

bip需求：费用申请单自定义字段根据供应商参照银行账户

需求：

对费用申请单的自定义字段设置成参照，根据供应商将银行账号展示出来；

修改过程：

修改路径：src\erm\erbudgetmg\erbudgetmg\card\requestApi.js

代码块

```
1      // 申请单自定义项31 (defitem31, 客商银行账号) 过滤条件, 仅供应商
2      if (item.attrcode == 'defitem44') {
3          item.queryCondition = function() {
4              var pk_cust = moduleid == presetVar.head.head1
5              ?
6              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier']
7              &&
8              props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier'].
9              value)
10             :
11             (props.cardTable.getClickRowIndex(moduleid).record.values['pk_supplier'] &&
12             props.cardTable.getClickRowIndex(moduleid).record.values['pk_supplier'].value);
13
14             var pk_currtype = moduleid == presetVar.head.head1
15             ?
16             (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_currtype']
17             &&
18             props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_currtype'].
19             value)
20             :
21             (props.cardTable.getClickRowIndex(moduleid).record.values['pk_currtype'] &&
22             props.cardTable.getClickRowIndex(moduleid).record.values['pk_currtype'].value);
23
24             return {
25                 pk_cust: pk_cust,
26                 accclass: '3',
27             }
28         }
29     }
```

```

19         pk_currtype: pk_currtype,
20         TreeRefActionExt: '',
21         DataPowerOperationCode: 'fi',
22         /**使用权组**/ isDataPowerEnable: 'Y'
23     };
24 };
25 }
26 }

```

代码学习（边做边学）

原先逻辑解读：

代码块

```

1      //客商银行账号
2      if(item.attrcode=='custaccount'){
3          item.queryCondition = ()=>{
4              let accclass = "";
5              let pk_cust = "";
6              let paytarget = (moudleid == presetVar.head.head1 ?
              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['paytarget'].value) :
              (props.cardTable.getClickRowIndex(moudleid).record.values['paytarget'].value))

7              if(paytarget=="1"){
8                  //收款对象为供应商
9                  accclass = "3";
10                 pk_cust = (moudleid == presetVar.head.head1 ?
              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['hbbm'].value)
              : (props.cardTable.getClickRowIndex(moudleid).record.values['hbbm'].value))

11             } else if(paytarget=="2"){
12                 //收款对象为客户
13                 accclass = "2";
14                 pk_cust = (moudleid == presetVar.head.head1 ?
              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['customer'].value) :
              (props.cardTable.getClickRowIndex(moudleid).record.values['customer'].value))

15             }
16             let bzbm = (moudleid == presetVar.head.head1 ?
              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['bzbm'].value)
              : (props.cardTable.getClickRowIndex(moudleid).record.values['bzbm'].value))

```

```

17         return { pk_cust:
    pk_cust, accclass: accclass, pk_currtype: bzbm, TreeRefActionExt: '', DataPowerOperati
    onCode: 'fi', /**使用权组**/ isDataPowerEnable: 'Y' };
18     }
19 }

```

1. if (item.attrcode == 'custaccount') {

这行代码是判断当前操作的项目是否为“客商银行账号”。如果item.attrcode的值等于'custaccount'，那么下面的逻辑才会执行。

2. let paytarget = ...;

获取paytarget值，这个值决定了收款对象的类型。paytarget可能有两值：

- "1"：收款对象为供应商
- "2"：收款对象为客户

3. if (paytarget == "1") { ... } 和 else if (paytarget == "2") { ... }

根据paytarget的值，代码会分别处理：

- **如果是供应商**：设置accclass为"3"，并且从表单或表格中获取hbbm（供应商的ID）。
- **如果是客户**：设置accclass为"2"，并且从表单或表格中获取customer（客户的ID）。

4. let bzbm = ...;

获取bzbm，即货币类型。这个值来自于表单或表格。

5. 返回过滤条件

最终返回一个包含了pk_cust（客户或供应商的ID）、accclass（账户类别）、pk_currtype（货币类型）、以及一些额外参数的对象。

关键点

- **props.form.getAllFormValue() 和 props.cardTable.getClickRowIndex()**：这些函数是从表单或表格中获取数据，props.form 和 props.cardTable 表示在前端框架中的数据状态，它们是用来访问表单和表格组件的。
- **条件判断**：通过if-else语句来根据不同的paytarget值决定操作，涉及到了常见的条件判断和赋值操作。
- **返回过滤条件**：最终返回的对象是用来作为过滤条件的，这常用于从后端请求数据时筛选数据。

修改代码解读：

修改代码：

代码块

```
1      // 申请单自定义项31 (defitem31, 客商银行账号) 过滤条件, 仅供应商
2      if (item.attrcode == 'defitem44') {
3          item.queryCondition = function() {
4              var pk_cust = moduleid == presetVar.head.head1
5                  ?
6              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier']
7                  &&
8              props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier'].
9                  value)
10                 :
11              (props.cardTable.getClickRowIndex(moduleid).record.values['pk_supplier'] &&
12                  props.cardTable.getClickRowIndex(moduleid).record.values['pk_supplier'].value);
13
14              var pk_currtype = moduleid == presetVar.head.head1
15                  ?
16              (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_currtype']
17                  &&
18              props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_currtype'].
19                  value)
20                 :
21              (props.cardTable.getClickRowIndex(moduleid).record.values['pk_currtype'] &&
22                  props.cardTable.getClickRowIndex(moduleid).record.values['pk_currtype'].value);
23
24              return {
25                  pk_cust: pk_cust,
26                  accclass: '3',
27                  pk_currtype: pk_currtype,
28                  TreeRefActionExt: '',
29                  DataPowerOperationCode: 'fi',
30                  /**使用权组**/ isDataPowerEnable: 'Y'
31              };
32          };
33      }
34  }
```

1. if (item.attrcode == 'defitem44') {

这行判断代码是根据item对象的attrcode属性来决定是否执行后面的逻辑。如果item.attrcode的值是'defitem44', 则会执行下面的item.queryCondition定义。

- defitem44是自定义项的标识符，表示这是一个在申请单中定义的特定字段。

2. item.queryCondition = function() {...}

这里是给item.queryCondition属性赋值一个函数。这个函数会在运行时被调用，用于构造一个查询条件对象。查询条件对象通常会作为过滤条件传递给后端，以便只获取符合条件的数据。

3. var pk_cust = ...

这一行代码用于定义一个变量pk_cust，它表示客户（或供应商）的ID。这个ID是根据当前操作模块来获取的：

- 如果当前模块是presetVar.head.head1，则从表单数据
(props.form.getAllFormValue(presetVar.head.head1)) 中获取字段pk_supplier的值。
- props.form.getAllFormValue(presetVar.head.head1)返回的是表单中所有的数据，其中rows[0].values['pk_supplier']获取的是供应商ID（pk_supplier字段的值）。
- 如果不是该模块，则从卡片表格数据
(props.cardTable.getClickRowIndex(moduleid).record.values['pk_supplier']) 中获取供应商ID。

通过这段代码，pk_cust会获取供应商的ID（无论是通过表单还是通过卡片表格）。

4. var pk_currtype = ...

这一行代码与pk_cust类似，定义了一个变量pk_currtype，表示货币类型的ID。它的获取逻辑与pk_cust类似：

- 如果当前模块是presetVar.head.head1，则从表单中获取pk_currtype字段的值。
- 如果不是该模块，则从卡片表格数据中获取pk_currtype的值。

5. 返回查询条件对象

最终，queryCondition函数会返回一个包含以下属性的对象，作为查询条件：

- **pk_cust**: 供应商ID，通过上面的逻辑得到。
- **accclass: '3'**: 账户类别，这里硬编码为'3'，表示供应商。
- **pk_currtype**: 货币类型ID，通过上面的逻辑得到。
- **TreeRefActionExt: ''**: 该字段为空，可能是为了满足某些接口要求，或者用于后端接口的扩展字段。

⚠ • **DataPowerOperationCode: 'fi'**: 这个字段可能是用于标识数据操作类型的，'fi'可能表示财务相关操作。

- **isDataPowerEnable: 'Y'**: 这个字段可能用于权限控制，标识是否启用数据权限。

（这两个字段看起来确实是 **Cursor** 自己增加的，用于满足后端扩展的需求和接口操作的标识，bip中并没有用到）

知识点：

1. props.form.getAllFormValue

是一个函数，用来在表格中 获取所有的表单数据；

2. props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier'].value

代码块

```
1  ##
   props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier'].
   value
2
3  这段代码是对表单数据结构的深入访问，它由几个部分组成：
4
5  - **presetVar.head.head1**：这是一个特定表单的标识符（ID）。它告诉代码从哪个表单中获取
   数据。
6
7  - **.rows[0]**：表单数据通常会有多行（类似表格），这里的 `rows[0]` 表示访问 **第一行的
   数据**。
8
9  - **.values['pk_supplier']**：每一行数据都是一个包含字段名和对应值的对象，
   `values['pk_supplier']` 就是访问该行数据中字段名为 `pk_supplier` 的字段值。
10
11 - **.value**：大多数表单字段的实际数据通常存放在 `value` 属性中。因此，
   `values['pk_supplier'].value` 就是获取 **`pk_supplier` 字段的具体值**，也就是供应
   商 ID。
```

特定表单的第一行数据 rows[0] 中，获取 pk_supplier 字段的 实际值（value）

3. var 关键字

var 是 JavaScript 中用来声明变量的关键字。它有以下特点：

·作用域：var 声明的变量是 函数作用域 的，意味着它的作用范围仅限于函数内部

4. return 语句

return 语句用于从一个函数中返回一个值。它可以是任何数据类型（字符串、数字、对象等）。

理解之后再次修改代码

代码块

```
1  if (item.attrcode == 'defitem44') {
```

```

2      item.queryCondition = function() {
3          var pk_cust = moduleid == presetVar.head.head1
4              ?
      (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_supplier']
      .value)
5              : ""; // 默认值为空字符串, 或根据需求使用其他值
6
7          var pk_currtype = moduleid == presetVar.head.head1
8              ?
      (props.form.getAllFormValue(presetVar.head.head1).rows[0].values['pk_currtype']
      .value)
9              : ""; // 同样给定默认值
10
11      return {
12          pk_cust: pk_cust,
13          accclass: '3',
14          pk_currtype: pk_currtype,
15          TreeRefActionExt: '',
16          DataPowerOperationCode: 'fi',
17          isDataPowerEnable: 'Y'
18      };
19  };
20  }

```

课后问题

1. `let` 和 `var` 有什么区别?
2. `moduleid == presetVar.head.head1` 是什么意思? --判断

[📖 前端学习计划](#)