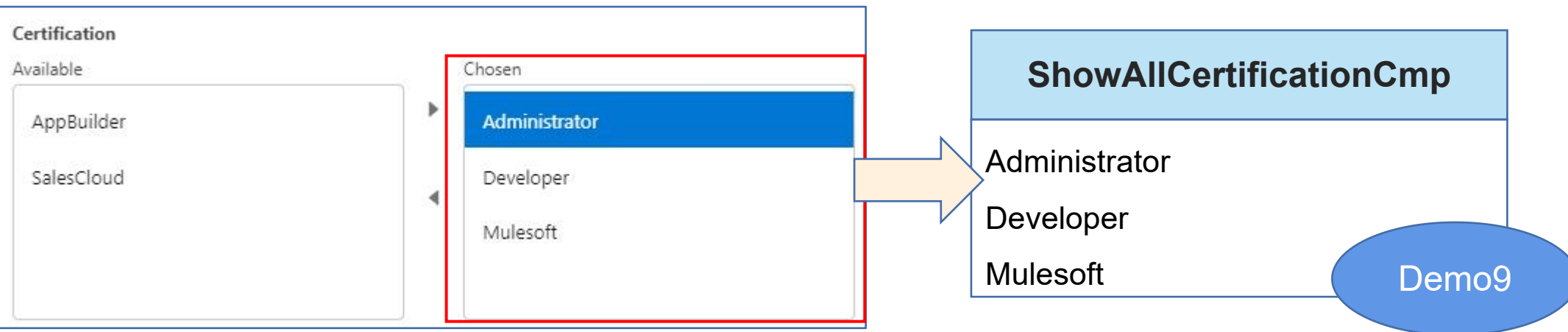


知识点： 使用Apex取得数据

- ①使用controller
- ②使用AuraEnabled方法并设置参数
- ③使用JavaScript API的Action

课题： 显示所有选择的值到Aura组件上面



详细步骤

- 1.准备ApexClass读取Study对象
- 2.在Aura的Controller里面使用Action去接收Apex的传值
- 3.在Aura的Controller中设置Attribute的值

知识点: 所有的基础类型的Attribute

Demo17

所有的基础类型的Attribute

| | |
|----------|--|
| Boolean | <code><aura:attribute name="showDetail" type="Boolean" /></code> |
| Date | <code><aura:attribute name="startDate" type="Date" /></code> |
| DateTime | <code><aura:attribute name="lastModifiedDate" type="DateTime" /></code> |
| Decimal | <code><aura:attribute name="totalPrice" type="Decimal" /></code> |
| Double | <code><aura:attribute name="widthInchesFractional" type="Double" /></code> |
| Integer | <code><aura:attribute name="numRecords" type="Integer" /></code> |
| Long | <code><aura:attribute name="numSwissBankAccount" type="Long" /></code> |
| String | <code><aura:attribute name="message" type="String" /></code> |

知识点:

- ①使用controller
- ②使用AuraEnabled方法并设置参数
- ③使用JavaScript API的Action

课题: 使用Apex取得数据显示所有Case信息



30 items • Sorted by Case Number • Filtered by My cases • Updated a few seconds ago

| <input type="checkbox"/> | Case Nu... ↑ | Subject |
|--------------------------|-----------------------------------|--|
| 1 | <input type="checkbox"/> 00001000 | Starting generator after electrical failure |
| 2 | <input type="checkbox"/> 00001001 | Performance inadequate for second consec... |
| 3 | <input type="checkbox"/> 00001002 | Seeking guidance on electrical wiring install... |
| 4 | <input type="checkbox"/> 00001003 | Easy installation process |
| 5 | <input type="checkbox"/> 00001004 | Maintenance guidelines for generator uncle... |
| 6 | <input type="checkbox"/> 00001005 | Electrical circuit malfunctioning |

ShowAllCaseCmp

Demo10

显示所有Study数据

00001000 →→ Starting generator after electrical failure
00001001 →→ Performance inadequate for second consecutive week
00001002 →→ Seeking guidance on electrical wiring installation for GC5060
00001003 →→ Easy installation process
00001004 →→ Maintenance guidelines for generator unclear

显示所有Study数据

知识点： 使用lightning:recordEditForm更新数据

- ①使用controller
- ②使用aura:handler来调用初始化方法
- ③使用AuraEnabled方法并设置参数
- ④使用JavaScript API的Action
- ⑤lightning:recordEditForm的submit按钮隐式更新数据
- ⑥lightning:card的使用

课题： 在Aura组件中更新Case信息

5005h00000IahYyAAJ

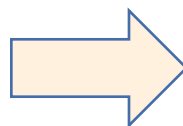
Subject

Starting generator after electrical failure YuTest

Status

New

Update case



UpdateCaseCmp

Demo11



Case

Starting generator after electrical failure YuTest

Priority
High

Status
New

Case Number
00001000

知识点： 使用Apex**显式更新**数据

- ①使用controller
- ②使用aura:handler来调用初始化方法
- ③使用AuraEnabled方法并设置参数
- ④使用JavaScript API的Action
- ⑤点击按钮调用controller方法调用AuraEnabled的Apex方法**显式更新**数据

课题： **显示更新数据** (但不能刷新页面)

Aura Call LWC Study

请输入备考 (Memo) 的值:

更新Study数据

UpdateStudyCmp

Related

Details

Study名

Aura Call LWC Study

Owner

 俞思忠

Record Type

Memo

YuTest

Demo12

知识点: 刷新页面
(Aura Component: Refresh View)

课题: 显示更新数据后的刷新页面

Aura Call LWC Study

请输入备考 (Memo) 的值:

更新Study数据

UpdateStudyCmp

Related

Details

Study名

Aura Call LWC Study

Owner

 俞思忠

Record Type

Memo

YuTest

Demo13

```
myAction.setCallback(this, function(response) {
    var state = response.getState();
    if (state === "SUCCESS") {
        component.set("v.studyDatas", response.getReturnValue());

        // Refresh the view, after data is updated
        $A.get('e.force:refreshView').fire();
        console.log("更新Study数据----->" + response.getReturnValue());
    }
});
$A.enqueueAction(myAction);
```

知识点: 刷新页面
(Aura Component: Refresh View)

课题: 隐式更新数据后的刷新页面

Aura Call LWC Study

请输入备考 (Memo) 的值:

更新Study数据

UpdateStudyCmp

Related

Details

Study名

Aura Call LWC Study

Owner

 俞思忠

Record Type

Memo

YuTest

Demo14

```
<lightning:recordEditForm objectApiName="Study__c"
  recordId="{!StudyItem.Id}"
  onsuccess="{!c.handleSuccess}">
  <lightning:inputField fieldName="Name"/>
  <lightning:inputField fieldName="Memo__c"/>

  <lightning:button label="更新Study数据" type="submit"/>
</lightning:recordEditForm>
```

```
//更新数据成功的后处理
handleSuccess: function(component, event, helper) {
  console.log("更新数据成功的后处理----->");

  // Refresh the view, after data is updated
  $A.get('e.force:refreshView').fire();
},
```

知识点： 作为aura:method的一个属性，可以传一个Javascript的方法
讲到调用Aura组件的方法的时候，会用到这个FunctionType属性

知识点:

- ①使用Custom Apex Class类型的Attribute
- ②使用JSON.stringify()把Object类型转化成文本类型

课题: 使用Apex取得数据显示所有Case信息并存放ApexClass类型的Attribute中

Demo16

ShowAllStudyCmp

显示所有Study数据

a005h00000eY9upAAC →→ Aura Call LWC Study
 a005h00000kQnJ2AAK →→ test
 a005h00000mCEakAAG →→ newpage
 a005h00000kPAYJAA4 →→ 動的検索項目
 a005h00000HE0UpAAL →→ A-0002
 a005h00000HE0doAAD →→ A-0008

```

1 public class StudyData {
2     @auraEnabled
3     public String StudyId {get;set;}
4     @auraEnabled
5     public String StudyName {get;set;}
6     @auraEnabled
7     public String StudyMemo {get;set;}
8 }
    
```

转化前后的效果

```

recordId----->[object Object],[object Object],[object Object]
studyDatas----->[{"StudyId":"a005h00000eY9upAAC","StudyName":"Aura Call LWC Study"},{"StudyId":"a005h00000kQnJ2AAK","StudyName":"test"},{"StudyId":"a005h00000mCEakAAG","StudyName":"newpage"},{"StudyId":"a005h00000kPAYJAA4","StudyName":"動的検索項目"},{"StudyId":"a005h00000HE0UpAAL","StudyName":"A-0002"},{"StudyId":"a005h00000HE0doAAD","StudyName":"A-0008"}]
    
```

| Study | |
|--------------------------------------|---------------------|
| Recently Viewed ▼ | |
| 23 items • Updated a few seconds ago | |
| <input type="checkbox"/> | Study名 |
| 1 <input type="checkbox"/> | Aura Call LWC Study |
| 2 <input type="checkbox"/> | A-0013 |
| 3 <input type="checkbox"/> | newpage |
| 4 <input type="checkbox"/> | test |

所有的Framework-Specific类型

| | |
|------------------|---|
| Aura.Component | 单独的一个Aura组件 (不推荐) |
| Aura.Component[] | <aura:attribute name="detail" type="Aura.Component[]"> (推荐) |
| Aura.Action | <aura:attribute name="onclick" type="Aura.Action"/> |

Demo17

知识点： 使用<aura:html tag="style">来指定CSS属性

```
<aura:html tag="style">
  .slds-modal__container{
  width: 100% !important;
  max-width: 70rem !important;
  }
</aura:html>
```

父Aura组件

```
<aura:component access="global" >
<aura:attribute name="parentAttribute1" type="String" default="A"/>
<aura:attribute name="parentAttribute2" type="String" default="B"/>

<c:childComponentWithMethod aura:id="childCmp"/>

<lightning:button label="Call child method"
onclick="{! c.callChildMethod }" />
</aura:component>
```

```
{
callChildMethod : function(component, event, helper) {
//取得属性值
var attribute1 = component.get('v.parentAttribute1');
var attribute2 = component.get('v.parentAttribute2');

//在组件中找到这个ID
var childComponent = component.find('childCmp');
childComponent.myMethod(attribute1, attribute2);
}
})
```

子Aura组件

```
<aura:component >
<aura:method name="myMethod"
action="{!c.executeMyMethod}">
<aura:attribute name="param1"
type="String"/>
<aura:attribute name="param2"
type="String"/>
</aura:method>
</aura:component>
```

```
{
executeMyMethod : function (component,
event, helper) {
var params =
event.getParam('arguments');
console.log('Param 1: '+ params.param1);
console.log('Param 2: '+ params.param2);
}
})
```

父Aura组件

```
<aura:component access="global" >
<aura:attribute name="parentAttribute1"
type="String" default="A"/>
<aura:attribute name="parentAttribute2"
type="String" default="B"/>

<c:childComponentWithMethod aura:id="child"
param1="AAA" param2="{!v.parentAttribute2}" />

</aura:component>
```

子Aura组件

```
<aura:component>
<aura:attribute name="param1" type="String"/>
<aura:attribute name="param2" type="String"/>

<h1>Hello, I am Child Component.
Here is the value from Parent : {!v.param1}</h1>

</aura:component>
```

