Salesforce Integrate Maps

---

lightningMapController.apxc

/*
Support API: 47.00
Author: Mahesh Ambati
Source: Prowess Software Services
*/

```
public with sharing class lightningMapController {
   @AuraEnabled
   public static list<accountLocationWrapper> getLocation(){
        list<accountLocationWrapper> lstALW = new list<accountLocationWrapper>();
      /*Query accounts records with billing address information */
      for(account acc : [Select Name,description, BillingCountry, BillingCity, BillingPostalCode, BillingStreet,
BillingState
                  From Account
                  Where BillingCountry != null
                  And BillingCity != null
                  limit 10]){
                     // first create "locationDetailWrapper" instance and set appropriate values
                     locationDetailWrapper oLocationWrap = new locationDetailWrapper();
                     oLocationWrap.Street = acc.BillingStreet;
                     oLocationWrap.PostalCode = acc.BillingPostalCode;
                     oLocationWrap.City = acc.BillingCity;
                     oLocationWrap.State = acc.BillingState;
                     oLocationWrap.Country = acc.BillingCountry;

                     // create "accountLocationWrapper" instance, set values and add to final list.
                     accountLocationWrapper oWrapper = new accountLocationWrapper();
                     oWrapper.icon = 'utility:location'; // https://www.lightningdesignsystem.com/icons/#utility
                     oWrapper.title = acc.Name;
                     oWrapper.description = acc.description;
                     oWrapper.location = oLocationWrap;

                     lstALW.add(oWrapper);
                  }
      // retrun the "accountLocationWrapper" list
      return lstALW;
   }
```

```
    /*Account details fetching */
            Public Static List<Account> getSearchAccount(String AccountSearch){

       string searchquery='select Id,Name from Account where name = like \'%'+AccountSearch+'%\' Limit 20';
       List<Account> acc = Database.query(searchquery);
       return acc;
    }

    /* wrapper class to store required properties for "lightning:map" component' */
    public class accountLocationWrapper{
       @AuraEnabled public string icon{get;set;}
       @AuraEnabled public string title{get;set;}
       @AuraEnabled public string description{get;set;}
       @AuraEnabled public locationDetailWrapper location{get;set;}
    }
    /* sub wrapper class to store location details for "accountLocationWrapper" location property.*/
    public class locationDetailWrapper{
       @AuraEnabled public string Street{get;set;}
       @AuraEnabled public string PostalCode{get;set;}
       @AuraEnabled public string City{get;set;}
       @AuraEnabled public string State{get;set;}
       @AuraEnabled public string Country{get;set;}
    }

}
```

lightningMapController.cmp

```
<!---
Support API: 47.00
Author: Mahesh Ambati
Source: Prowess Software Services
-->

-----------------------------------------------------------------------

<aura:component controller="lightningMapController"
implements="force:appHostable,flexipage:availableForAllPageTypes,flexipage:availableForRecordHome,force:hasRecordId,forceCommunity:availableForAllPageTypes,force:lightningQuickAction" access="global">
    <!-- aura attributes to store Map component information -->
    <aura:attribute name="mapMarkersData" type="Object"/>
    <aura:attribute name="AccountSearch"  type="string"/>
    <aura:attribute name="mapCenter" type="Object"/>
    <aura:attribute name="zoomLevel" type="Integer" default="4" />
    <aura:attribute name="markersTitle" type="String" />
    <aura:attribute name="showFooter" type="Boolean" default="true"/>

    <!-- init handler which will call 'doInit' fucntion on component load-->
    <aura:handler name="init" value="{! this }" action="{! c.doInit }"/>
```

```
    <lightning:input label="AccountSearch" value="{!v.AccountSearch}"/>

   <lightning:Button label="Search" onclick="{!c.Search}"/>
   <!-- render map component only when we have at least 1 record in list.(mapMarkersData)
-->
   <aura:if isTrue="{!v.mapMarkersData.length > 0}" >
      <!-- the map component -->
      <lightning:map mapMarkers="{! v.mapMarkersData }"
                     center="{! v.mapCenter }"
                     zoomLevel="{! v.zoomLevel }"
                     markersTitle="{! v.markersTitle }"
                     showFooter="{ !v.showFooter }" />
   </aura:if>
</aura:component>
```

lightningMapController.js

```
({
   doInit: function (component, event, helper) {
      // call getLocation apex class method
      var action = component.get("c.getLocation");
      action.setCallback(this, function(response) {
         var state = response.getState();
         if (state === "SUCCESS") {
            // set mapMarkersData attribute values with 'accountLocationWrapper' data
            component.set('v.mapMarkersData',response.getReturnValue());
            // set the mapCenter location.
            component.set('v.mapCenter', {
               location: {
                  Country: 'United States'
               }
            });
            // set map markers title
            component.set('v.markersTitle', 'Account members.');
         }
         else if (state === "INCOMPLETE") {
            // do something
         }
            else if (state === "ERROR") {
               var errors = response.getError();
               if (errors) {
                  if (errors[0] && errors[0].message) {
                     console.log("Error message: " +
                              errors[0].message);
                  }
               } else {
                  console.log("Unknown error");
               }
            }
```

```
        });
        $A.enqueueAction(action);
    },


    // get Accounts
     Search : function(component, event, helper) {
         var action = component.get("c.getSearchAccount");
        action.setParams({ searchText : component.get("v.AccountSearch") });

        action.setCallback(this, function(response) {
            var state = response.getState();
            if (state === "SUCCESS") {
                component.set('v.mapMarkersData',response.getReturnValue());
                  component.set('v.mapCenter', {
                   location: {
                      Country: 'United States'
                   }
                });

                //  component.set('v.searchResult',response.getReturnValue().length);
                  //alert("From server: " + response.getReturnValue());
            }
            else if (state === "INCOMPLETE") {
                // do something
            }
                else if (state === "ERROR") {
                    var errors = response.getError();
                    if (errors) {
                        if (errors[0] && errors[0].message) {
                            console.log("Error message: " +
                                    errors[0].message);
                        }
                    } else {
                        console.log("Unknown error");
                    }
                }
        });
        $A.enqueueAction(action);



    }
})
```

TestApp.app

```
<aura:application extends="force:slds">
   <c:lightningMapController/>
<!-- here c: is org. default namespace prefix-->
</aura:application>
```

After implementing Salesforce Maps we have added Lightning component to Lightning ComponentTab

Navigation: Setp---→ Quick find box ----→ Tabs---→ Lightning Component Tabs -→ New → Component Name Like “C:lightningMapController”

Scope: Search the account on the map by using city name

Screen Shots:-