## 远程数据组装下拉框

1.后台组装数据样式

[  
 {"value":"value0","key":"key0"},  
 {"value":"value1","key":"key1"},  
 {"value":"value2","key":"key2"},  
 {"value":"value3","key":"key3"},  
 {"value":"value4","key":"key4"}  
]

2.在com.yonyou.iuap.zhuzi\_test.service.OrderInfoEnumService中定义一个orderTypeMap对象.

private Map<String, String> orderTypeMap = new HashMap<String, String>();  
public List selectData(){  
 List<Map> list = new ArrayList<>();  
 for(int i=0;i<5;i++){  
 Map<String,Object> map = new HashMap<>();  
 map.put("key","key"+i);  
 map.put("value","value"+i);  
 orderTypeMap.put("value"+i,"key"+i);  
 list.add(map);  
 }  
 return list;  
 }

为了解决数据回显,需要修改afterListQuery,增加如下代码,具体可以参考demo案例

if(entity.getOrderType() != null){  
 selectData();//解决每个用户可能看到的下拉框数据不同的情况  
 String value = orderTypeMap.get(entity.getOrderType().toString());  
 entity.setOrderTypeEnumValue(value);  
}

3.在controller类中定义接口

@RequestMapping("/selectData")  
 @ResponseBody  
 public Object selectData(HttpServletRequest request){  
 List<Map> list = orderInfoEnumService.selectData();  
  
 return JSONArray.fromObject(list);  
 }

4.前端在service.js 和 model.js 中分别定义获取远程数据的方法;

service.js

export const getSelect = (params) => {  
 return request(URL.GET\_SELECT, {  
 method: "get",  
 data: params  
 });  
}

model.js

async getSelectData(param,getState){  
 let result = api.getSelect(param);  
 return result;  
 }

5.在edit.js中的state中增加一个属性,此处定义为selectDataSource,在componentWillMount方法中增加如下内容

let {data} = await actions.OrderInfo.getSelectData({  
 param:'1'  
 });  
 console.log(data);  
 this.setState({  
 selectDataSource:data  
 })

然后修改下拉框代码

<Select  
 disabled={btnFlag == 2}  
 {  
 ...getFieldProps('orderType', {  
 initialValue: typeof orderType === 'undefined' ? "" : orderType ,  
 rules: [{  
 required: false, message: '请选择订单类型',  
 }],  
 }  
 )}  
 //style={{ width: 200 }}  
 placeholder="Select a person"  
 //onChange={this.handleChange}  
 data={this.state.selectDataSource}  
/>

完成上述步骤,下拉框中的数据就可以通过远程url的方式进行获取并展示