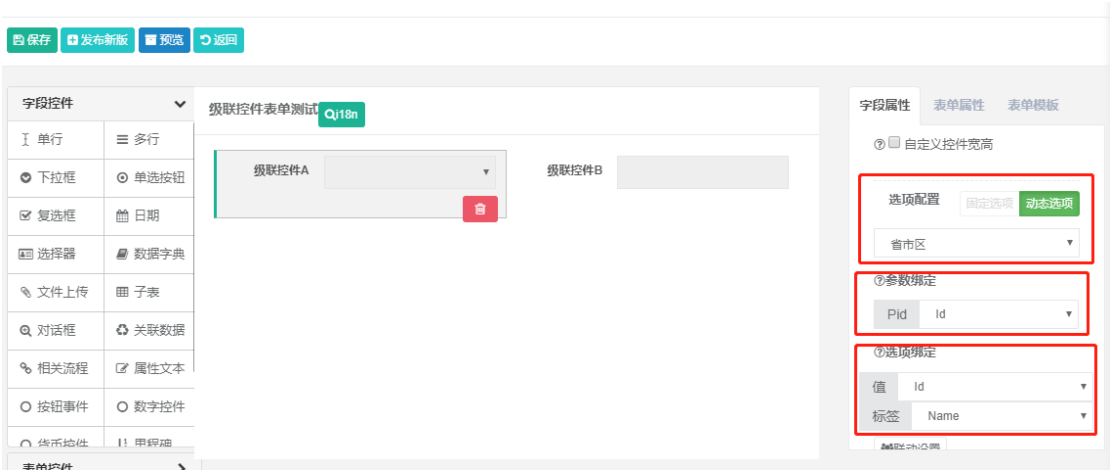
10.1.3 表单设计配置

10.1.3.1 选择控件类型为 级联选择



10.1.3.2 选项配置

选择添加好的关联数据管理，绑定参数中用数据的 id 的值去查询相同 Pid 的数据，并且设置绑定的 value 和 label 对应级联组件的 value 和 label



10.1.4 表单预览效果

级联控件表单测试

级联控件A:

北京市 / 西城区

```
{  "jlkj": {    "jlkjA": "{\\"path\\": [110000, 110100],\\"pathLabels\\": [\\"北京市\\",\\"东城区\\"]}",    "jlkjB": "    "initData": {}  }}
```

10.2 接口方式

10.2.1 添加接口管理

首页 / 关联数据管理编辑

名称:

级联数据接口测试

别名:

jlsljKCS

分页:

是

否

数据来源:

数据源

REST接口

请求地址:

\${portal}/sys/test/v1/getCascaderData

请求类型:

POST

GET

数据源:

设置列

保存

返回

保存 关闭 添加字段 新增

获取字段信息

字段	注解	操作
----	----	----

条件字段

返回字段

字段名	显示名	条件	值来源	默认值
value	value	等于	参数传入	

页

保存

关闭

添加字段

新增

获取字段信息

字段

注解

操作

条件字段

返回字段

字段名称	返回名称	管理
value	value	删除
label	label	删除
leaf	leaf	删除

10.2.2 接口

参考 portal\src\main\java\com\hotent\portal\controller\TestController.java 类

```

60 @RequestMapping(value = "getCascaderData", method = RequestMethod.GET, produces = { "application/json; charset="
61 @ApiOperation(value = "级联选择", httpMethod = "GET", notes = "级联选择")
62 public Object getCascaderData( @ApiParam(name = "value", value = "", required = false)
63 @RequestParam(required = false) String value) throws Exception {
64     ArrayNode createArrayNode = JsonUtil.getMapper().createArrayNode();
65     ObjectNode createObjectNode = JsonUtil.getMapper().createObjectNode();
66     switch (value) {
67     case "1":
68         createObjectNode.put("value", "11");
69         createObjectNode.put("label", "广州市");
70         createObjectNode.put("leaf", true);
71         createArrayNode.add(createObjectNode);
72
73         createObjectNode = JsonUtil.getMapper().createObjectNode();
74         createObjectNode.put("value", "12");
75         createObjectNode.put("label", "阳江市");
76         createObjectNode.put("leaf", true);
77         createArrayNode.add(createObjectNode);
78     case "2":
79         break;
80
81         createObjectNode.put("value", "21");
82         createObjectNode.put("label", "东城区");
83         createObjectNode.put("leaf", true);
84         createArrayNode.add(createObjectNode);
85
86         createObjectNode = JsonUtil.getMapper().createObjectNode();
87         createObjectNode.put("value", "22"+value);
88         createObjectNode.put("label", "西城区"+value);
89         createObjectNode.put("leaf", true);
90         createArrayNode.add(createObjectNode);
91     case "3":
92         break;
93
94         createObjectNode.put("value", "31"+value);
95         createObjectNode.put("label", "广东省"+value);
96         createObjectNode.put("leaf", false);
97         createArrayNode.add(createObjectNode);
98
99         createObjectNode = JsonUtil.getMapper().createObjectNode();
100        createObjectNode.put("value", "32"+value);
101        createObjectNode.put("label", "北京市"+value);
102        createObjectNode.put("leaf", false);
103        createArrayNode.add(createObjectNode);
104    }
105    return createArrayNode;
106 }

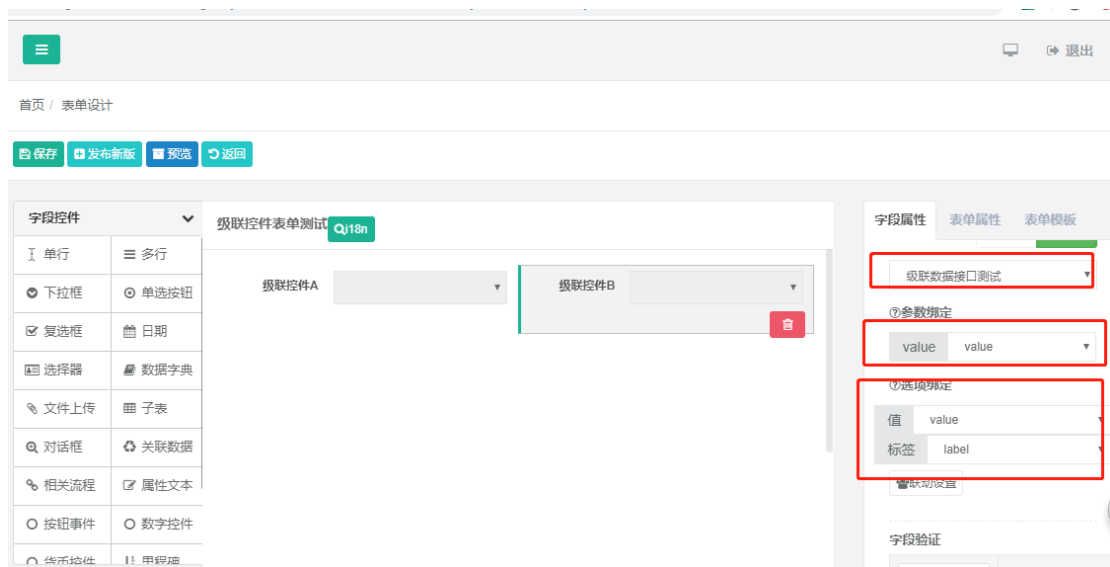
```

```
Java EE - portal/src/main/java/com/hotent/portal/controller/TestController.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

*TestControl... fieldControl... application.yml CustomQueryC... CustomQueryM... FreeMakerUtil... CustomQueryM... service

87 createObjectNode.put("label", "西城区"+value);
88 createObjectNode.put("leaf", true);
89 createArrayNode.add(createObjectNode);
90 break;
91
92 default:
93 createObjectNode.put("value", "1"+value);
94 createObjectNode.put("label", "广东省"+value);
95 createObjectNode.put("leaf", false);
96 createArrayNode.add(createObjectNode);
97
98 createObjectNode = JsonUtil.getMapper().createObjectNode();
99 createObjectNode.put("value", "2"+value);
100 createObjectNode.put("label", "北京市"+value);
101 createObjectNode.put("leaf", false);
102 createArrayNode.add(createObjectNode);
103 break;
104 }
105
106 return createArrayNode;
107 }
108 }
```

10.2.3 表单配置



选择配置好的 restful 接口

参数绑定 是节点中的哪个属性 这里使用节点的 value 属性作为参数查询下级的分类

选项绑定值 value 和 label 的绑定值设置

10.2.4已办页面预览效果



11 树组件

11.1 设置数据源 级联选择配置一样



- 请求测试地址 `${portal}/sys/test/v1/getCascaderData`
- 请求类型 GET
- 设置列



条件字段		返回字段
字段名称	返回名称	管理
value	value	删除
label	label	删除
leaf	leaf	删除

返回值 label 用于显示树节点的文字

leaf 标识该树的节点是否为叶子节点

value 值可用于参数传值，获取当前节点下的子节点信息

11.2 界面配置



11.3 设置点击节点事件执行的 js



11.4 预览效果



点击节点执行 js 代码



11.5 注意事项

```
@RequestMapping(value="getCascaderData", method=RequestMethod.GET, produc
@ApiOperation(value = "级联选择", httpMethod = "GET", notes = "级联选择")
public Object getCascaderData( @ApiParam(name = "value", value = "", requ
@RequestParam(required=false) String value) throws Exception {
    ArrayNode createArrayNode = JsonUtil.getMapper().createArrayNode();
    ObjectNode createObjectNode = JsonUtil.getMapper().createObjectNode();
    if (StringUtil.isEmpty(value)) {
        value = "0";
    }
    switch (value) {
    case "1":
        createObjectNode.put("value", "11");
        createObjectNode.put("label", "广州市");
        createObjectNode.put("leaf", true);
        createArrayNode.add(createObjectNode);

        createObjectNode = JsonUtil.getMapper().createObjectNode();
        createObjectNode.put("value", "12");
        createObjectNode.put("label", "阳江市");
        createObjectNode.put("leaf", true);
        createArrayNode.add(createObjectNode);
        break;
    case "2":
        createObjectNode.put("value", "21");
        createObjectNode.put("label", "东城区");
```

返回的节点信息中 如果需要告诉树该节点是否已经是叶子节点 则主要返回 leaf 属性为 true，不能为 0/1