59. Data Table and Looping Statements, On Blur, Like Keyword - 27 June 2022

Today we will learn:

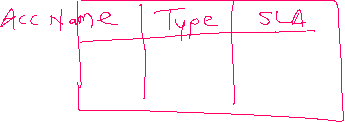
1] Data Table

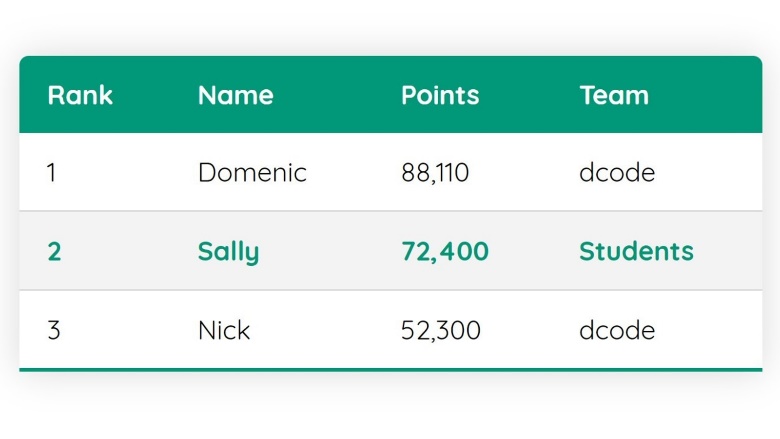
2] Looping Statement / Iteration

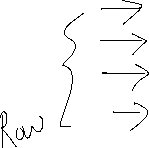
3] On Blur

4] Like Keyword

User Story 1] Display all the Accounts when a Type is selected from the Picklist.







<table aria-describedby="conatact-list" class="slds-table slds-table\_bordered ">

<!--Header of the table-->

**<thead>**

**<tr class="slds-line-height\_reset">**

**<th class="slds-size\_1-of-6 " scope="col">**

**<div class="slds-truncate " title="Account Name">Account Name</div>**

**</th>**

**<th class="slds-size\_1-of-6 " scope="col">**

**<div class="slds-truncate " title="Account Type">Type</div>**

**</th>**

**</tr>**

</thead>

<!--Body of the table-->



<tbody>



<template for:each={accList} for:item="acc">

<tr key={acc.id}>

<td>

{acc.Name}

</td>

<td>

{acc.Type}

</td>

</tr>

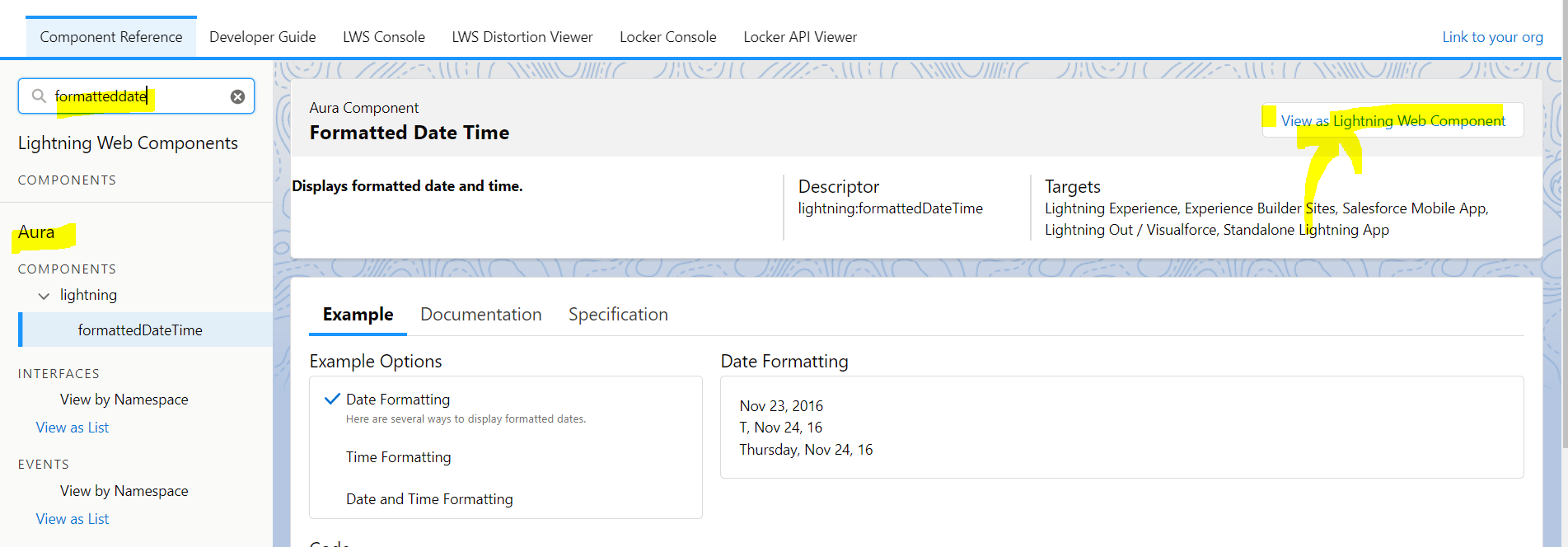
</template>

</tbody>

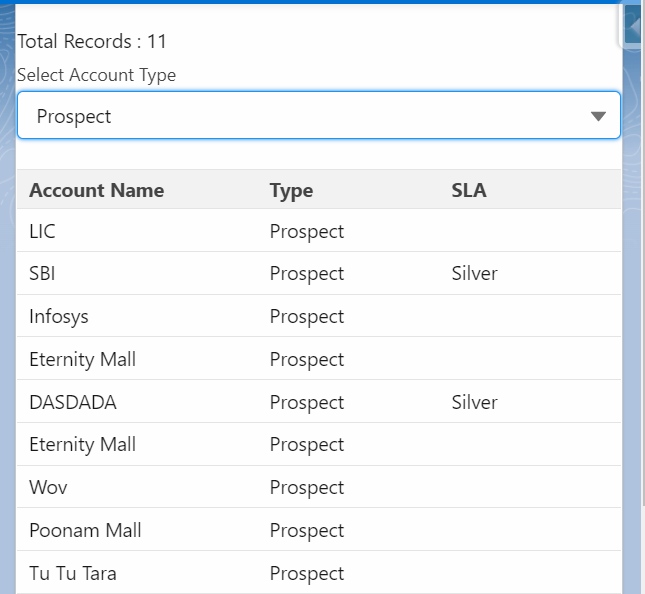
</table>

List<Account> accList = [select ID, Name from Account]; //11

|  |  |
| --- | --- |
| **Apex Code** | **Table Script (Iteration)** |
| for(Account objAcc : accList){  System.debug(objAcc.Id);  System.debug(objAcc.Name);  } | <template for:each={accList} for:item="objAcc">  <tr key={acc.id}>  <td>  {acc.Name}  </td>  <td>  {acc.Type}  </td>  </tr>  </template> |



<lightning-formatted-date-time value="1547250828000"></lightning-formatted-date-time></p>



<template>

    <lightning-card title="Account Table"> <br />

        Total Account Records Found ={countOfAccountRecords} <br />

        <lightning-combobox

            name="accType"

            label="Select Account Type"

            value={accType}

            placeholder="Select Account Type"

            options={accountTypes}

            onchange={selectedAccountTypeHandler} ></lightning-combobox> <br /><br/>

            <table aria-describedby="conatact-list" class="slds-table  slds-table\_bordered ">

                <!--Header of the table-->

                <thead>

                    <tr class="slds-line-height\_reset">

                        <th class="slds-size\_1-of-6 " scope="col">

                            <div class="slds-truncate " title="Account Name">Account Name</div>

                        </th>

                        <th class="slds-size\_1-of-6 " scope="col">

                            <div class="slds-truncate " title="Account Type">Type</div>

                        </th>

                        <th class="slds-size\_1-of-6 " scope="col">

                            <div class="slds-truncate " title="Account SLA">SLA</div>

                        </th>

                        <th class="slds-size\_1-of-6 " scope="col">

                            <div class="slds-truncate " title="Account CreatedDate">Created Date</div>

                        </th>

                   </tr>

                </thead>

                <!--Body of the table-->

                <tbody>

                       <template for:each={accList} for:item="objAcc">

                            <tr key={objAcc.id}>

                                  <td>

                                        {objAcc.Name}

                                  </td>

                                  <td>

                                    {objAcc.Type}

                                </td>

                                <td>

                                    {objAcc.SLA\_\_c}

                              </td>

                              <td>

                                <lightning-formatted-date-time value={objAcc.CreatedDate}></lightning-formatted-date-time>

                            </td>

                            </tr>

                       </template>

                </tbody>

            </table>

    </lightning-card>

</template>

import { LightningElement } from 'lwc';

import searchAccountAsPerType from '@salesforce/apex/AccountProvider.searchAccountAsPerType'

export default class AccountTable extends LightningElement {

    objAccount = {'sObjectType' : 'Account'};

    accList;

    countOfAccountRecords = 0;

    get accountTypes() {

        return [

            { label: 'Prospect', value: 'Prospect' },

            { label: 'Customer - Direct', value: 'Customer - Direct' },

            { label: 'Other', value: 'Other' },

        ];

    }

    selectedAccountTypeHandler(event){

        console.log(event.detail.value);

        this.objAccount.Type = event.detail.value;

        searchAccountAsPerType({ objAcc : this.objAccount})

        .then((result) =>{

            console.log(JSON.stringify(result) );

            this.accList = result;

            this.countOfAccountRecords = result.length;

        })

        .catch((error) =>{

            console.log(JSON.stringify(error) );

            this.accList = null;

            this.countOfAccountRecords = 0;

            alert('Locha....');

        });

    }

}

public with sharing class AccountProvider {

@AuraEnabled

  public static List<Account> searchAccountAsPerType(Account objAcc){

    try {

        if(objAcc != null){

            return [select Id,Type,Name, SLA\_\_c, CreatedDate from Account where Type =: objAcc.Type];

        }

        else{

            return null;

        }

    } catch (Exception e) {

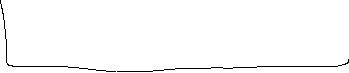
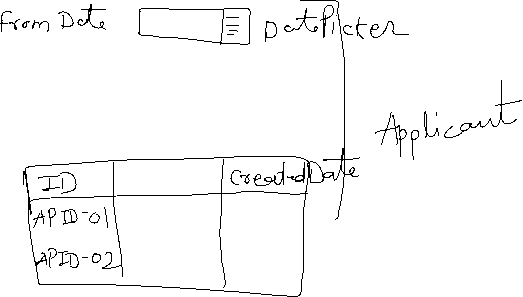
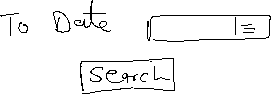
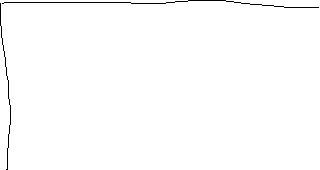
        throw new AuraHandledException(e.getMessage());

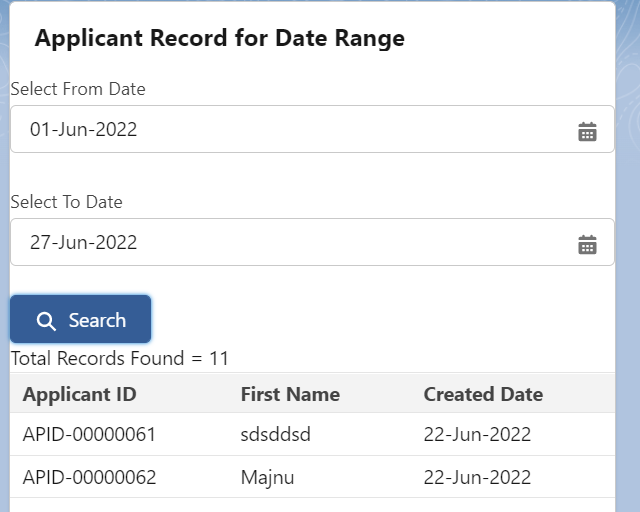
    }

  }

}

User Story 2] Display the Applicant Records created in the selected Date Range.





<template>

    <lightning-card title="Applicant Record for Date Range">

        <lightning-input type="Date" data-formfield="fromDate" label="Select From Date"></lightning-input> <br/>

        <lightning-input type="Date" data-formfield="toDate" label="Select To Date"></lightning-input> <br/>

        <lightning-button variant="brand" icon-name="utility:search" label="Search" onclick={searchApplicantRecordHandler}></lightning-button> <br />

<template if:true={showAppTable}>

    Total Records Found = {totalRecords} <br />

        <table aria-describedby="conatact-list" class="slds-table  slds-table\_bordered ">

            <thead>

                <tr class="slds-line-height\_reset">

                    <th class="slds-size\_1-of-6 " scope="col">

                        <div class="slds-truncate " title="Applicant ID">Applicant ID</div>

                    </th>

                    <th class="slds-size\_1-of-6 " scope="col">

                        <div class="slds-truncate " title="First Name">First Name</div>

                    </th>

                    <th class="slds-size\_1-of-6 " scope="col">

                        <div class="slds-truncate " title="Created Date">Created Date</div>

                    </th>

               </tr>

            </thead>

            <!--Body of the table-->

            <tbody>

                   <template for:each={applicantList} for:item="objApp">

                        <tr key={objApp.id}>

                              <td>

                                    {objApp.Name}

                              </td>

                              <td>

                                {objApp.First\_Name\_\_c}

                            </td>

                            <td>

                                <lightning-formatted-date-time value={objApp.CreatedDate}></lightning-formatted-date-time>

                            </td>

                        </tr>

                   </template>

            </tbody>

        </table>

    </template>

    <template if:false={showAppTable}>

        No Records Found...!!!

    </template>

    </lightning-card>

</template>

import { LightningElement } from 'lwc';

import searchApplicantDateRange from '@salesforce/apex/ApplicantProvider.searchApplicantDateRange';

export default class ApplicantRecordsDateRange extends LightningElement {

    fromDate;

    toDate;

    applicantList;

    totalRecords;

    showAppTable = false;

    searchApplicantRecordHandler(){

        this.fromDate = this.template.querySelector('lightning-input[data-formfield="fromDate"]').value;

        this.toDate = this.template.querySelector('lightning-input[data-formfield="toDate"]').value;

        console.log(this.fromDate+','+this.toDate);

        searchApplicantDateRange({fromDate : this.fromDate , toDate : this.toDate})

        .then((result) =>{

            console.log(JSON.stringify(result) );

            this.applicantList = result;

            this.totalRecords = result.length;

            if(result.length > 0 ){

                this.showAppTable = true;

            }

            else{

                this.showAppTable = false;

            }

        })

        .catch((error) =>{

            console.log(JSON.stringify(error) );

            this.showAppTable = false;

        });

    }

}

public with sharing class ApplicantProvider {

@AuraEnabled

   public static List<Applicant\_\_c> searchApplicantDateRange(Date fromDate,Date toDate){

    try {

        if(fromDate!=null && toDate !=null){

            return [select Id,Name, First\_Name\_\_c, CreatedDate from Applicant\_\_c where CreatedDate>=:fromDate and CreatedDate <=: toDate];

        }

        else{

            return null;

        }

    } catch (Exception e) {

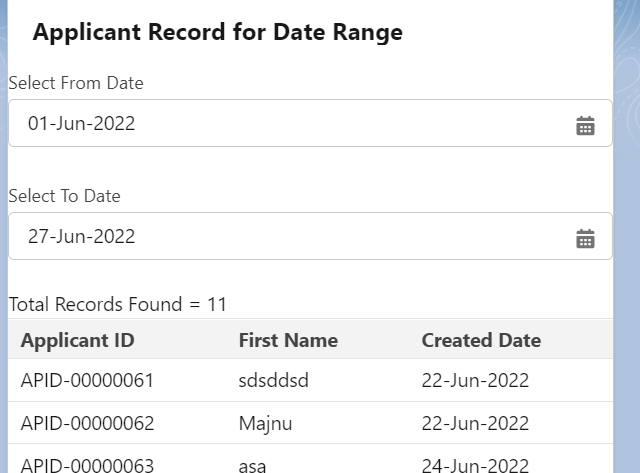
        throw new AuraHandledException(e.getMessage());

    }

   }

}

onBlur

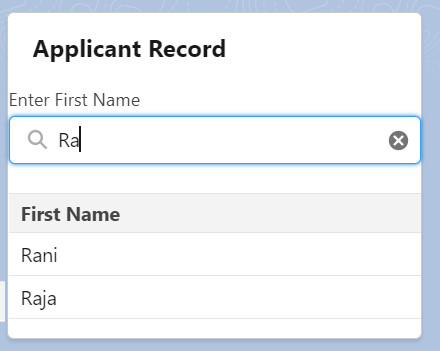


<lightning-input type="Date" data-formfield="fromDate" label="Select From Date"></lightning-input> <br/>

        <lightning-input type="Date" data-formfield="toDate" label="Select To Date" onblur={searchApplicantRecordHandler}></lightning-input> <br/>

User Story 3] Create a Field “Enter First Name”. When Type “A”, show all the Applicants whose Name is Starting with ‘A’. IN A TABLE

If Enter ‘AB’, then show all the Applicants whose Name is Starting with ‘AB’. IN A TABLE



<template>

    <lightning-card title="Applicant Record ">

       <lightning-input  type="search" data-formfield="applicantFirstName" onkeyup={searchApplicantHandler} label="Enter First Name"></lightning-input> <br />

       <table aria-describedby="conatact-list" class="slds-table  slds-table\_bordered ">

        <!--Header of the table-->

        <thead>

            <tr class="slds-line-height\_reset">

                <th class="slds-size\_1-of-6 " scope="col">

                    <div class="slds-truncate " title="First Name">First Name</div>

                </th>

           </tr>

        </thead>

        <!--Body of the table-->

        <tbody>

               <template for:each={applicantList} for:item="objApp">

                    <tr key={objApp.id}>

                          <td>

                                {objApp.First\_Name\_\_c}

                          </td>

                    </tr>

               </template>

        </tbody>

    </table>

    </lightning-card>

</template>

import { LightningElement } from 'lwc';

import searchApplicantByAlphabets from '@salesforce/apex/ApplicantProvider.searchApplicantByAlphabets';

export default class ApplicantRecordsDateRange extends LightningElement {

    firstName;

    applicantList;

   searchApplicantHandler(){

        this.firstName = this.template.querySelector('lightning-input[data-formfield="applicantFirstName"]').value;

        console.log(this.firstName);

        searchApplicantByAlphabets({ firstName: this.firstName})

        .then((result)=>{

            console.log(JSON.stringify(result));

            this.applicantList = result;

        })

        .catch((error) =>{

            console.log(JSON.stringify(error));

        });

    }

}

public with sharing class ApplicantProvider {

    @AuraEnabled

    public static string createNewApplicantRecord(Applicant\_\_c objApplicant){

        try {

            if(objApplicant != null ){

                    insert objApplicant;

                    return 'Applicant Record Created Successfully...!!!';

            }

            else{

                    return 'Locha, something went wrong...!!!';

            }

        } catch (Exception e) {

            throw new AuraHandledException(e.getMessage());

        }

    }

   @AuraEnabled

   public static Applicant\_\_c searchApplicantRecord(Applicant\_\_c objApp){

    try {

            if(objApp != null){

             return  ([select Id, First\_Name\_\_c, Last\_Name\_\_c, Gender\_\_c from Applicant\_\_c where Name=: objApp.Name LIMIT 1] );

            }

            else{

                return null;

            }

       } catch (Exception e) {

           throw new AuraHandledException(e.getMessage());

       }

   }

   @AuraEnabled

   public static List<Applicant\_\_c> searchApplicantDateRange(Date fromDate,Date toDate){

    try {

        if(fromDate!=null && toDate !=null){

            return [select Id,Name, First\_Name\_\_c, CreatedDate from Applicant\_\_c where CreatedDate>=:fromDate and CreatedDate <=: toDate];

        }

        else{

            return null;

        }

    } catch (Exception e) {

        throw new AuraHandledException(e.getMessage());

    }

   }

   @AuraEnabled

   public static List<Applicant\_\_c> searchApplicantByAlphabets(String firstName){

    try {

        String myFirstName = firstName+'%';

        if( !String.isBlank(firstName)){

        return [select Id, Name, First\_Name\_\_c from Applicant\_\_c where First\_Name\_\_c like : myFirstName ];

        }

        else{

            return null;

        }

    } catch (Exception e) {

        throw new AuraHandledException(e.getMessage());

    }

   }

}

Aise bhi kar sakte Ho…!!!!

 if( !String.isBlank(firstName)){

        return [select Id, Name, First\_Name\_\_c from Applicant\_\_c where First\_Name\_\_c like : firstName+'%' ];

        }

        else{

            return null;

        }

===============