53. LWC Bundle Structure, Naming Conventions,

JavaScript Properties and Data Bindings - 20 June 2022

Today we will learn:

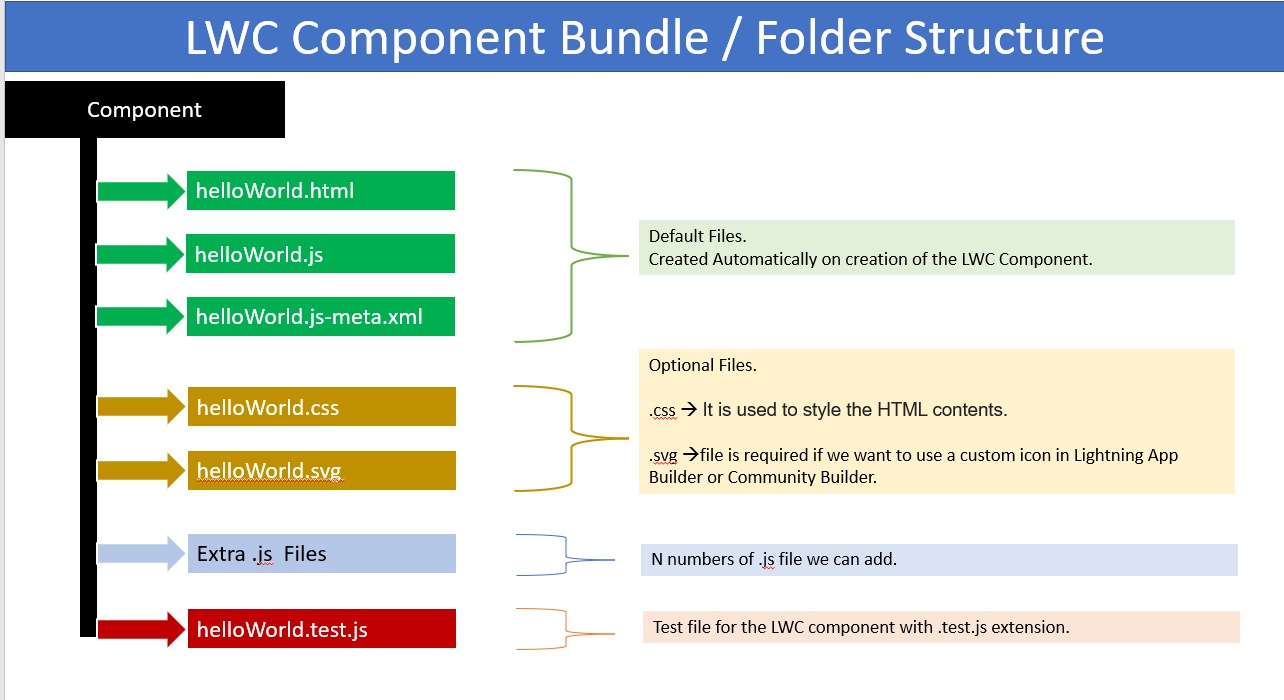
1] LWC Bundle / Folder Structure

2] Naming Conventions

3] JavaScript Properties/Variables/Method

4] Data Binding (One Way)

**1] LWC Bundle/Folder Structure**



* **HTML File**

Every UI component must have an HTML file with root tag “template” so if you want to have any UI element like headings or paragraph we need to place inside these template tags.

#### ****Javascript File****

Every LWC component must have a JavaScript file. The JavaScript file defines the behaviour of HTML elements present in HTML file.

#### ****Configuration File – XML file****

Configuration file defines the metadata values for the component. For example, the design configurations for lightning app builder or a community builder, so if you want to use your component inside the app page or record page or homepage or community then you need to specify those configuration here inside the meta file.

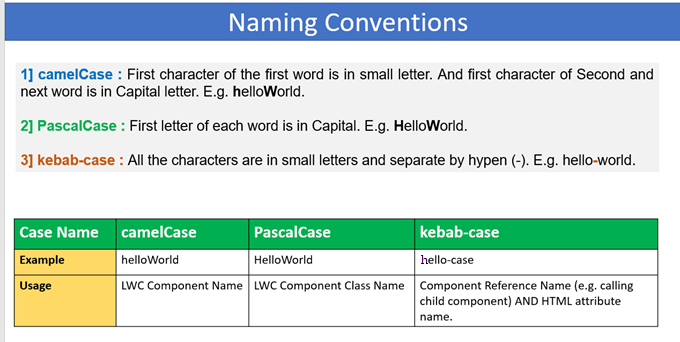
#### ****CSS fil****e

CSS file is not present by default at the time of component creation, but we can create it as an when required.

#### ****SVG File****

SVG file is not present by default at the time of component creation, but we can create it as an when required. SVG file is required if we want to use a custom icon in Lightning App Builder or Community Builder.

**2] Naming Conventions**





**Camel Case 🡺 Variable / Method Name**

**HelloWorld**

Snake Case 🡪hello\_world 🡪 It is used for variable/ Constant Declaration.

|  |  |  |
| --- | --- | --- |
| **camelCase** | **PascalCase** | **kebab-case** |
| Model Mill | Model Mill | Modal Mill |
| modelMill | ModelMill | modal-mill |

Parent Component 🡪 **modelComponent**

Child Component 🡪 **carComponent**

**modelComponent START**

**<c- car-component /> 🡪 kebab-case**

**modelComponent END**

Parent Component 🡪 aatiKyaKandala

Child Component 🡪 nahiAaungi

aatiKyaKandala START

<c-nahi-aaungi /> 🡺 kebab-case

aatiKyaKandala END

Parent Component 🡪 model

Child Component 🡪 car

model START

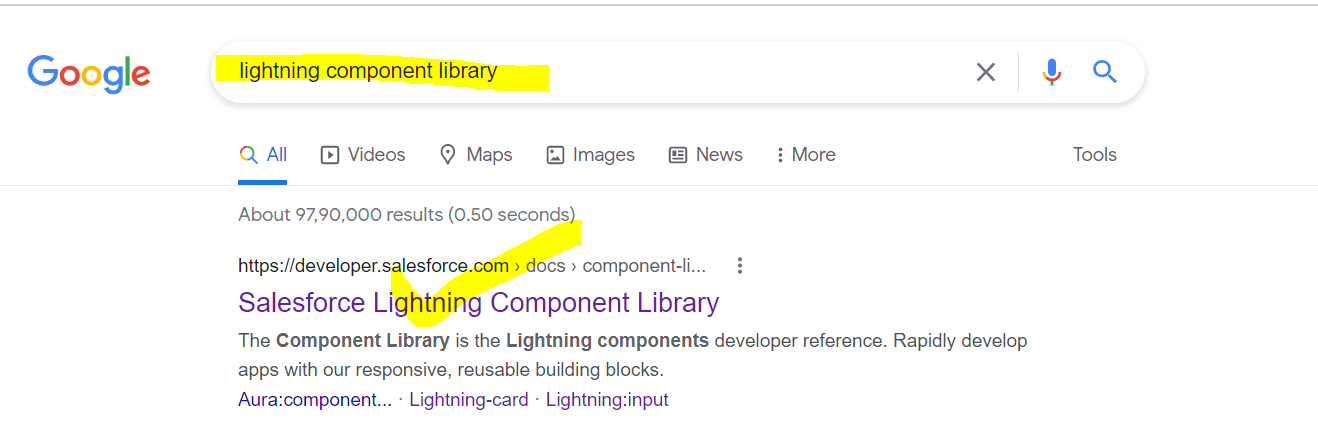
<c-car />

model END

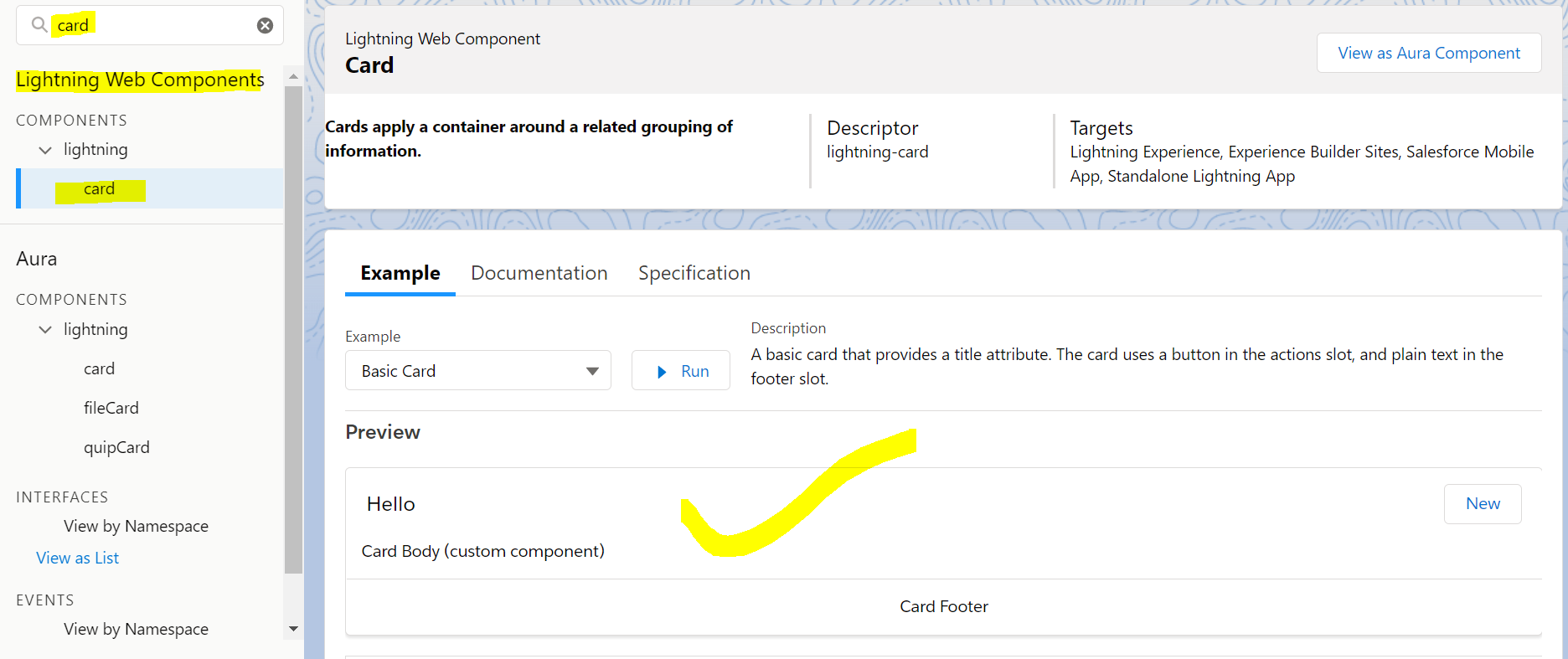
================================================================================

**3] JavaScript Properties (Variables / Method)**

- Undefined , String, Number, Decimal, Boolean, Date, Object, Array/List



<https://developer.salesforce.com/docs/component-library>



<template>

    <lightning-card  title="Hello">

        Silver Microsystems...!!!! <br />

        Name = {name} <br/>

        My Name is = {myName} <br />

        My Number = {myNumber} <br />

        My Salary is = {mySalary} <br/>

        Flag = {flag} <br />

        Current Date = {currentDate} <br />

        My Account Name is =  {objAccount.Name} <br/>

                Rating = {objAccount.Rating} <br />

                SLA = {objAccount.SLA\_\_c} <br />

        Employee List = {empList}

    </lightning-card>

</template>

import SystemModstamp from '@salesforce/schema/Account.SystemModstamp';

import { LightningElement } from 'lwc';

export default class HelloWorld extends LightningElement {

    name //Undefined

    myName = 'Ganesh Padole' //String

    myNumber = 420 //Integer

    mySalary = 4000.67 // Decimal

    flag = true //Boolean

    currentDate = new Date() //Current Date

    objAcc = {'sObjectType' : 'Account'} //Standard Object

    objApplicant = {'sObjectType' : 'Applicant\_\_c'} //Custom Object

    objAccount = {

        Name : 'Cinemax',

        Rating : 'Hot',

        Type : 'Prospect',

        SLA\_\_c : 'Gold'

    }

    empList = ["Ram" , "Sita", "Laxman"];

//Method

    myMethod(){

    }

}

<?xml version="1.0" encoding="UTF-8"?>

<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">

    <apiVersion>54.0</apiVersion>

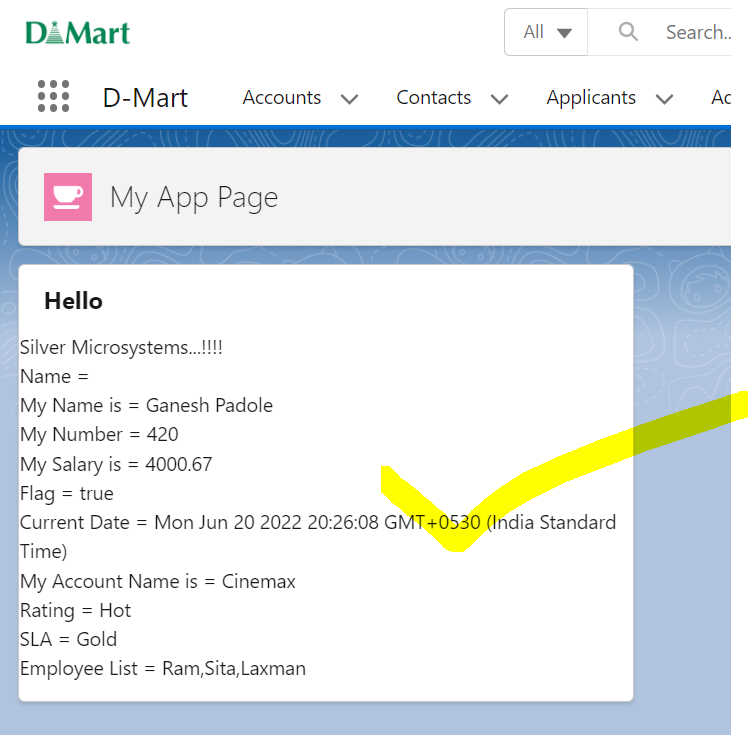
    <isExposed>true</isExposed>

    <targets>

        <target>lightning\_\_AppPage</target>

    </targets>

</LightningComponentBundle>

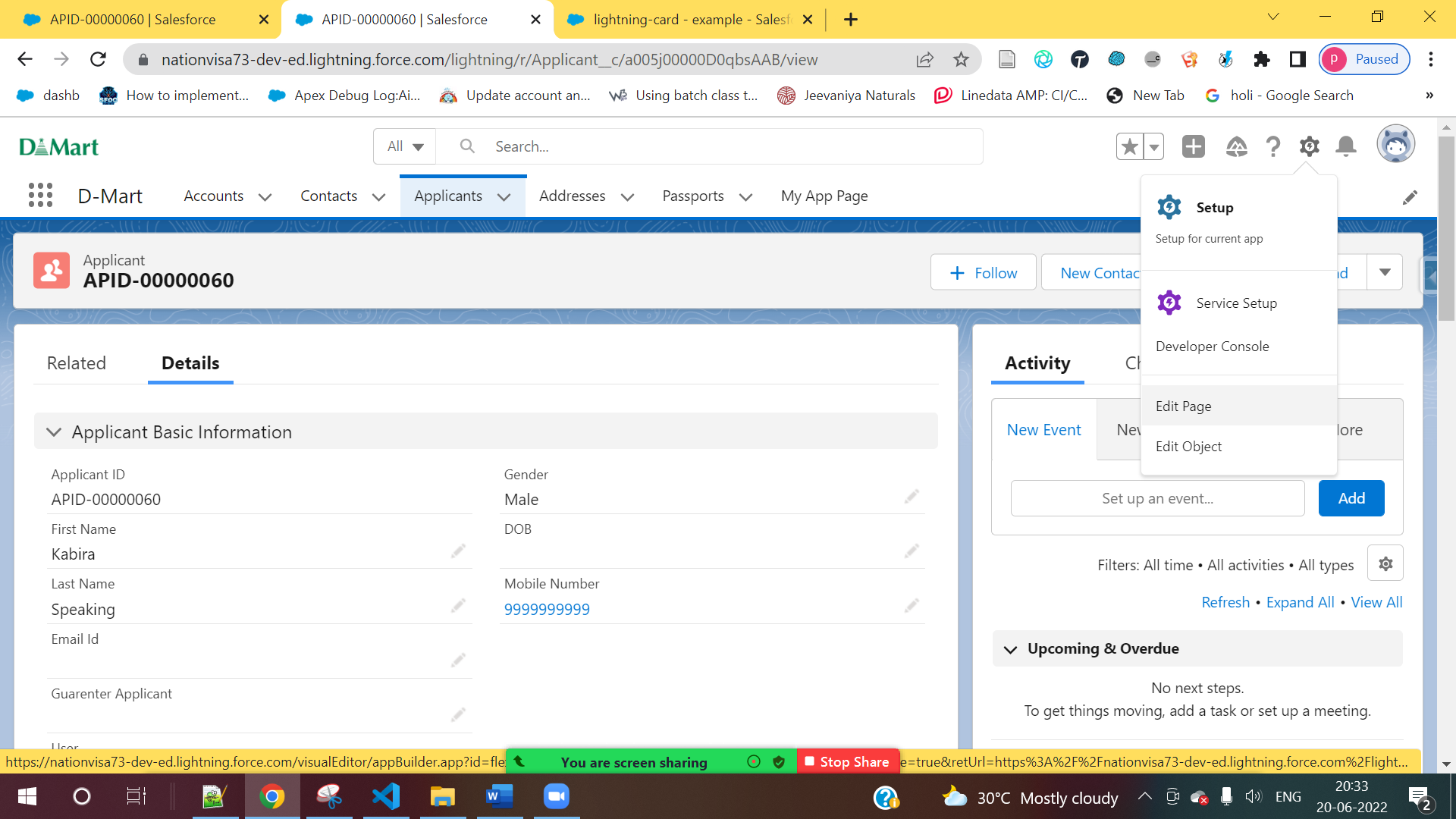


Usage of LWC Component

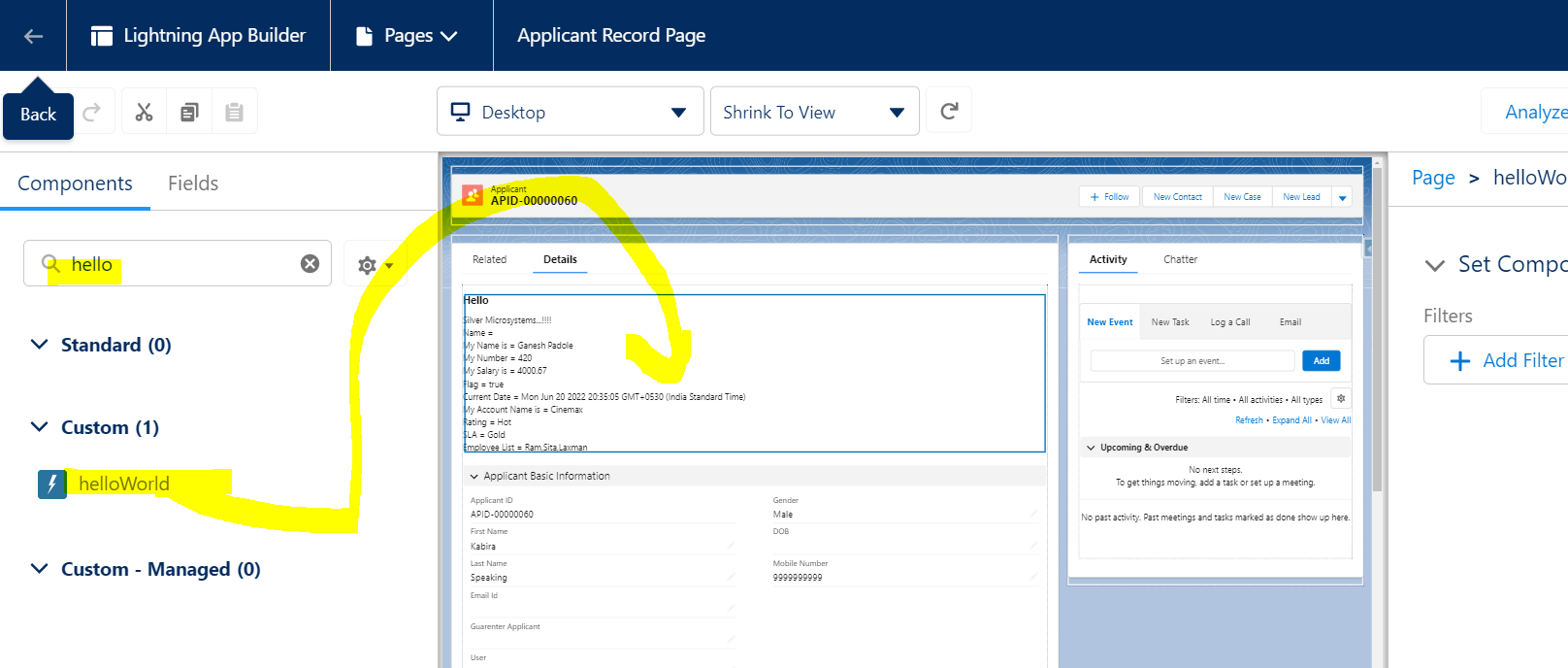
* App Page
* Details Page
* Utility Bar
* Home Page
* Quick Action Button
* Tab
* Community

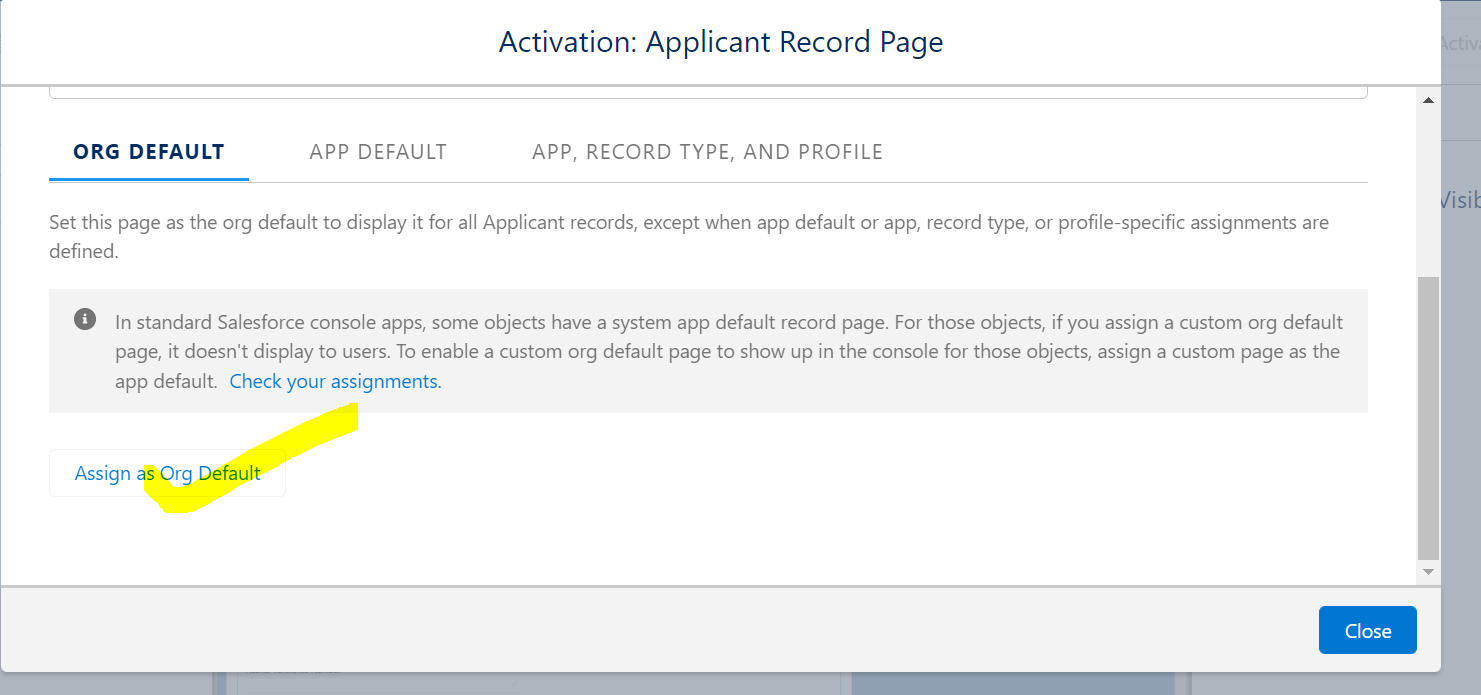
Details Page

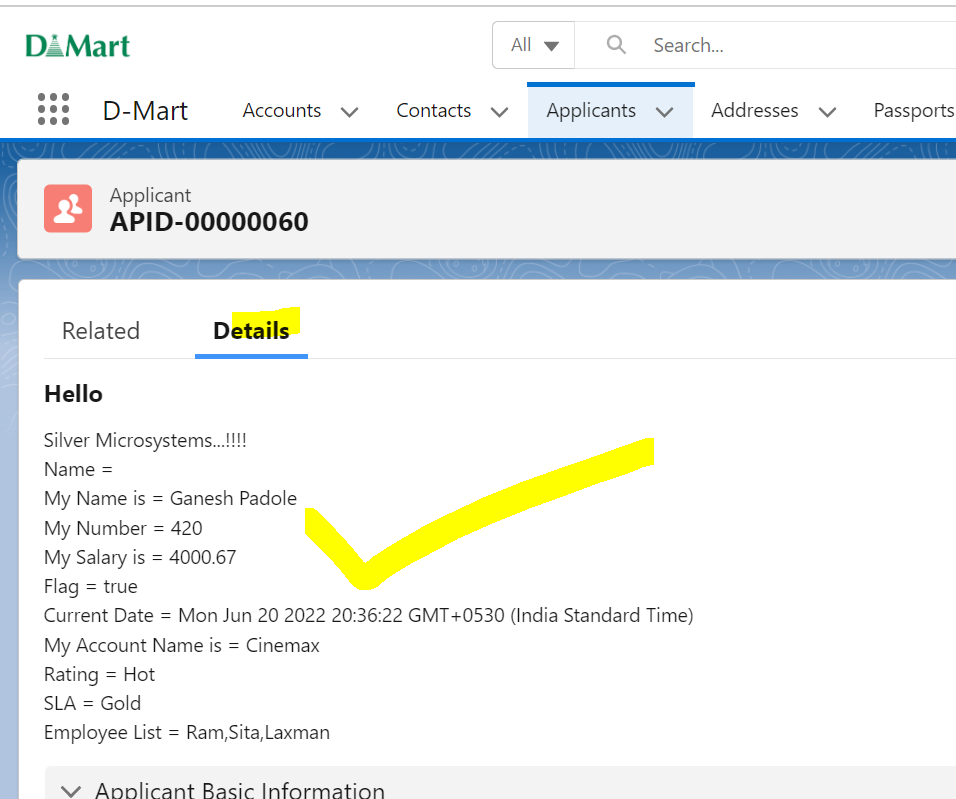
 <target>lightning\_\_RecordPage</target>





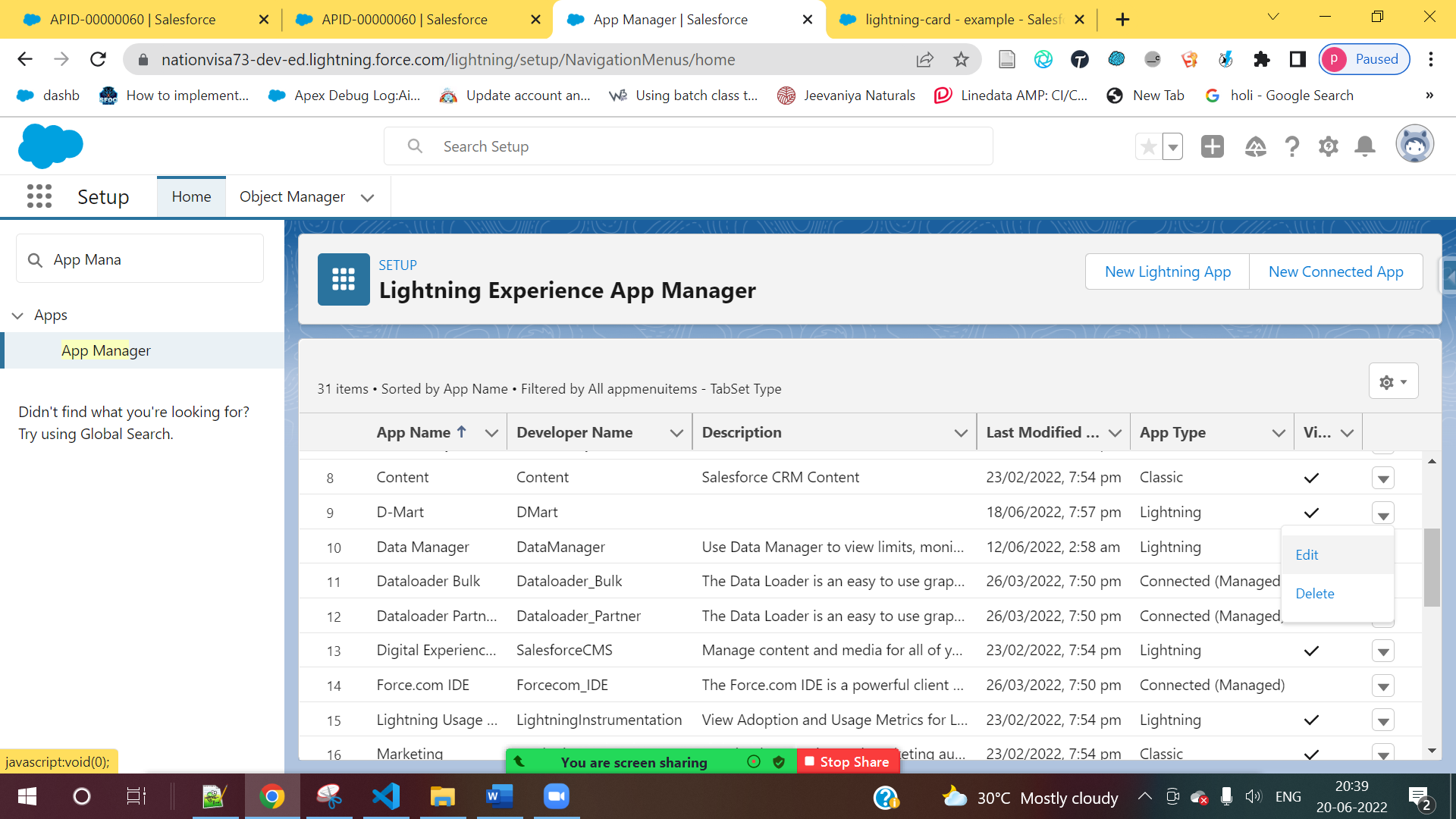


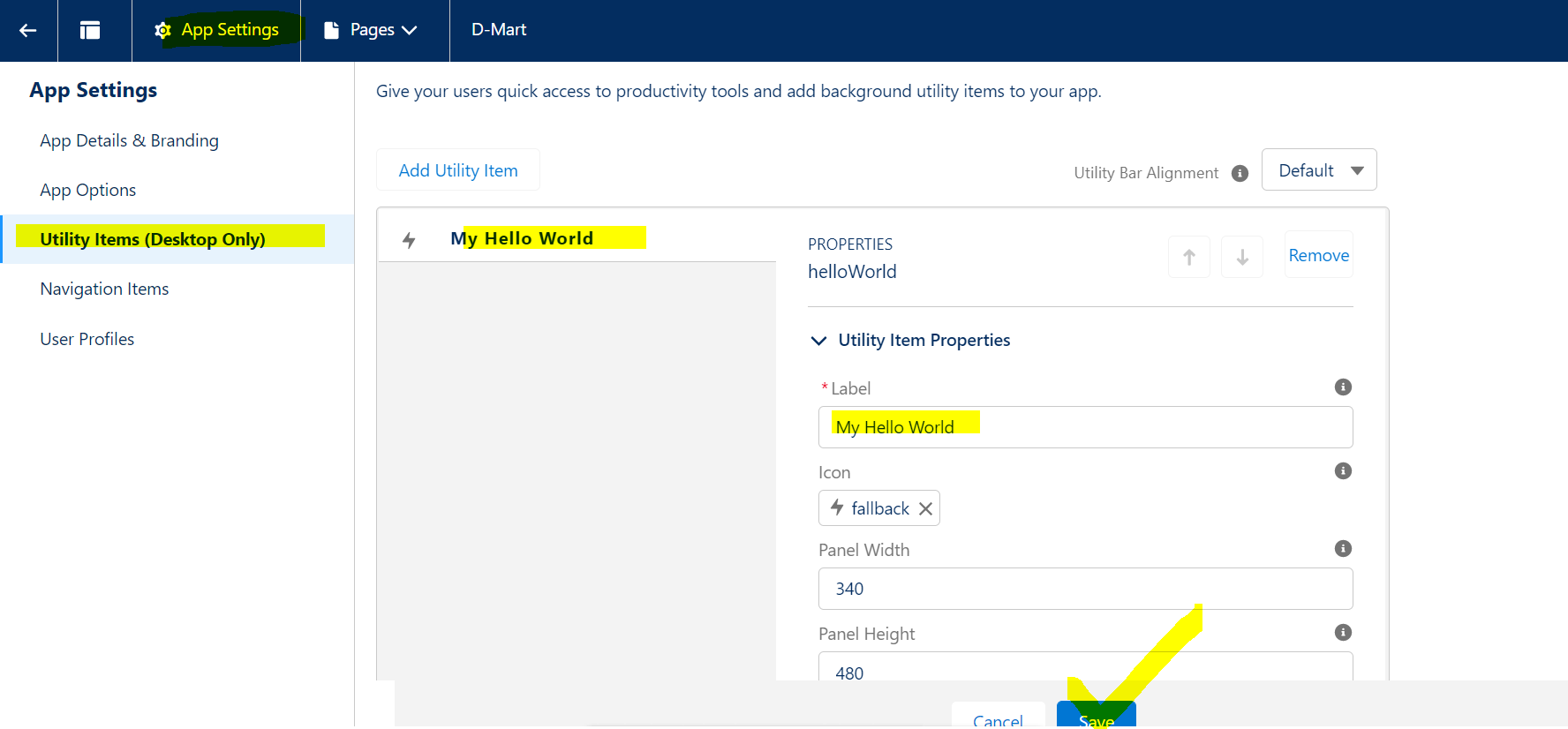


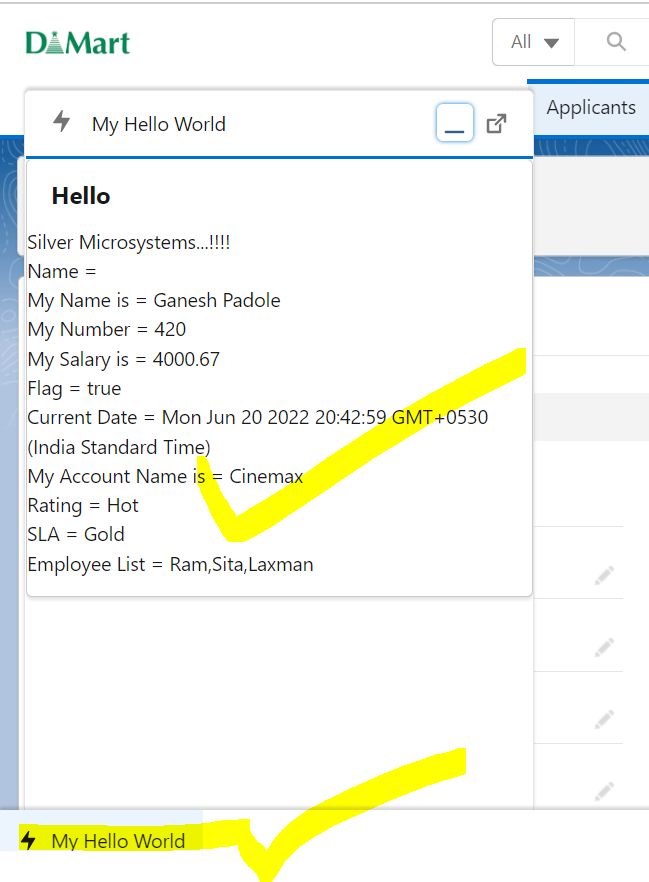


Utility Bar

  <target>lightning\_\_UtilityBar</target>

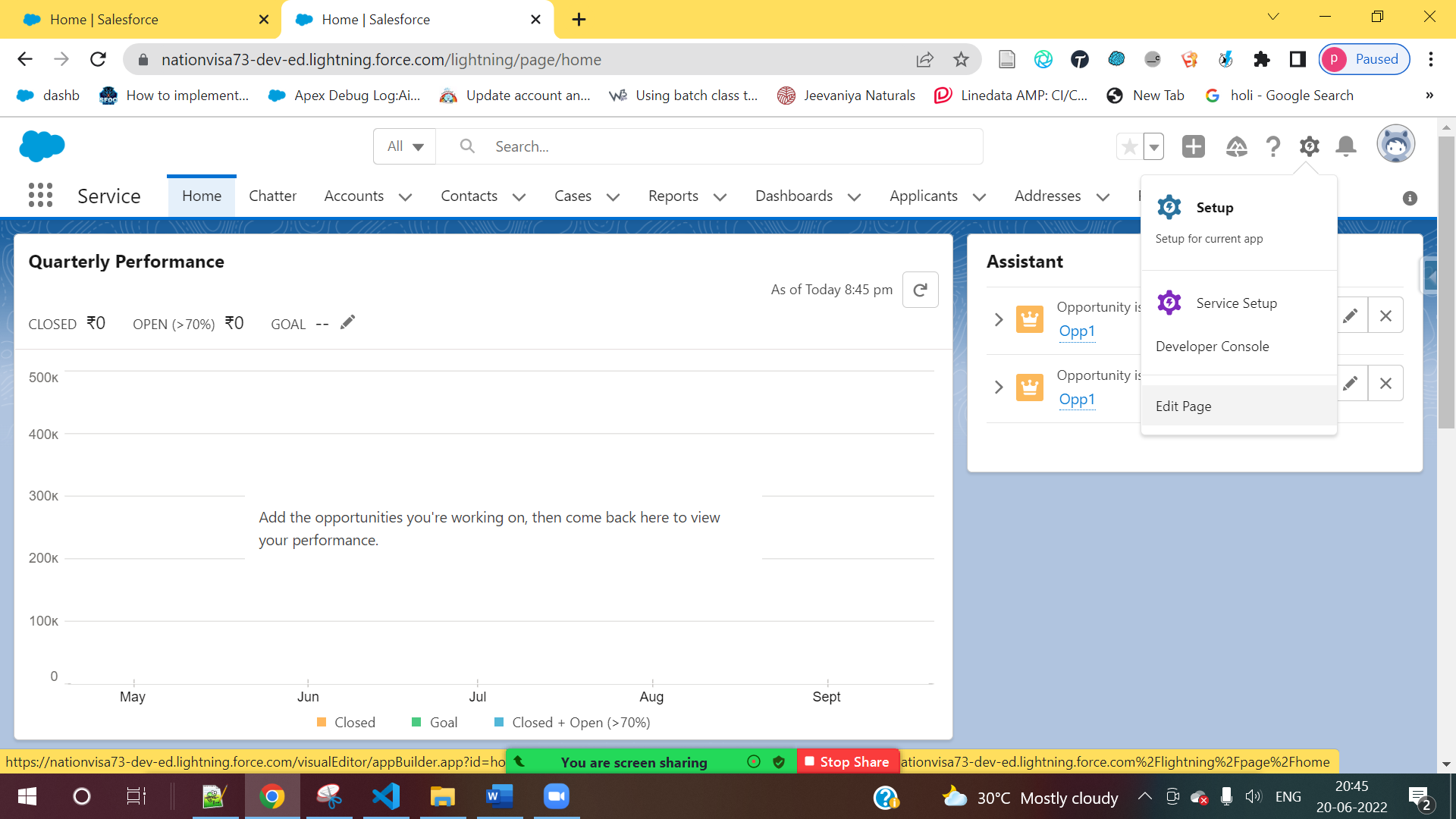


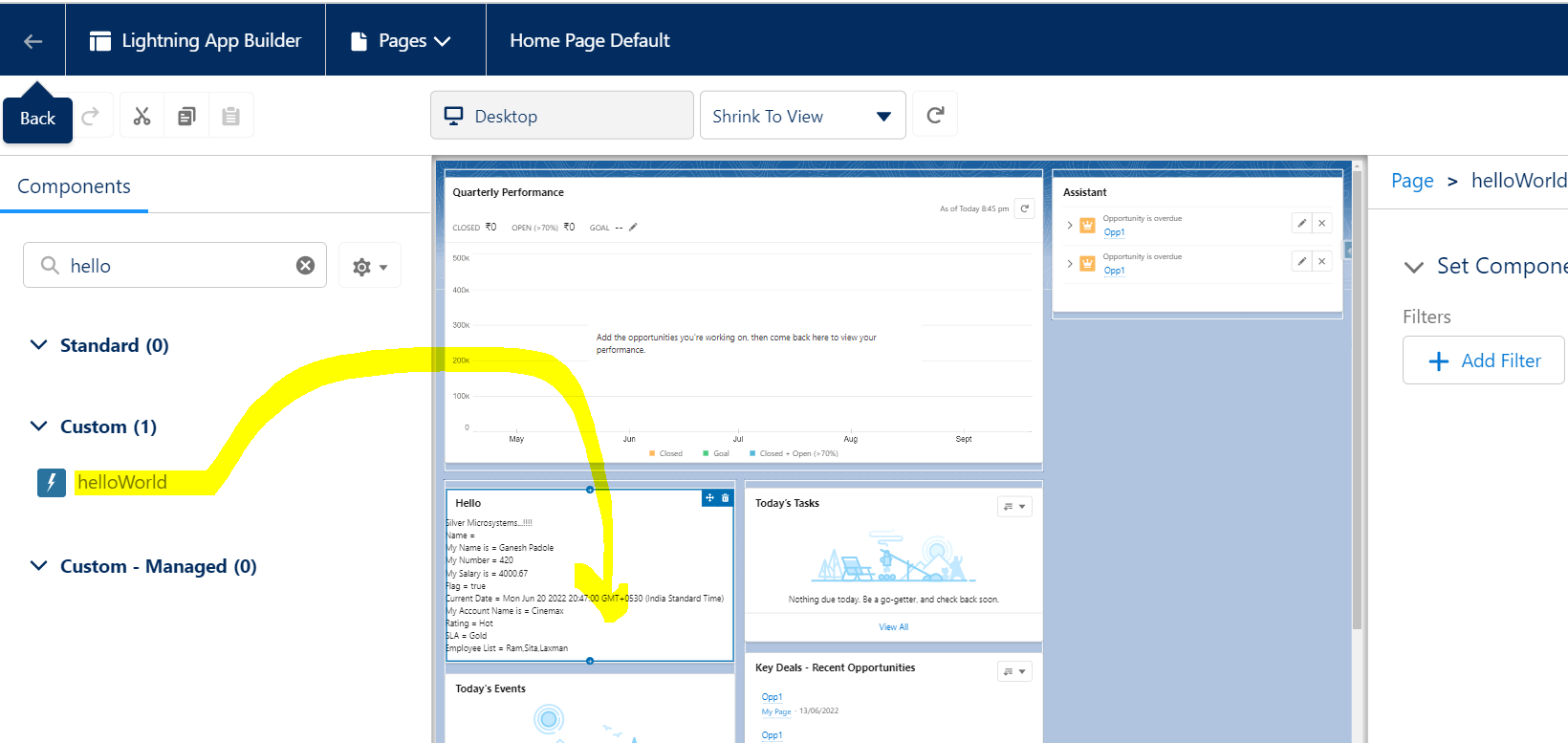


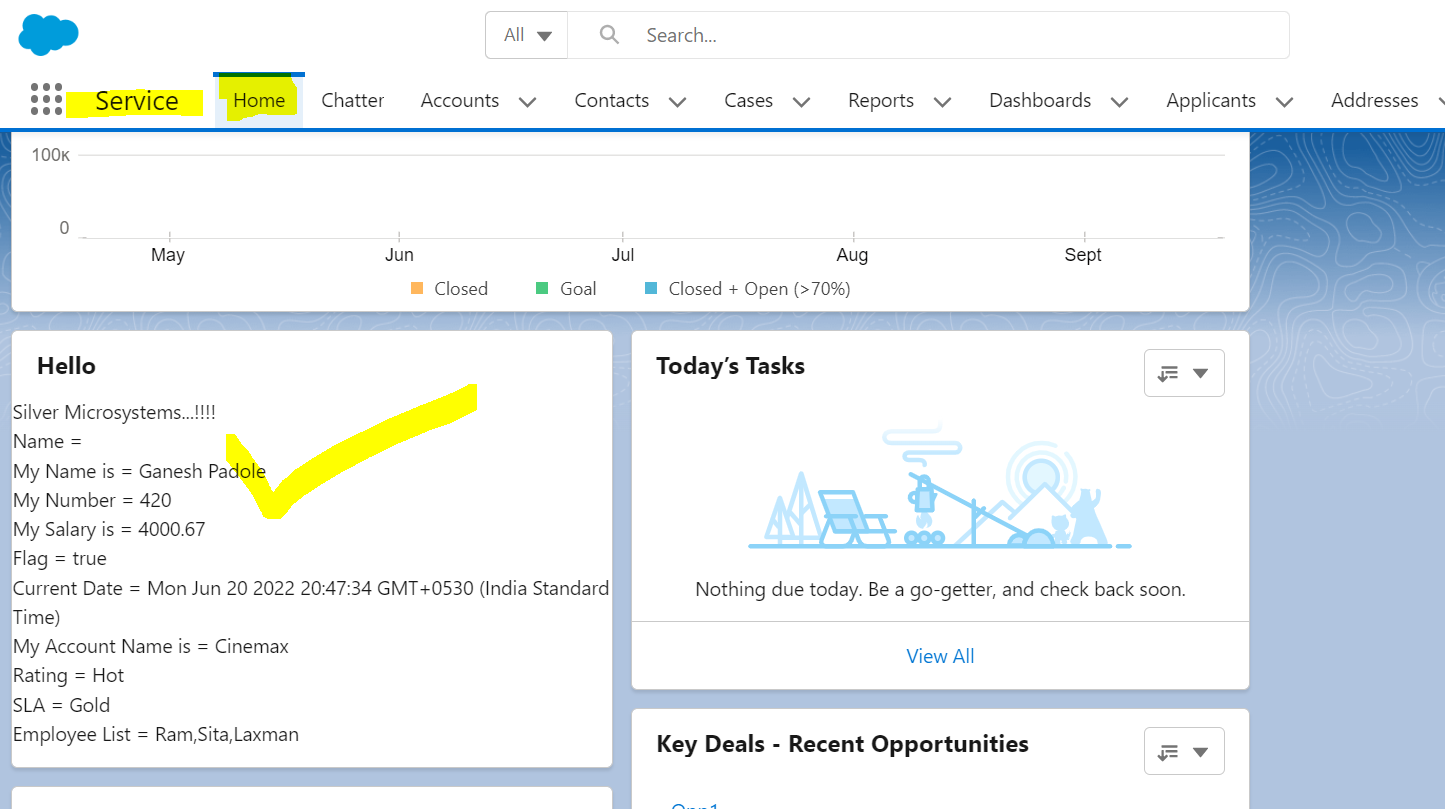


Home Page

<target>lightning\_\_HomePage</target>

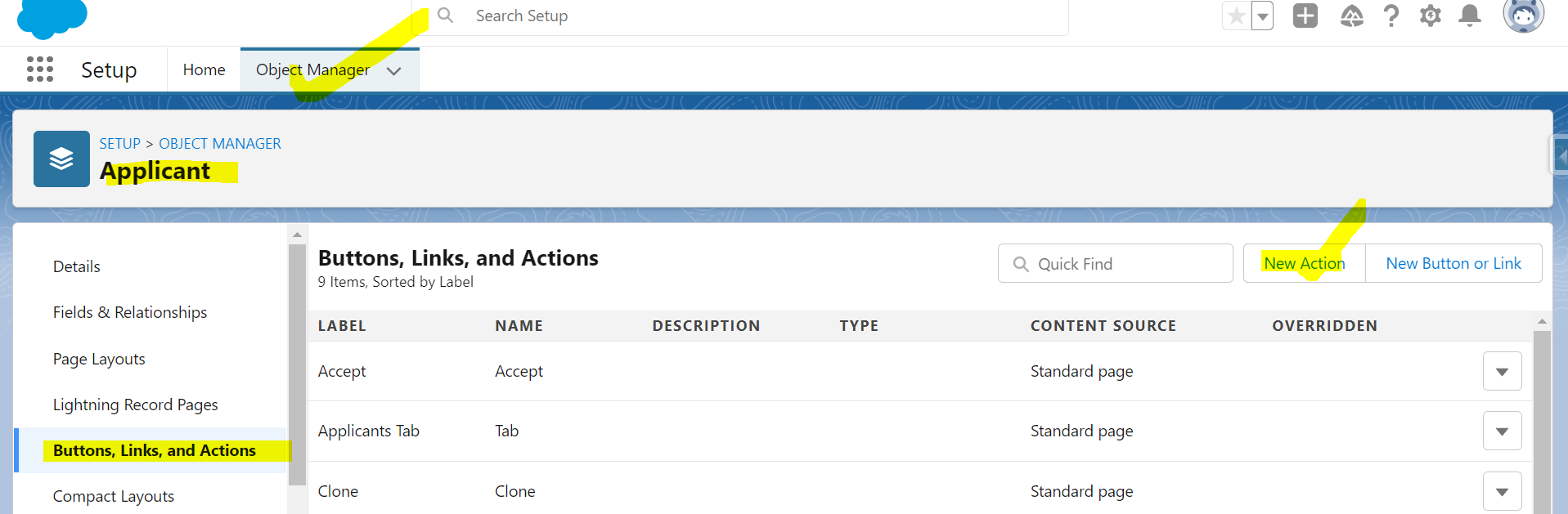


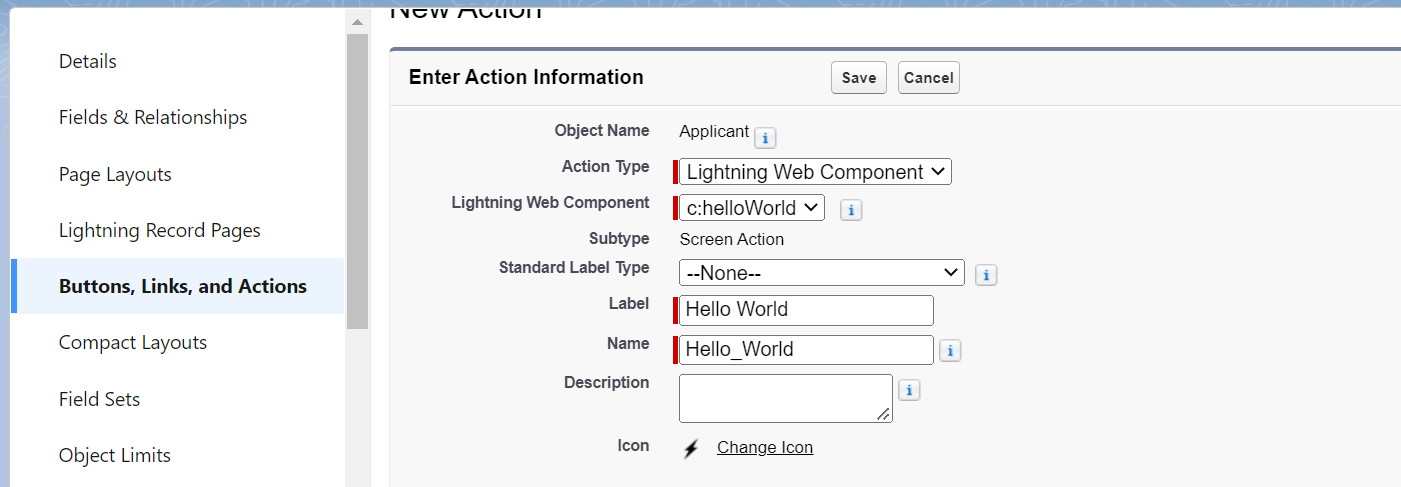


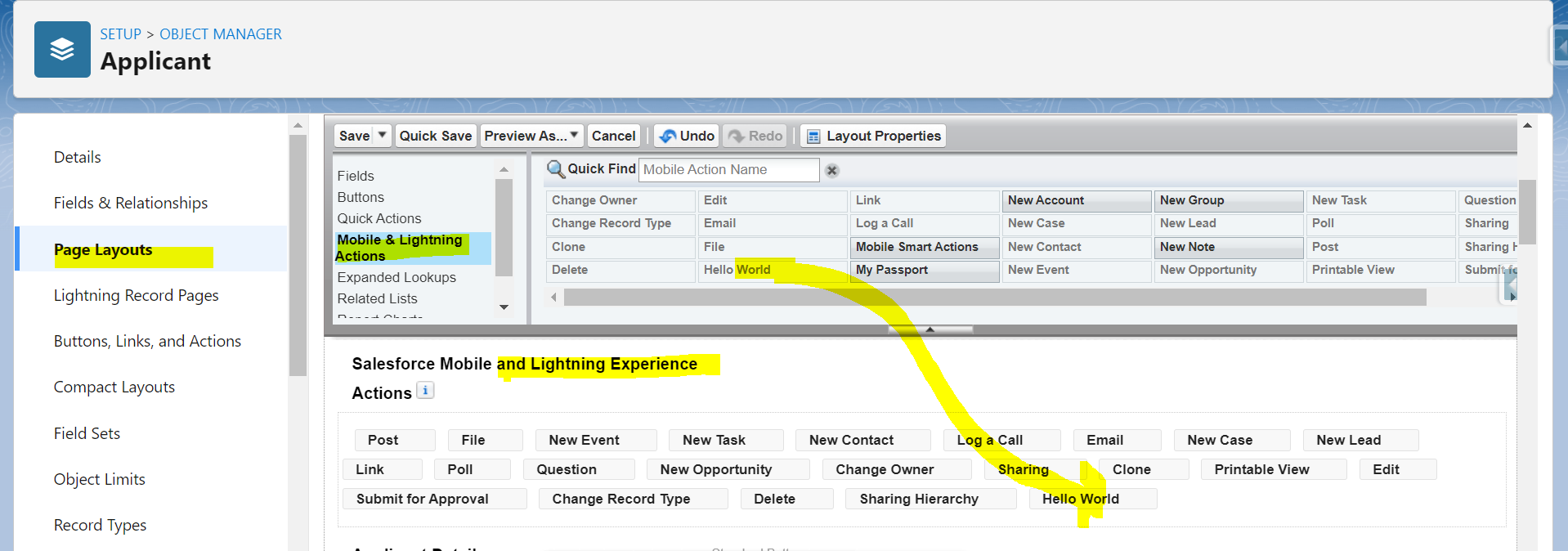


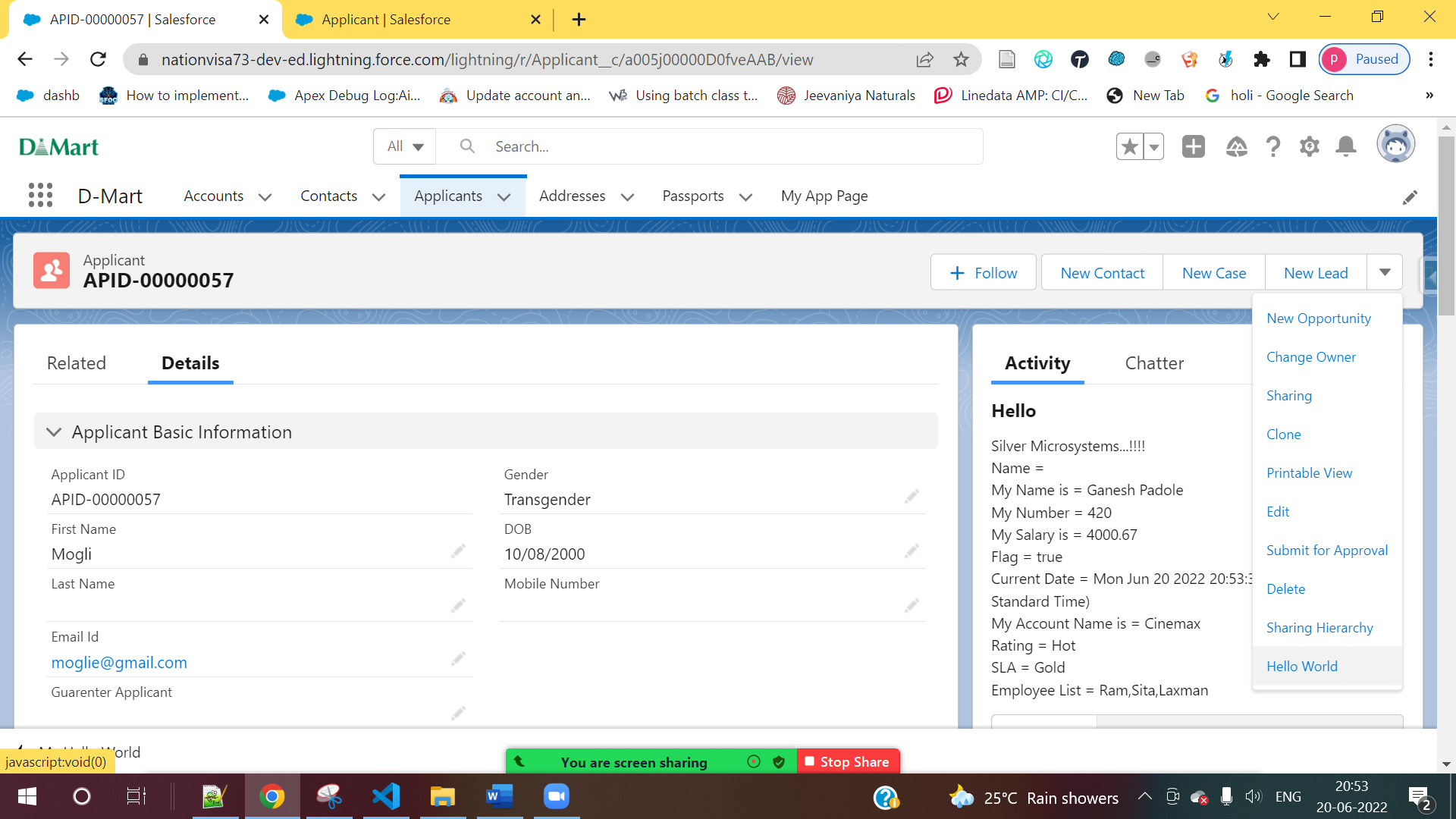
Quick Action Button

  <target>lightning\_\_RecordAction</target>

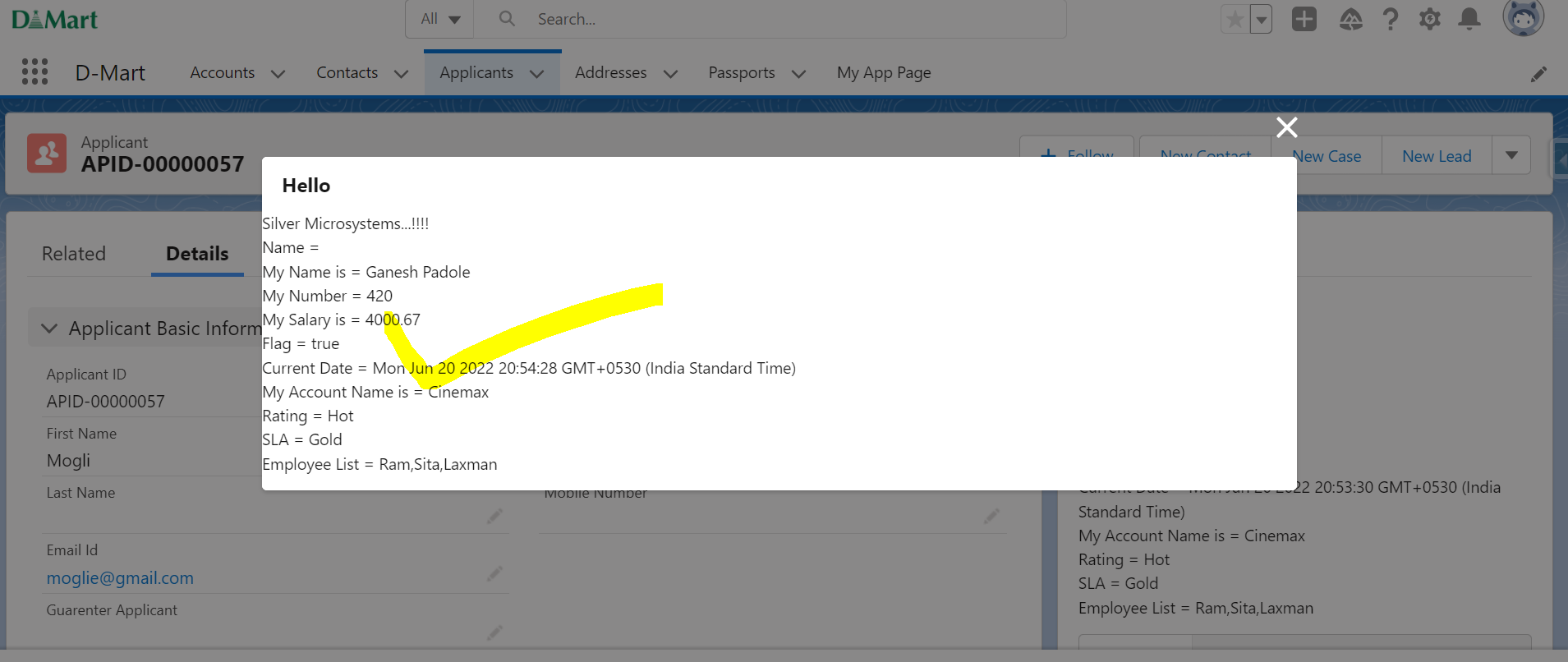






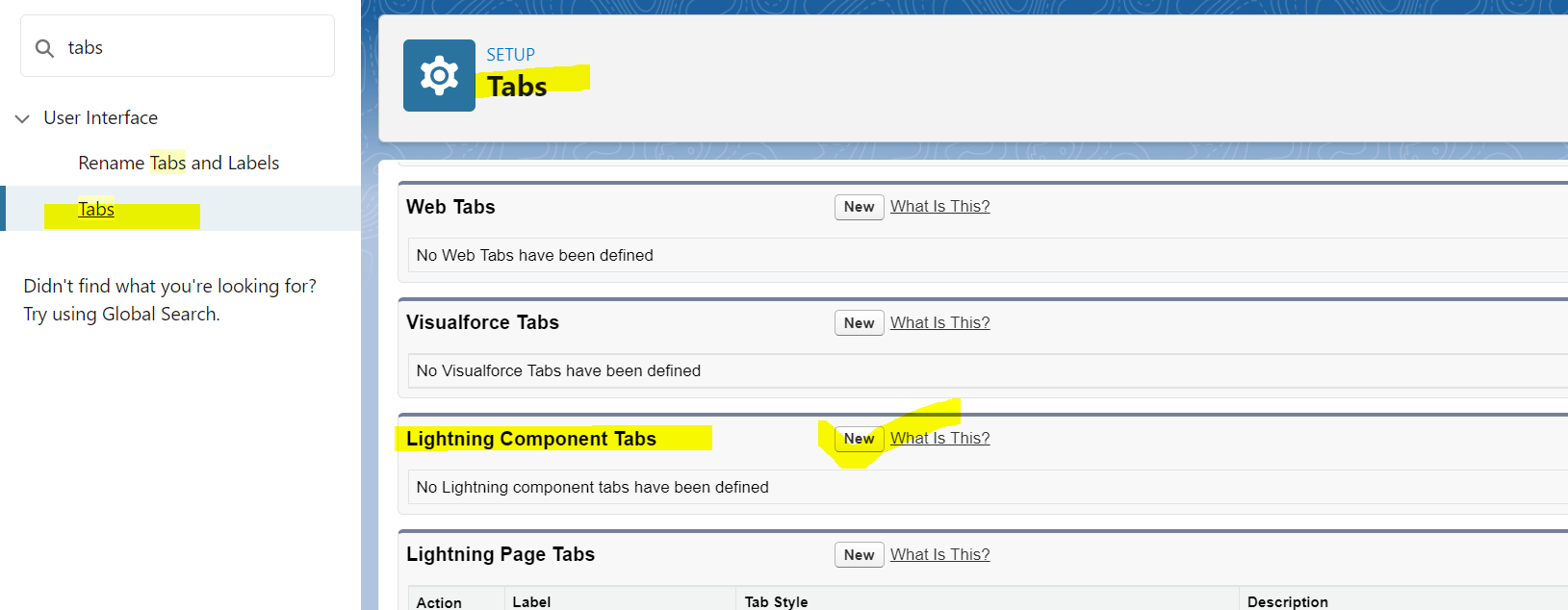


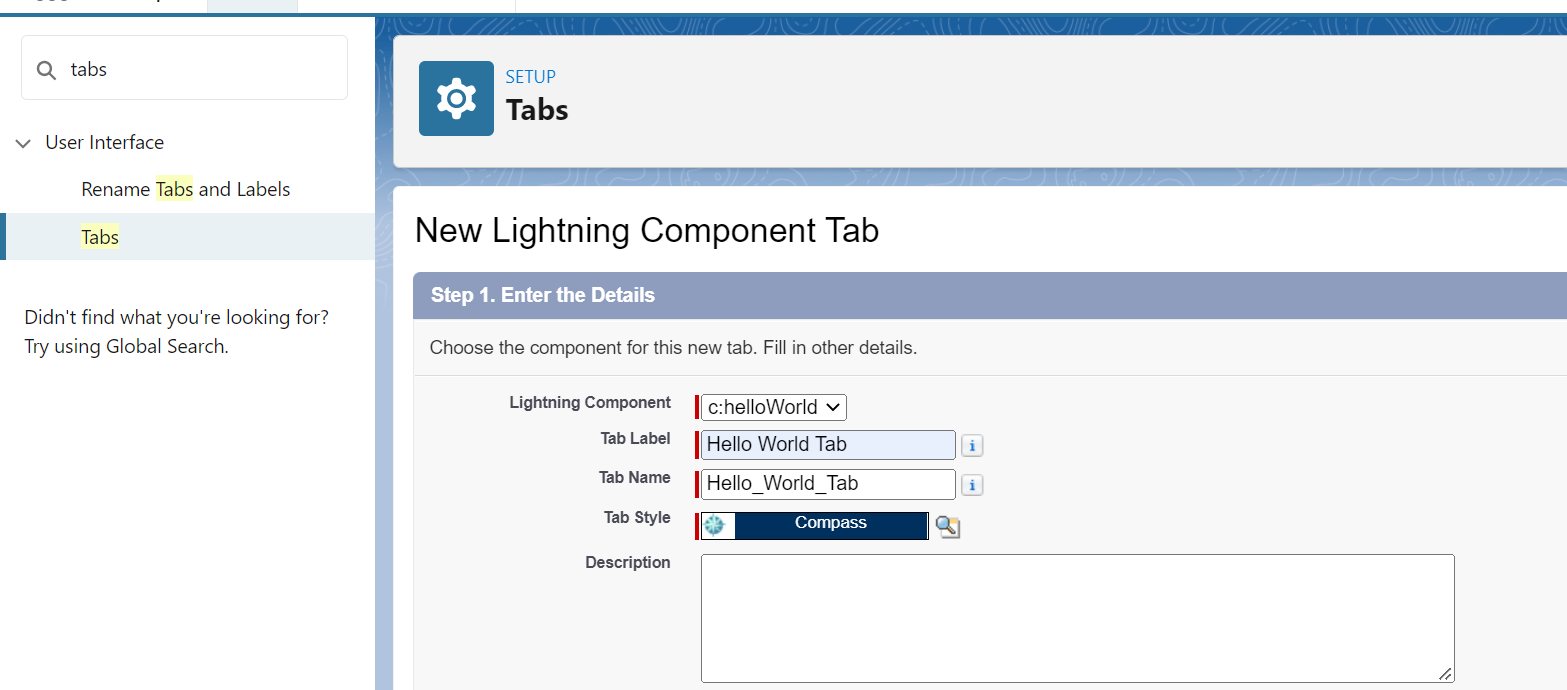


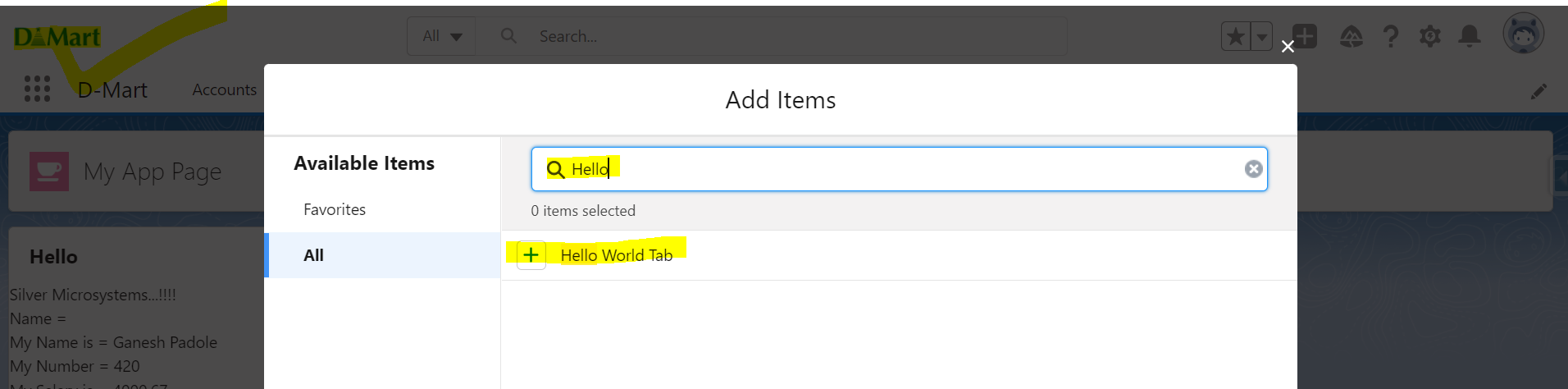


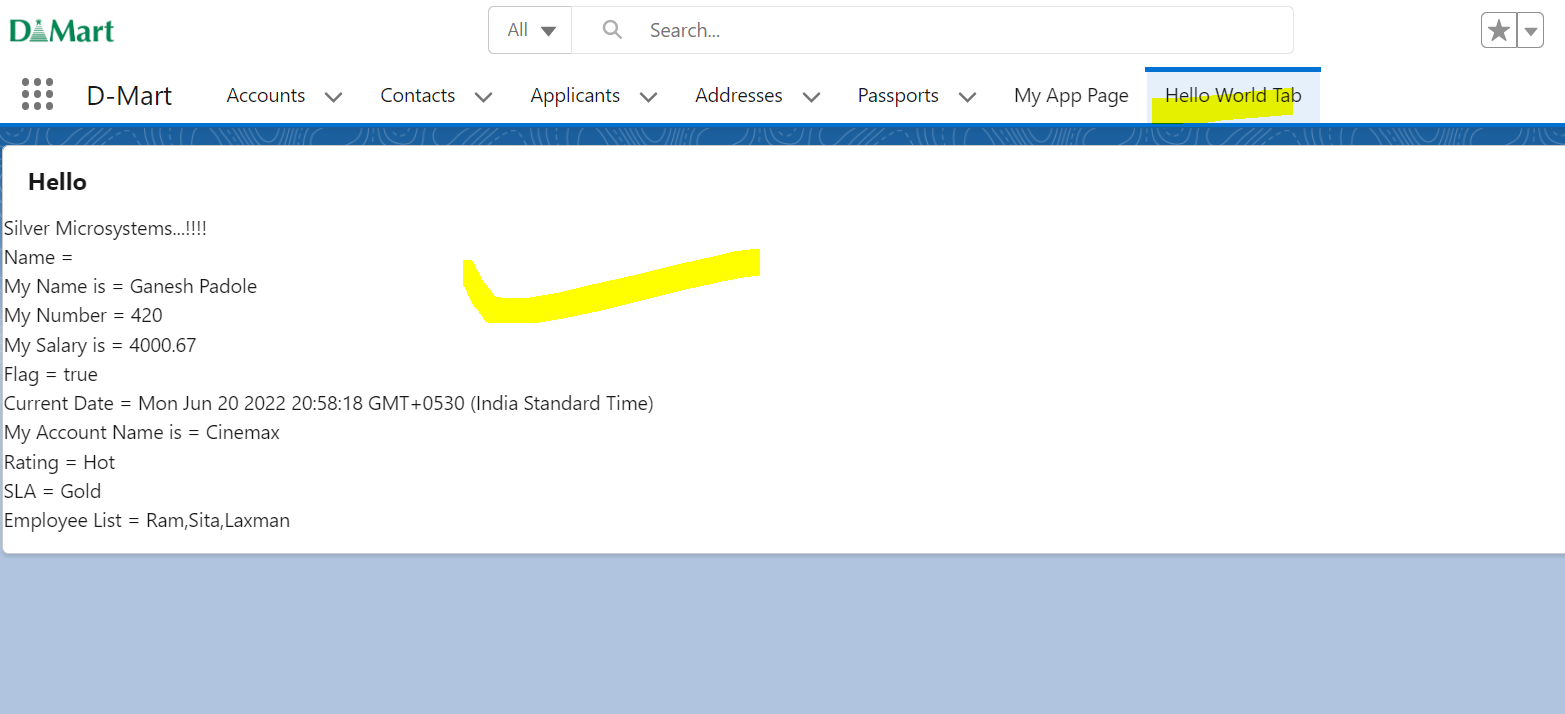
Tab

  <target>lightning\_\_Tab</target>

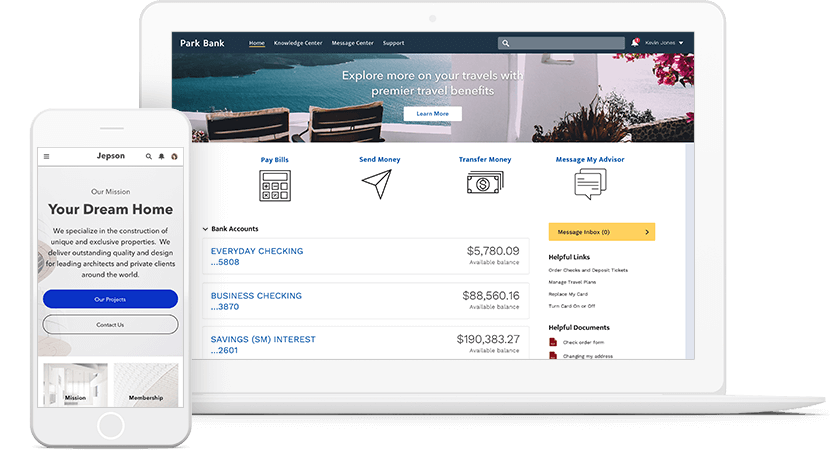




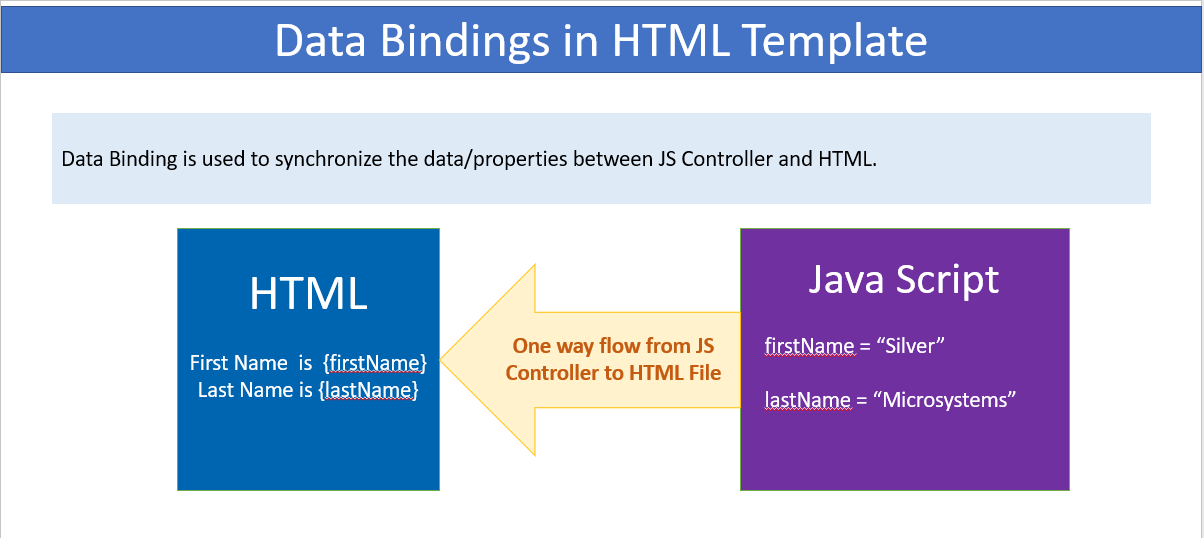




Community



**4] Data Bindings (One Way)**



**IMPORTANT POINTS TO REMEMBER:**

1] Do not add space around the property. Example **{ firstName }**

2] We can access primitive or object property directly in HTML Template.

3] Computed expression is not allowed. Example **{10+25}** or employeeList**[0]**.

4] Use **Dot** when to access object property.

5] Property in **{ }** must be available in JS Controller. Example { **firstName**}, then firstName should be exist

in JS Controller

|  |  |
| --- | --- |
| **Create an Object** | **Account Record** |
| objAcc = {'sObjectType' : 'Account'} | objAccount = {          Name : 'Cinemax',          Rating : 'Hot',          Type : 'Prospect',          SLA\_\_c : 'Gold'      } |
| Account objAcc = new Account(); | Account objAccount = new Account(Name=’Cinemax’, Rating=’Hot’); |

.