Creating Components for Sobjects

Apex invocations using anonymous callback function

How to invoke Aura Enabled Apex contollers similar to vf pages

- Apex controller methods can be invoked from Component controllers
- Write your apex class and method first with required logic
- <aura:component controller="ContactController">
- Add a reference to the controller to your component (similar to visualforce pages)
- Add @AuraEabled Annotation to the method you plan to invoke in your lightning component

```
public with sharing class ContactController {

   @AuraEnabled
   public static String retcontactname(Id id) {
  List<Contact> con = new List<Contact>();
     con = [SELECT name FROM Contact where id =:id];
     return con[0].name;
   }
   }
```

How to invoke Apex controllers

- Based on Javascript fetch http and .then methods
- Get a reference of the method inside the javascript component controller methods for example : var action = component.get("c.retcontactname");
- Set parameters for the apex class method
- Set object.setcallback method which invokes the controller method and gets the result of the call. Assign the return variables if any to any attributes of the component
- Finally add this line $A.enqueueAction(action); to the method ,this will queue your request to the server.

```
//first get action variable intialised with controller meth
var action = component.get("c.retcontactname");
action.setParams({ id: component.get("v.Conid") });
action.setCallback(this, function(a) {
    component.set("v.ContactName", a.getReturnValue());
    alert(a.getReturnValue());
});
$A.enqueueAction(action);
```

Callbacks status

- NEWThe action was created but is not in progress yet
- RUNNINGThe action is in progress
- SUCCESSThe action executed successfully
- ERRORThe server returned an error
- INCOMPLETEThe server didn’t return a response.

Common Errors-will take up most of your debuggin' time

- Case sensitivity affects variable names
- Missing , when multiple controllers are added
- Wrong field names in the component
- Writing logic in application bundle instead of component bundle
- Parameter names or values are wrong
- SetCallback errors : Wrong method name (casesensitive),Wrong params passed,incorrect variable names .
- SetCallback works but result is delayed due to network,add redirect code inside the callback } sections
- Js methods like action.setCallback , $A.enqueueAction will not work if not written in exact case as mentioned in documentation. For example enqueaction wont work.

- Create lightning Component to Save .edit ,delete Contact Data to replace visualforce page functionality

Key interfaces to be implemented

- implements="lightning:actionOverride,
- force:lightningQuickAction,
- force:appHostable,flexipage:availableForAllPageTypes,
- flexipage:availableForRecordHome,l
- ightning:actionOverride,
- **force:hasRecordId" automatically created recordid variables and fills with Id of current record in edit mode**

Important methods

- Save and Edit logical functions should be created based on recordid
- Save and edit will call same apex class but different methods for different purposes
- Save and edit will pass correct parameters to class via
- action.setParams({Idd :component.get("v.recordId")});
- //
- action.setCallback(this, function(a) {

Important methods

- Use the do init method in js controller to invoke an apex class to fill the data in edit mode,incase the button is overriden by lightning component

Implement SLDS

- Use SLDS ready bootstrap to create forms
- Define correct Sobject variables
- Define relevant attributes for fields
- Fill up the data in edit mode by doing a fetch in the init methods.
- Pass and save correct data and perform redirection
- var event = $A.get('e.force:navigateToSObject');
- event.setParams({
-
- "recordId": x,
- "slideDevName": "related"
-
- });
- event.fire();
-

How are components rendered

- Components can be embedded in an Lightning application
- Expose them as lightning tabs or Apps
- Expose them as standard button overrides
- Within Developer console you can only preview application containing components
- Redirection may not happen within developer console
- Practise: Create the component and show it in app/tab