第4天

componentEvent.evt

```
<aura:event type="COMPONENT" description="component event">
<aura:attribute name="greetingsMessage1" type="String"/>
</aura:event>
```

子Aura组件

```
<aura:component  access="global" >
<aura:attribute name="param1" type="String"
default="こんにちは！ "/>
<aura:registerEvent name="cmpEvent"
type="c:componentEvent"/>
<p><lightning:button
 label="进行传值" onclick="{!c.fireComponentEvent}" />
</p>
</aura:component>
```

```
var cmpEvent = component.getEvent("cmpEvent");
        cmpEvent.setParams({
            "greetingsMessage1": component.get('v.param1'),
        });
        cmpEvent.fire();
```

父Aura组件

```
<aura:component  access="global" >
<!-- 注意name="cmpEvent" 就是子组件的
aura:registerEvent的名称 -->
<aura:handler name="cmpEvent"
event="c:componentEvent"
action="{!c.handleComponentEvent}"/>

<c:childComponent/>
</aura:component>
```

```
var greetingsMessage1 =
event.getParam("greetingsMessage1");
```

Demo22

第4天

appEvent.evt

```
<aura:event type="APPLICATION" description="component event">
<aura:attribute name="greetingsMessage1" type="String"/>
</aura:event>
```

发消息的Aura组件

```
<aura:component  access="global" >
<aura:attribute name="param1" type="String"
default="こんにちは！ "/>
<aura:registerEvent name="appEvent"
type="c:appEvent"/>
<p><lightning:button
 label="进行传值" onclick="{!c.fireComponentEvent}" />
</p>
</aura:component>
```

```
var appEvent = $A.get("e.c:appEvent");
        appEvent.setParams({
           "greetingsMessage1": component.get('v.param1'),
           "greetingsMessage2": component.get('v.param2')
       });
       appEvent.fire();
```

接收消息的Aura组件

```
<aura:component  access="global" >

<aura:handler  event="c:appEvent"
action="{!c.handleApplicationEvent}"/>


</aura:component>
```

```
var greetingsMessage1 =
event.getParam("greetingsMessage1");
```

Demo23

第4天

Component Event

1. 在aura:event的attribute使用type="COMPONENT".
2. 我们使用cmp.getEvent("evtName") 来取得一个实例化的Component事件.
3. 在aura:handler中需要指定Name属性
4. ComponentEvent只能在子组件中定义，并被父组件接收消息.

Application Event

1. 在aura:event的attribute使用type="APPLICATION"
2. 我们使用$A.get("e.myNamespace:myAppEvent")来取得实例化的一个Application事件.
3. 在aura:handler中不需要指定Name属性
4. Application event 可以在整个Application中使用.

知识点:
①使用force:closeQuickAction来关闭当前quick action图面

第4天

关闭QuickAction面板

```
$A.get("e.force:closeQuickAction").fire();
```

知识点:
①使用force:createRecord来创建数据

第4天

force:createRecord的使用

```
var createRecordEvent = $A.get("e.force:createRecord");

    var RecTypeID  = '0125h000000Rp6D';

    createRecordEvent.setParams({
        "entityApiName": 'Study__c',
        "recordTypeId": RecTypeID,

        'defaultFieldValues': {
            'Certification__c' : 'Administrator; Developer',
            'Memo__c' : '我的备注',
            'CloseDate__c' : currentYear,
            "Name": '程序生成的数据',
        }
    });

    createRecordEvent.fire();
```

Demo25

知识点:
①使用force:editRecord来修改数据

第4天

force:editRecord的使用

```
var editRecordEvent = $A.get("e.force:editRecord");
editRecordEvent.setParams({"recordId": component.get("v.contact.Id")
});

editRecordEvent.fire();
```

Demo26

知识点:
①使用force:navigateToList来进行页面跳转

第4天

force:navigateToList的使用

```
var navEvent = $A.get("e.force:navigateToList");
navEvent.setParams({
    "listViewId": listviews.Id,
    "listViewName": null,
    "scope": "Study__c"
});
navEvent.fire();
```

Demo27

第4天

知识点:
①使用force:navigateToObjectHome来进行页面跳转
回到Object主页

force:navigateToObjectHome的使用

```
var homeEvent = $A.get("e.force:navigateToObjectHome");
homeEvent.setParams({
    "scope": "Study__c"
});
homeEvent.fire();
```

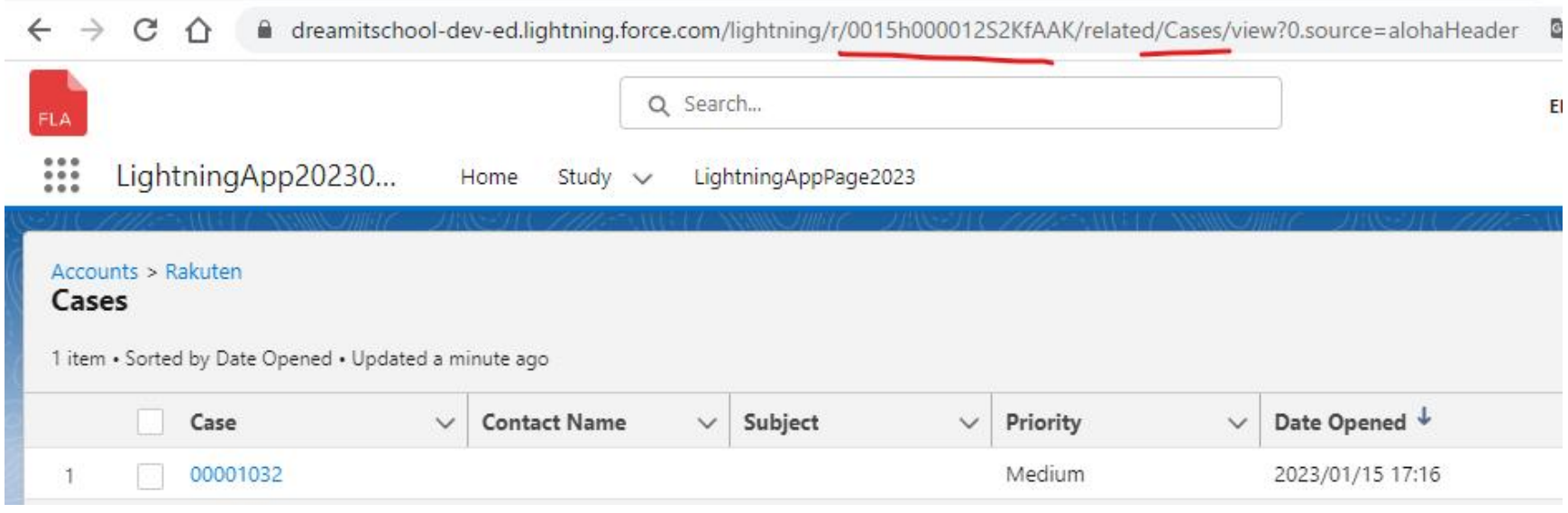
Demo28

第4天

知识点:
①使用force:navigateToRelatedList来进行页面跳转
跳到关联列表

force:navigateToRelatedList的使用

```
var relatedListEvent =
$A.get("e.force:navigateToRelatedList");
relatedListEvent.setParams({
    "relatedListId": "Cases",
    "parentRecordId": '0015h000012S2KfAAK'
});
relatedListEvent.fire();
```



Demo29

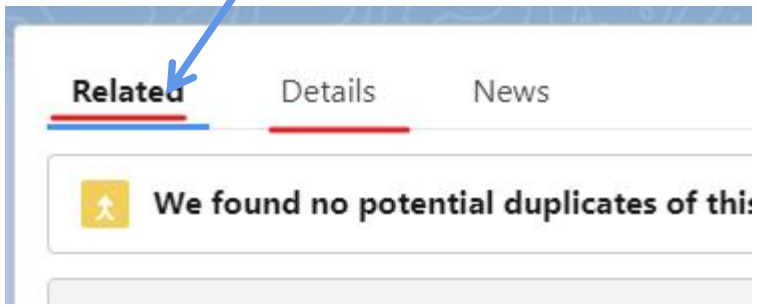
第4天

知识点:
①使用force:navigateToSObject来进行页面跳转
跳到数据的详细列表

force:navigateToSObject的使用

```
var navEvt = $A.get("e.force:navigateToSObject");
navEvt.setParams({
  "recordId": "0015h00000dzTJ7AAM",
  "slideDevName": "related"
});

navEvt.fire();
```



Demo30

第4天

知识点:
①使用force:navigateToURL来进行页面跳转
跳到一个URL

使用force:navigateToURL进行页面跳转

```
var urlEvent = $A.get("e.force:navigateToURL");
urlEvent.setParams({
  "url": "/006/o",
  // "url": 'https://space.bilibili.com/1601673630
});
urlEvent.fire();
```

Demo31

知识点:
①使用force:recordEdit 来保存数据

第4天

先使用force:recordEdit加载数据

```
<force:recordEdit aura:id="edit" recordId="{!v.recordId}"/>

<ui:button label="Save" press="{!c.saveRecord}"/>
```

再使用force:recordSave保存数据

```
saveRecord: function(component, event, helper) {
    component.find("edit").get("e.recordSave").fire();
},
```

Demo32

知识点:
①使用force:recordSaveSuccess来处理数据保存完成的后处理

第4天

1。先使用force:recordEdit加载数据

```
<force:recordEdit aura:id="edit" recordId="{!v.recordId}"/>

<ui:button label="Save" press="{!c.saveRecord}"/>
```

2。再使用force:recordSave保存数据

```
saveRecord: function(component, event, helper) {
    component.find("edit").get("e.recordSave").fire();
},
```

3。再使用force:recordSaveSuccess保存数据

```
handleSaveSuccess: function(component, event, helper)
{
       // Refresh the view, after data is updated
        $A.get('e.force:refreshView').fire();

    },
```

Demo32

第4天

知识点:
①使用force:showToast来显示弹出消息

先使用force:recordEdit加载数据

```
var toastEvent = $A.get("e.force:showToast");
toastEvent.setParams({
    "title": "成功!",
    "message": " 你的修改已经被保存成功"
});

toastEvent.fire();
```

Demo33

第4天

知识点:
①使用lightning:openFiles来显示文件

使用lightning:openFiles来显示单个文件

```
$A.get('e.lightning:openFiles').fire({
    recordIds: [component.get("v.currentContentDocumentId")]
});
```

Demo34

第4天

知识点:
①使用aura:valueInit来进行初始化

```
<aura:handler name="init" value="{!this}" action="{!c.doInit}"/>
```

Demo35

第4天

知识点:
①使用aura:valueChange来判断Attribulte的值是否改变

```
<aura:attribute name="MemoInput" type="String" />
```

```
<!-- Handle the aura:valueChange event -->
<aura:handler name="change" value="{!v.MemoInput}" action="{!c.handleValueChange}"/>
```

```
console.log("old value: " + event.getParam("oldValue"));
console.log("current value: " + event.getParam("value"));
```

Demo35

第4天


Study
AuraDelete
Customized削除
OverwriteWidth
削除確認
このレコードを削除します。よろしいですか?
キャンセル
削除
Study名
1
A-0001
2
A-0002
3
A-0003
4
A-0004

Demo24