

Naan Mudhalvan
Salesforce Developer(Course)
Assignment no 1

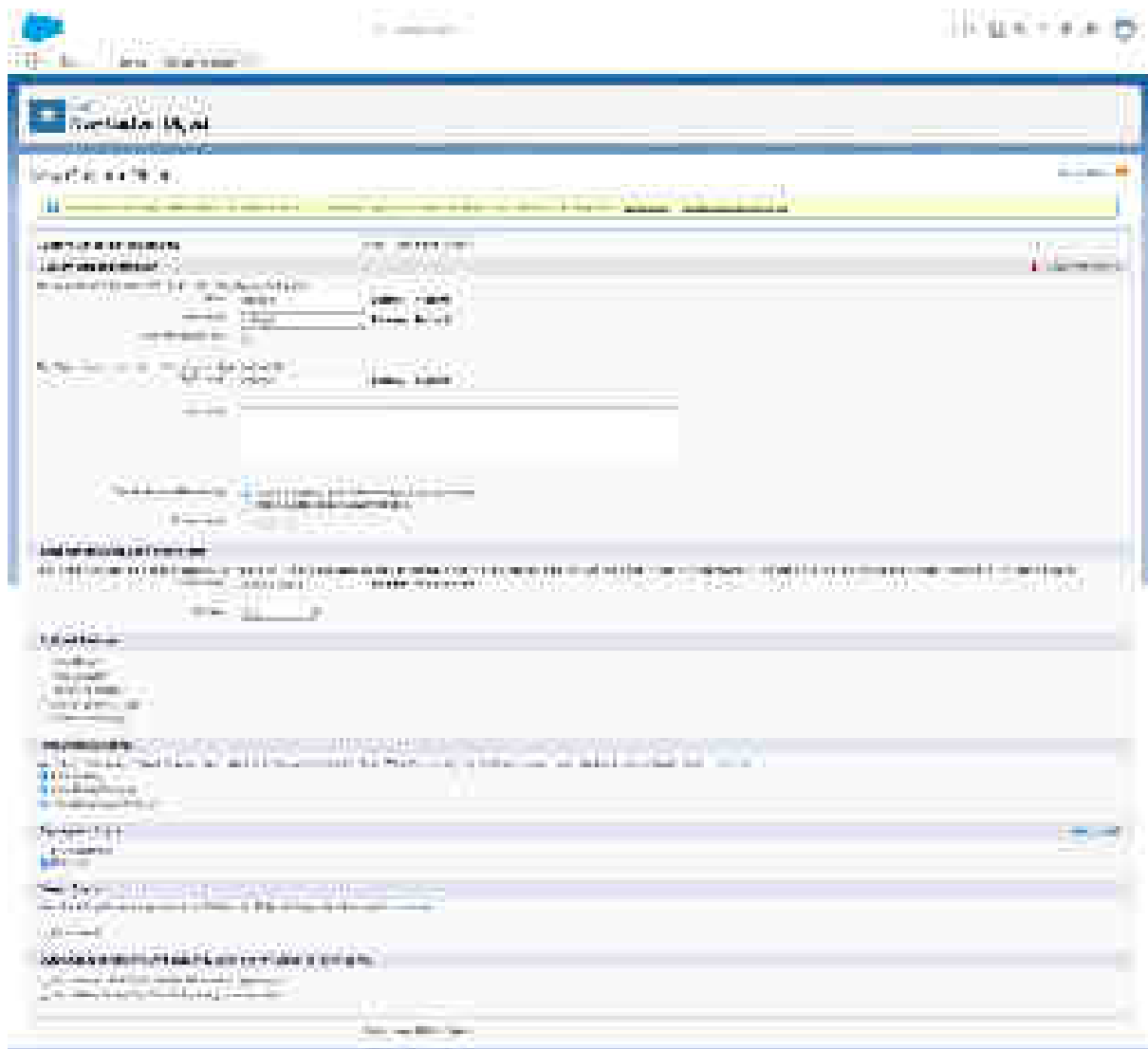
Name : Devadharshini C
Naan Mudhalvan id : au611420104015
Year & Dep : 4th year & CSE
Batch : 2024
Zone no : Zone 8

1. Create a Master-Detail Relationship between two Custom objects and also create a Roll Up Summary Field to Calculate total number of records.

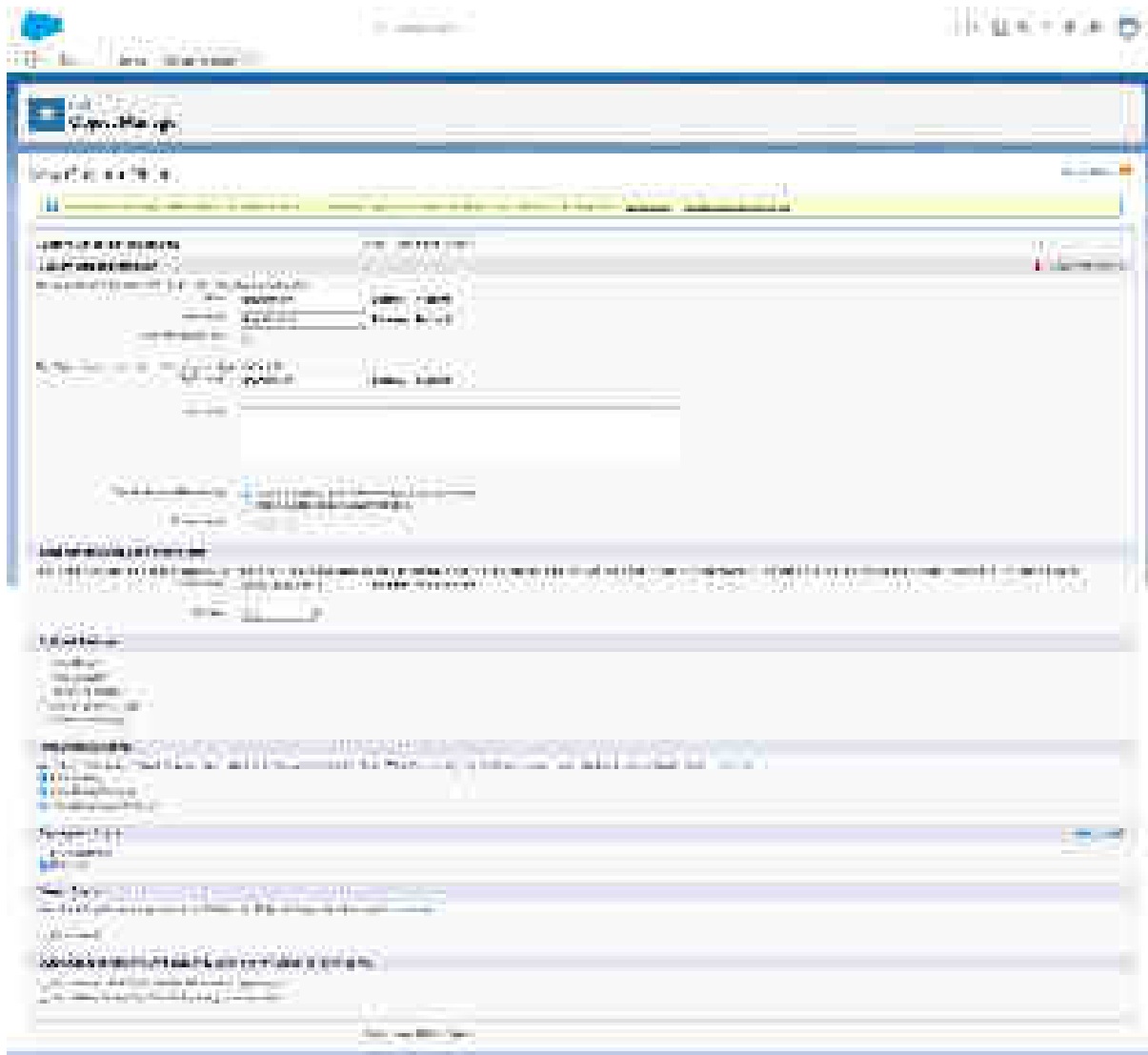
Solution:

Step 1: Create Custom Objects

Assuming you have two custom objects, let's call them "College_C" and "C Department_C". If you haven't already created these objects, you can do so by going to Setup > Object Manager > Create > Custom Object.



Second custom objects, let's call them "Department_C"

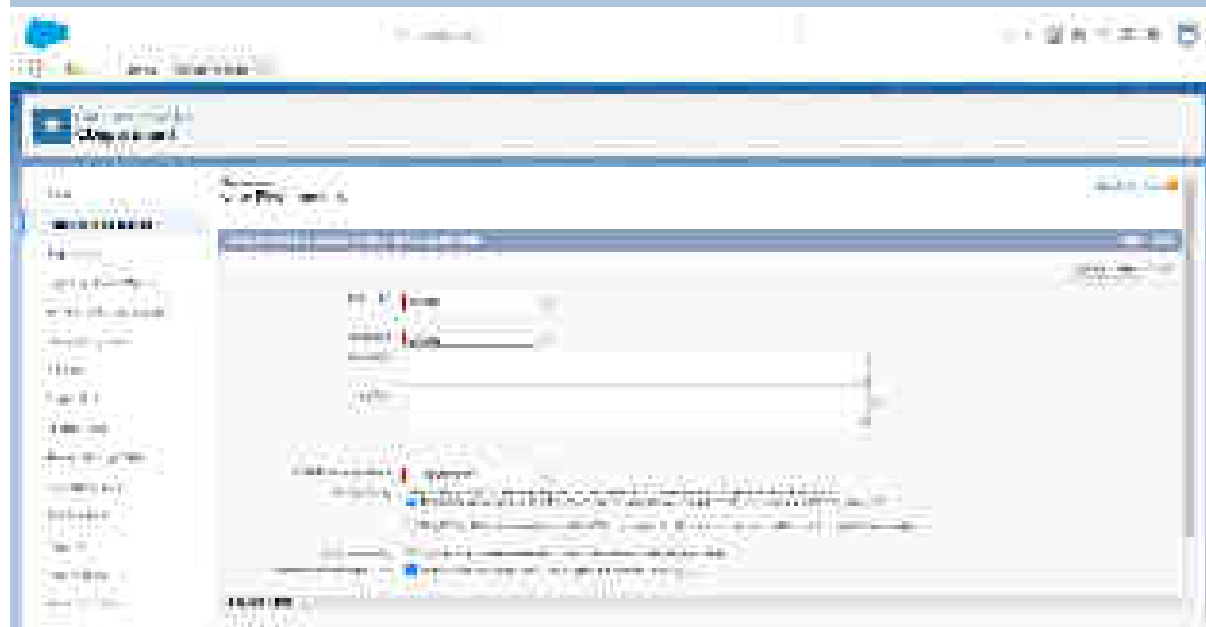


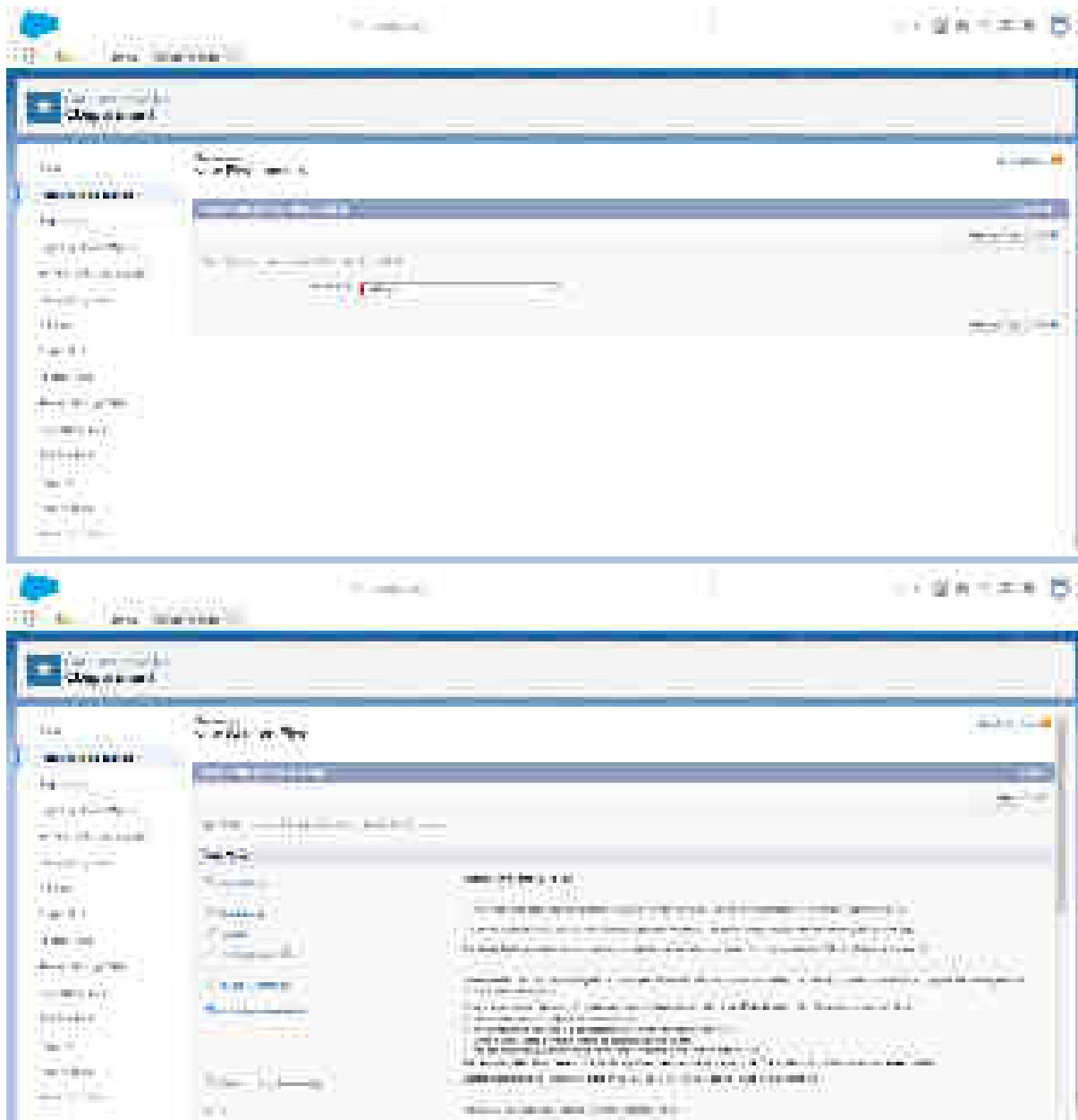
Step 2: Create a Master-Detail Relationship

To create a Master-Detail relationship between these two custom objects, follow these steps:

1. Go to Setup > Object Manager.
2. Click on "College__c" to open its settings.
3. In the left sidebar, click on "Fields & Relationships."

4. Click the "New" button to create a new custom field.
5. Choose "Master-Detail Relationship" as the data type.
6. Enter a label for the relationship, e.g., "Department __c."
7. Choose " Department__c" as the related object.
8. Configure other settings as needed and click "Next."
9. Specify the field-level security and add it to relevant page layouts.
10. Click "Next" and "Save" to create the relationship.





Step 3: Create the Roll-Up Summary Field

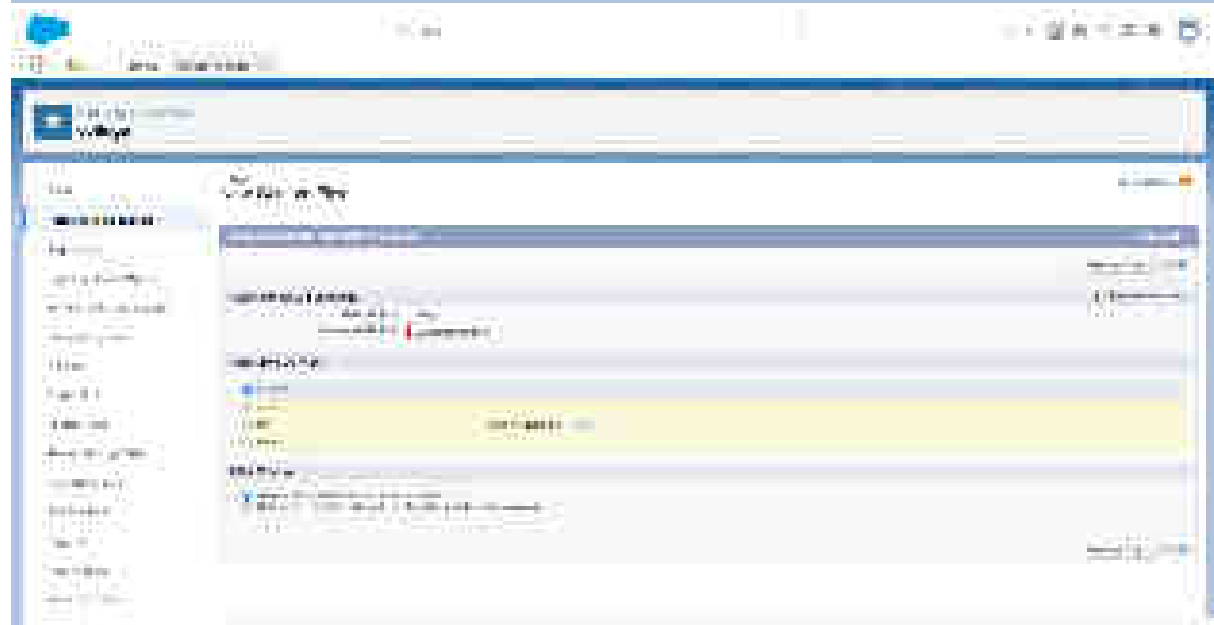
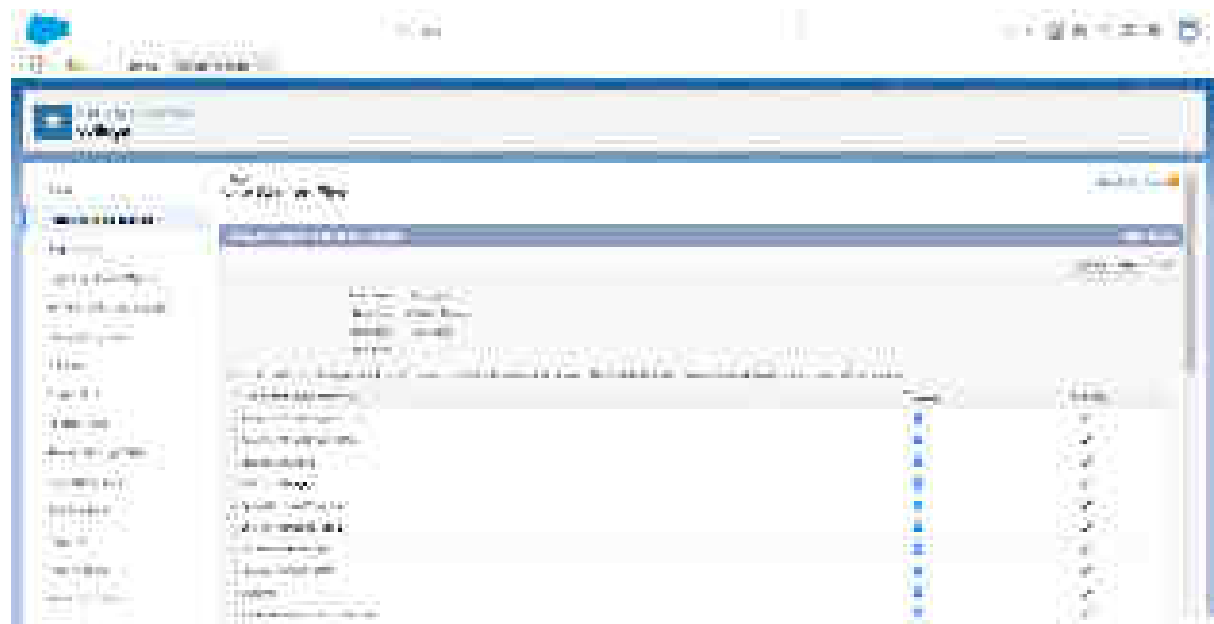
Now, let's create a Roll-Up Summary Field on the "College_C" to calculate the total number of related records in "Department_C":

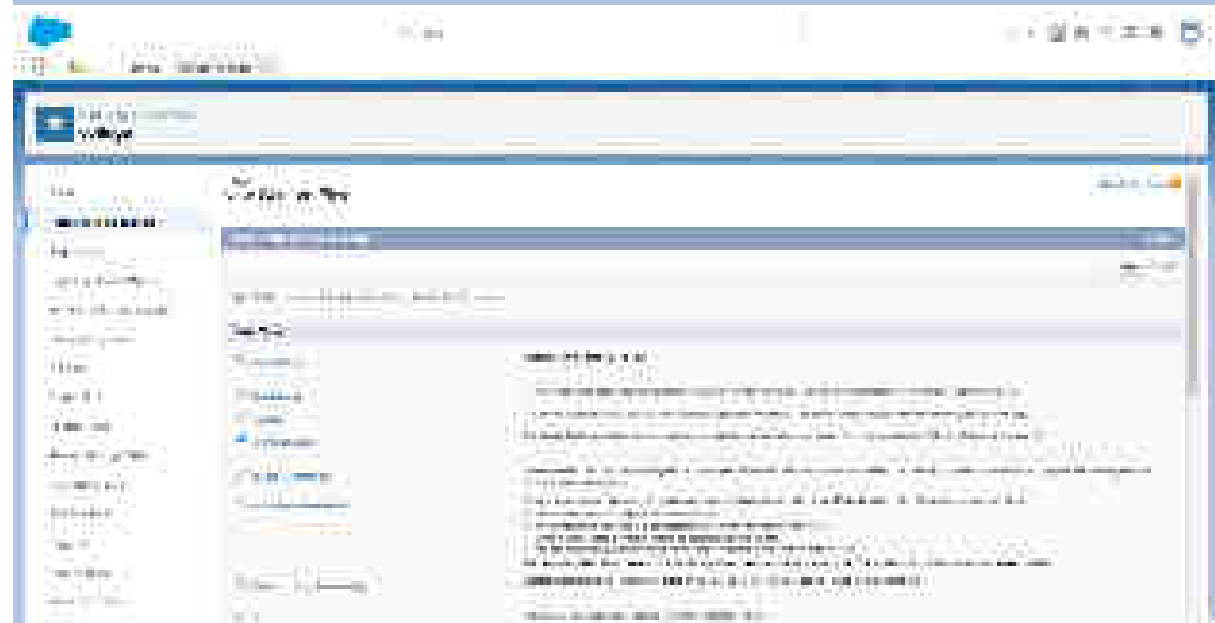
1. Still on the "College__c" settings, go to "Fields & Relationships."

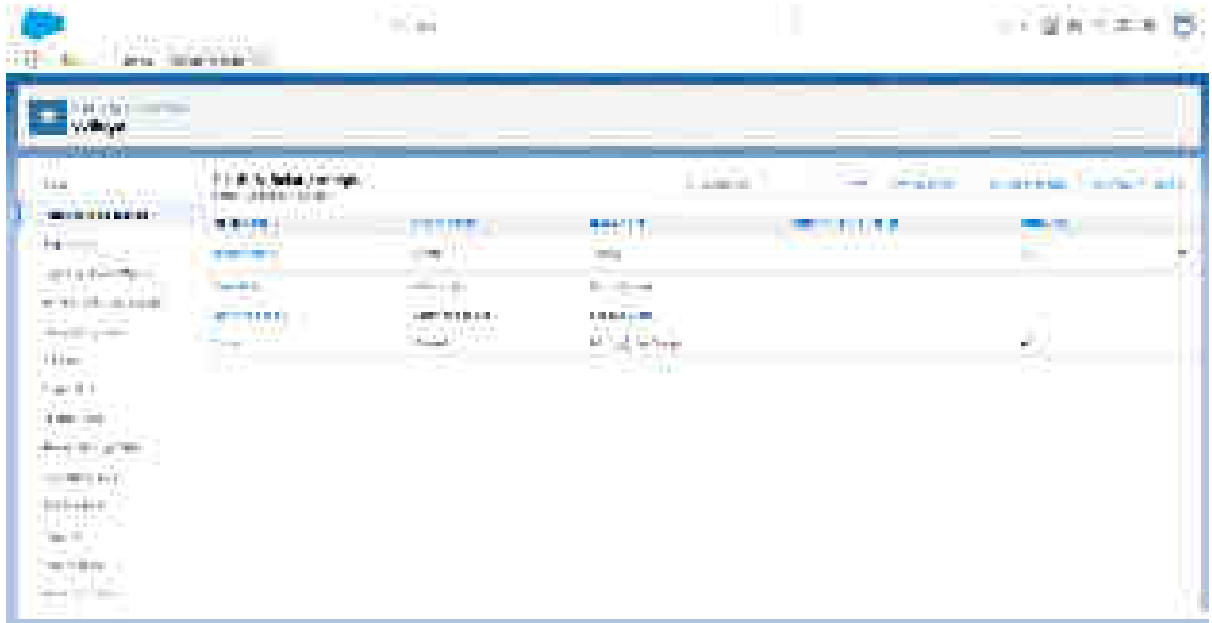
- 2. Click the "New" button to create a new custom field.**
- 3. Choose "Roll-Up Summary" as the data type.**
- 4. Enter a label for the field, e.g.,**
- 5. Choose "Count" as the Roll-Up Type.**
- 6. Select " Department__c" as the object to roll up information from.**
- 7. Specify the filter criteria if you want to filter the related records.**
- 8. Configure other settings as needed and click "Next."**
- 9. Specify the field-level security and add it to relevant page layouts.**
- 10. Click "Next" and "Save" to create the Roll-Up Summary Field.**









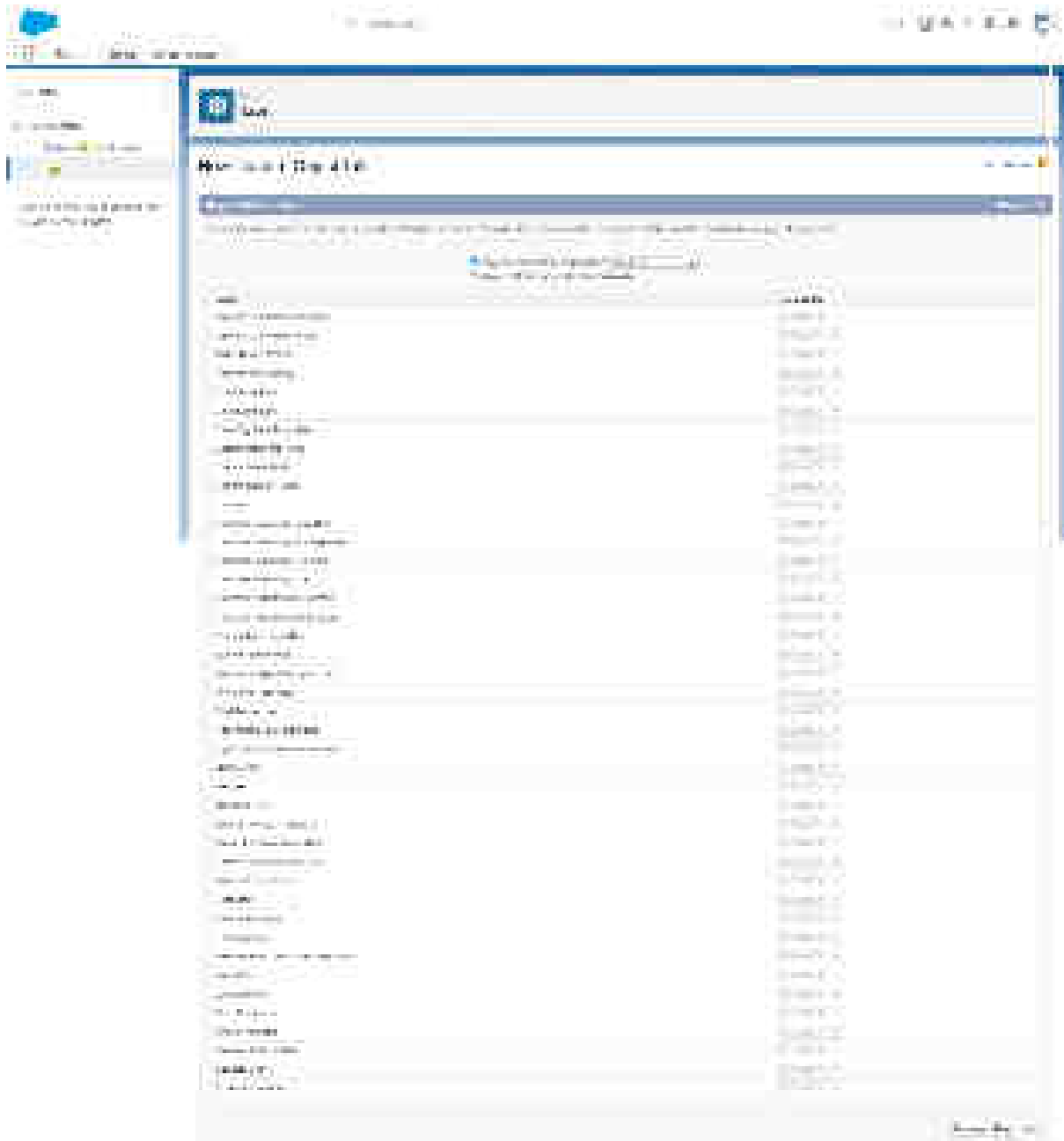


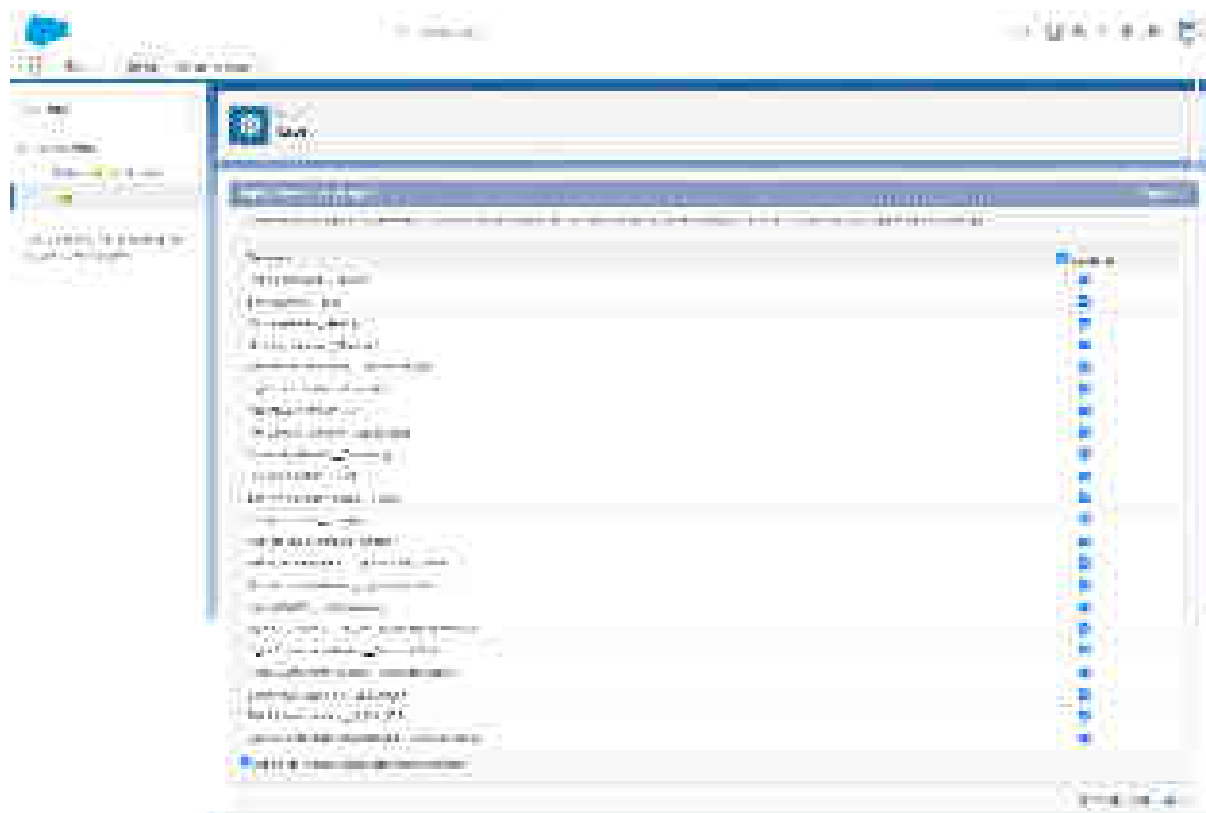
Step 4: Create a Lightning App

- 1.Type and select "App Manager."**
- 2.Click "New Lightning App."**
- 3.Fill in basic information (Name, Developer Name, Description).**
- 4.Choose the App Type (Standard, Console, Custom).**
- 5.Customize the Logo and Colour Scheme.**
- 6.Configure Navigation Items (objects to appear in the app's menu).**
- 7.Set the App Visibility (default access).**
- 8.Optionally, choose Record Pages (Lightning Record Pages).**
- 9.Review and Save the app.**

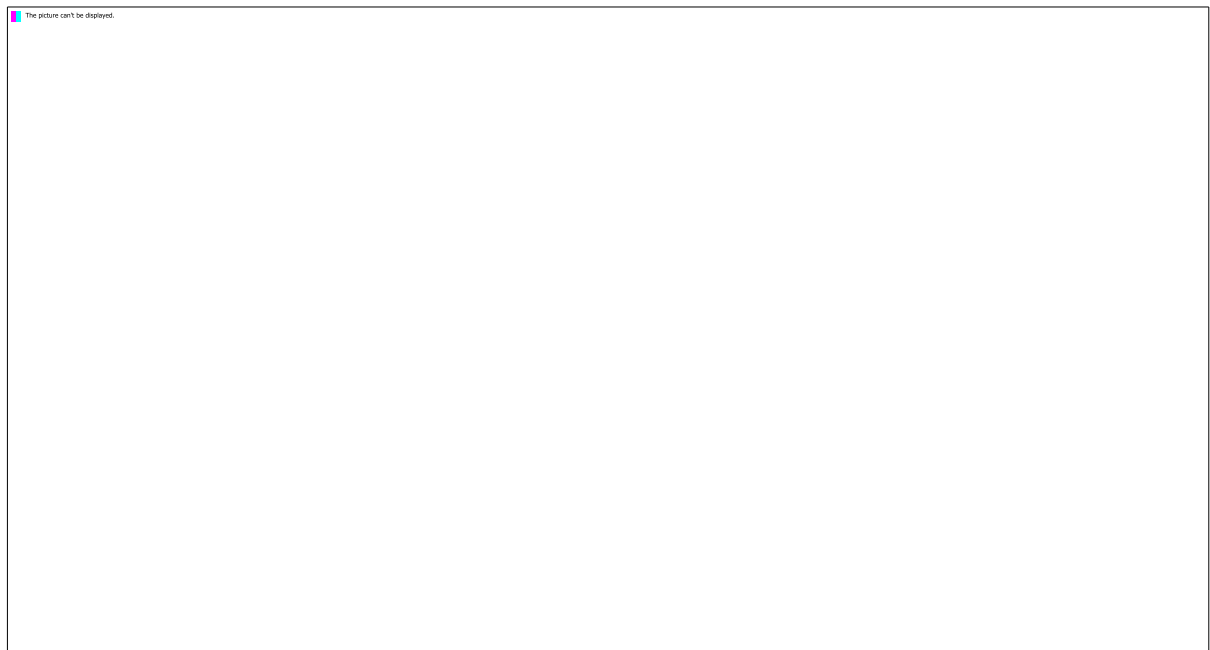
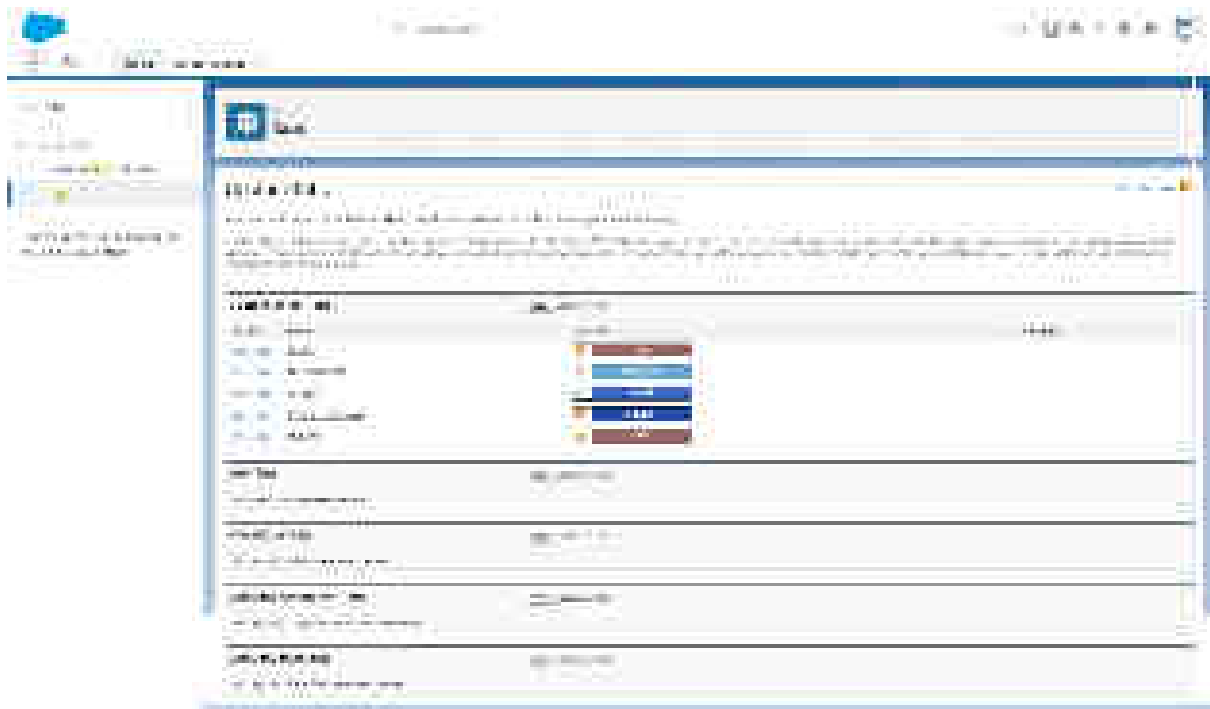
10. Assign the app to users or profiles.

11. Test the app with the assigned users.





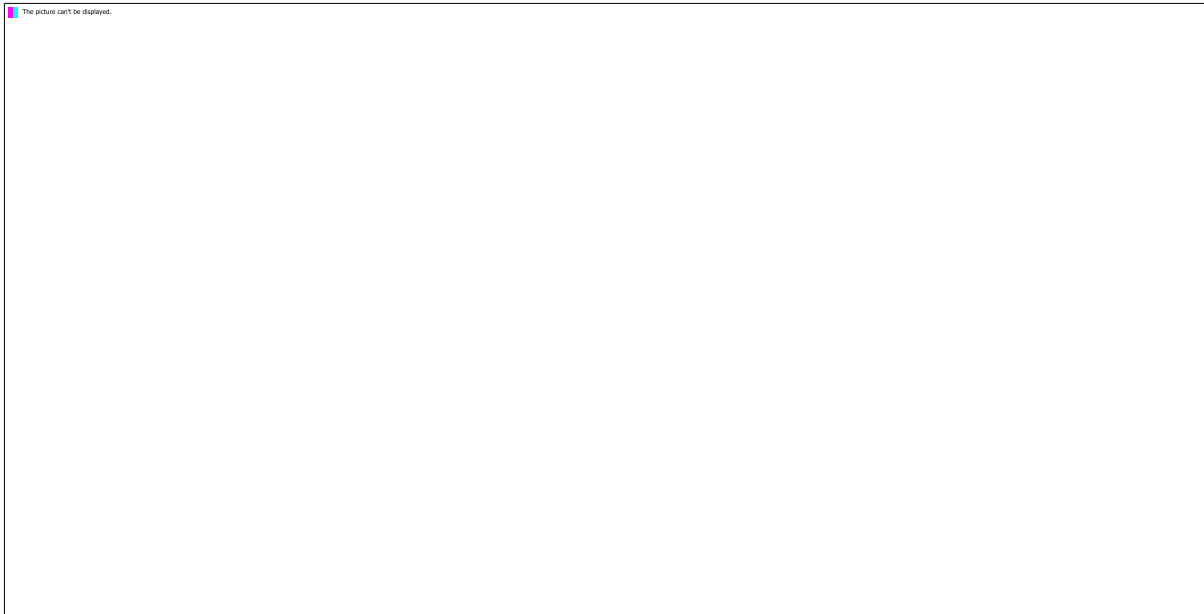





Conclusion:


Now, whenever you create or update a record in the "Department__c" related to a "College__c," the "TotalCount__c" field on the "College__c" will automatically update to show the total number of related records.

Remember to adjust field-level security, validation rules, and page layouts as needed to ensure that your custom objects and fields are appropriately configured for your organization's requirements.





 The picture can't be displayed.

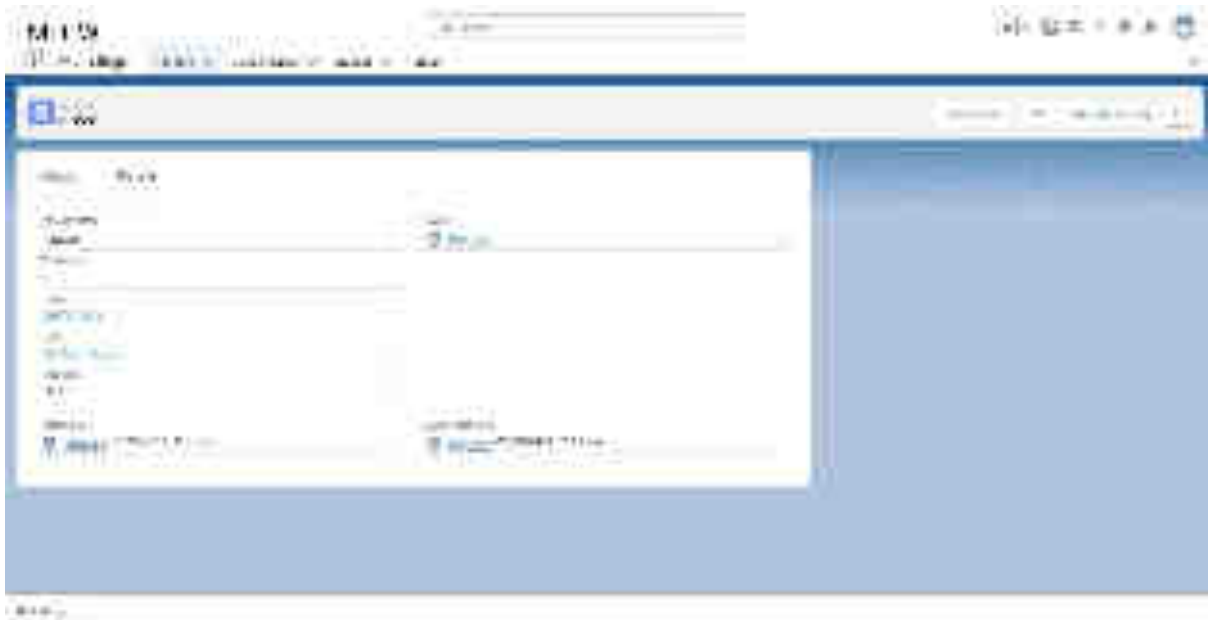
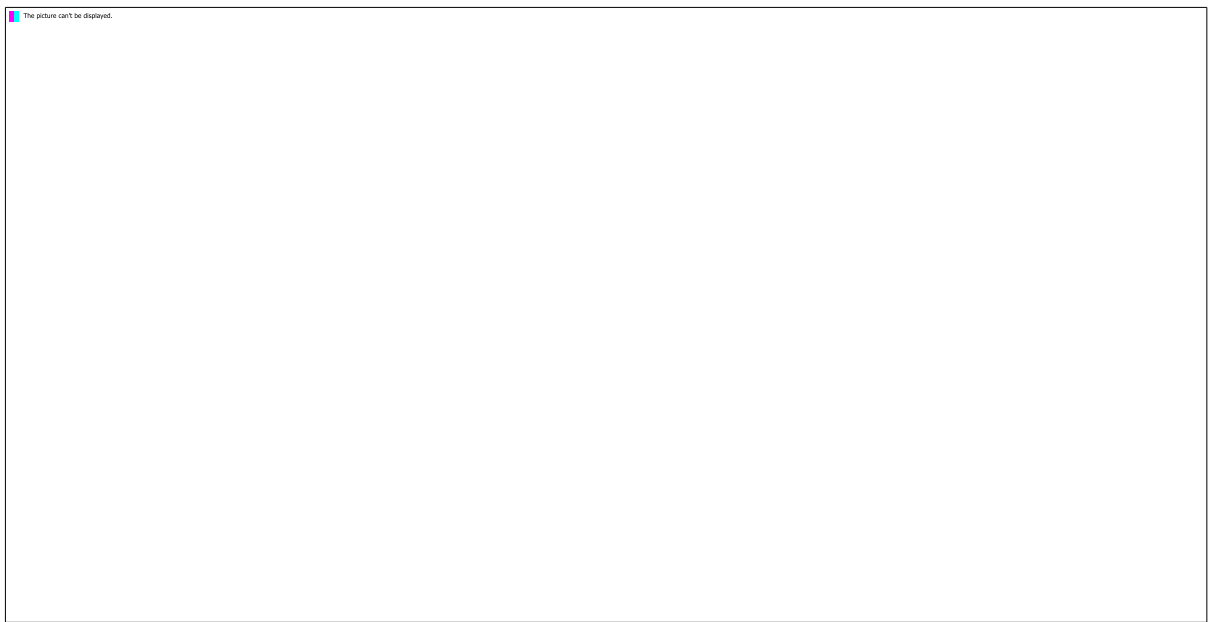
 The picture can't be displayed.

 The picture can't be displayed.

 The picture can't be displayed.

 The picture can't be displayed.

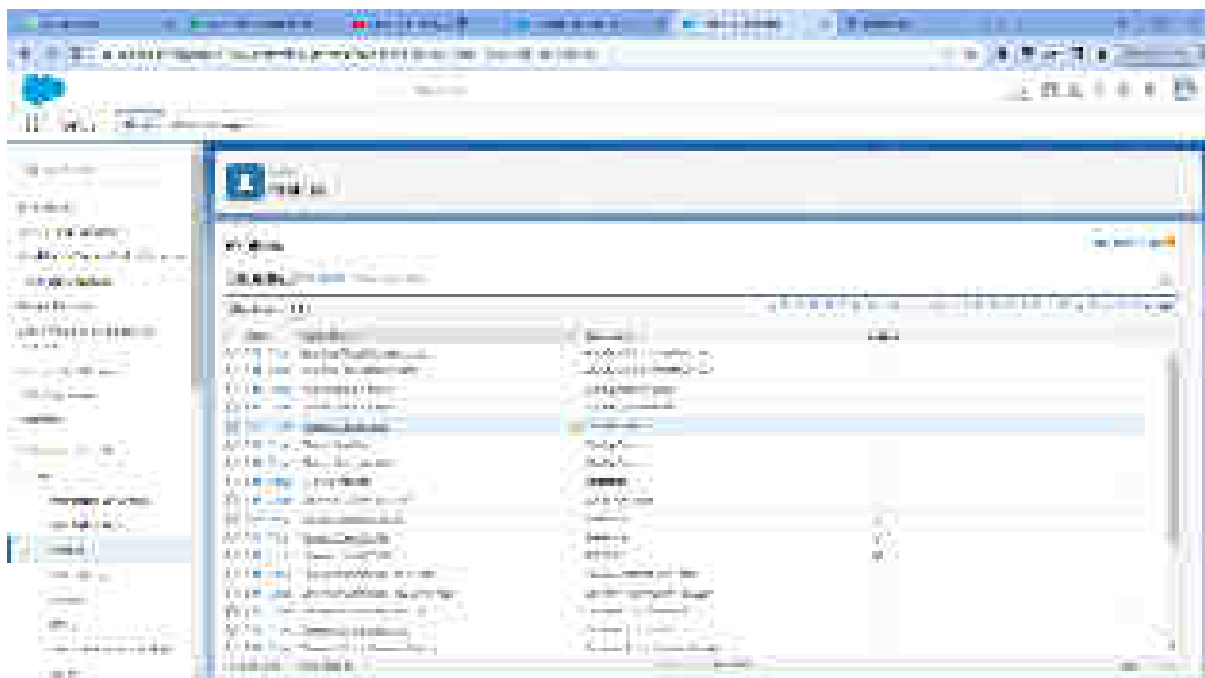
 The picture can't be displayed.



2. If there is 2 user, User A and User B in the organisation and we want in Account object that User A should not see the User B Record and user B should not see User A record then apply the Security for the users.

Solution:

Step 1: Create two separate custom profiles, one for User A and one for User B.



Navigation: Home, Reports, Settings, Help

Search: [Search bar]

Left Sidebar: Home, Reports, Settings, Help, [Other menu items]

Header: User Profile, [User Name], [Avatar]

Reports

Report Name: [Report Name]

Report ID: [Report ID]

Item	Value	Unit
Item 1	100	kg
Item 2	200	kg
Item 3	300	kg
Item 4	400	kg
Item 5	500	kg
Item 6	600	kg
Item 7	700	kg
Item 8	800	kg
Item 9	900	kg
Item 10	1000	kg

Footer: [Page Number], [Page Title]

Navigation: Home, Reports, Settings, Help

Search: [Search bar]

Left Sidebar: Home, Reports, Settings, Help, [Other menu items]

Header: User Profile, [User Name], [Avatar]

Reports

Report Name: [Report Name]

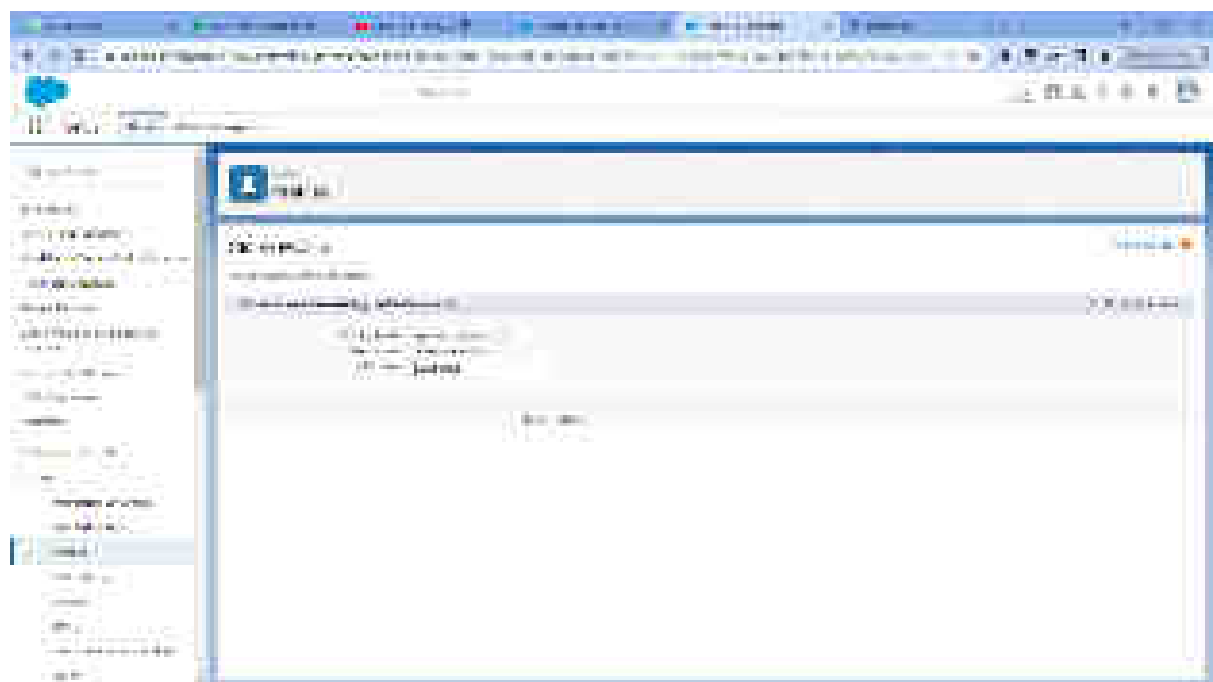
Report ID: [Report ID]

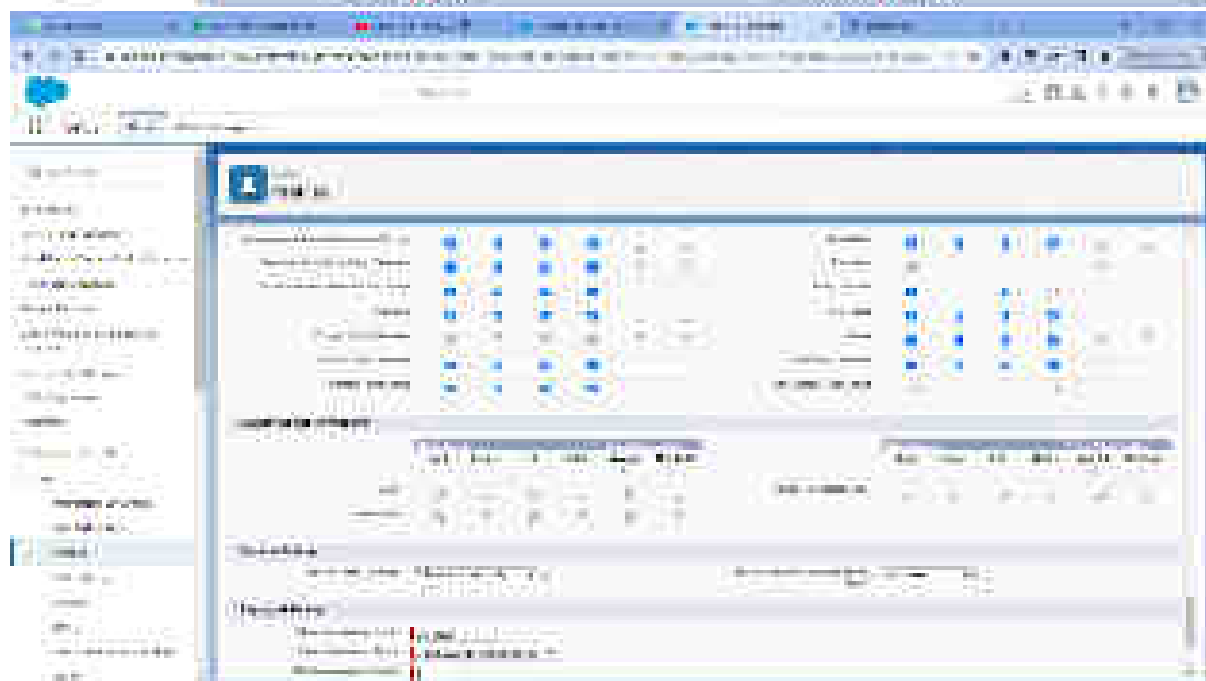
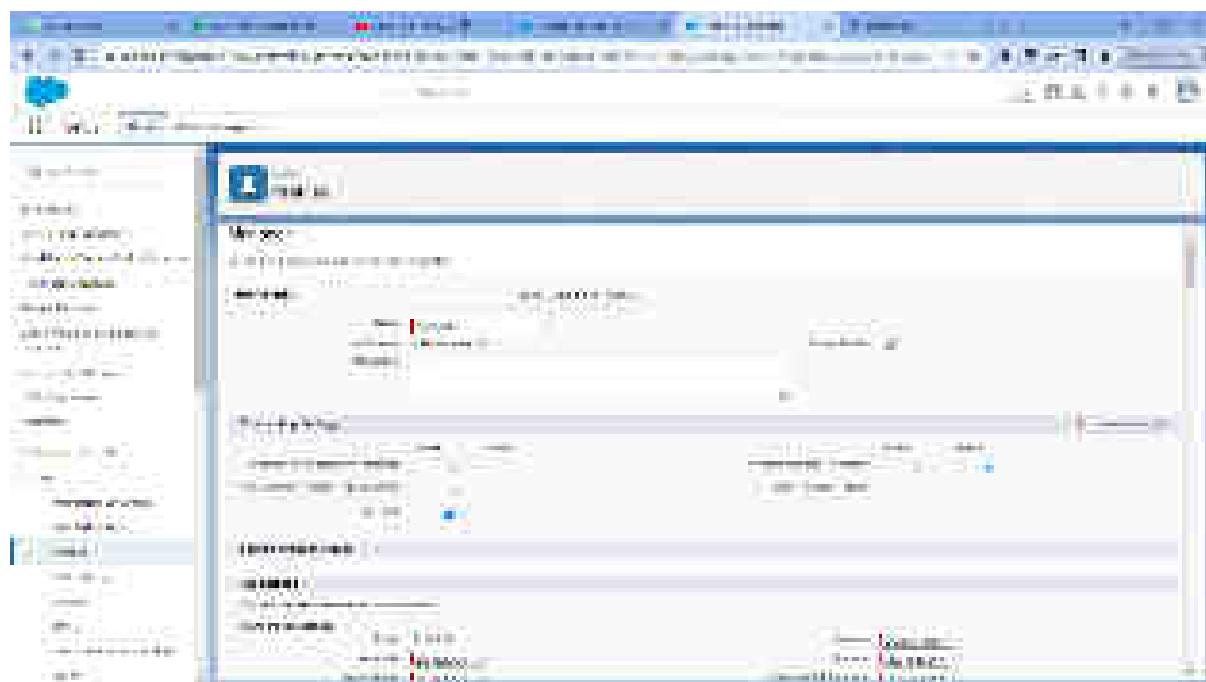
Report Description: [Report Description]

Report Data: [Report Data]

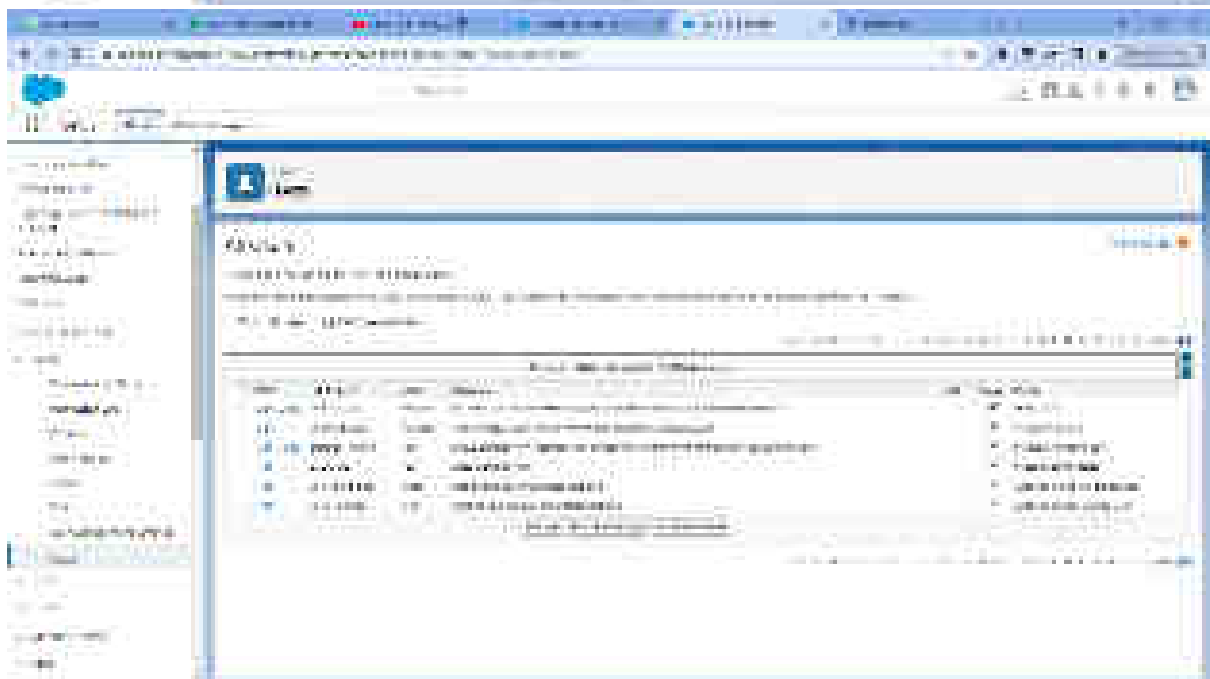
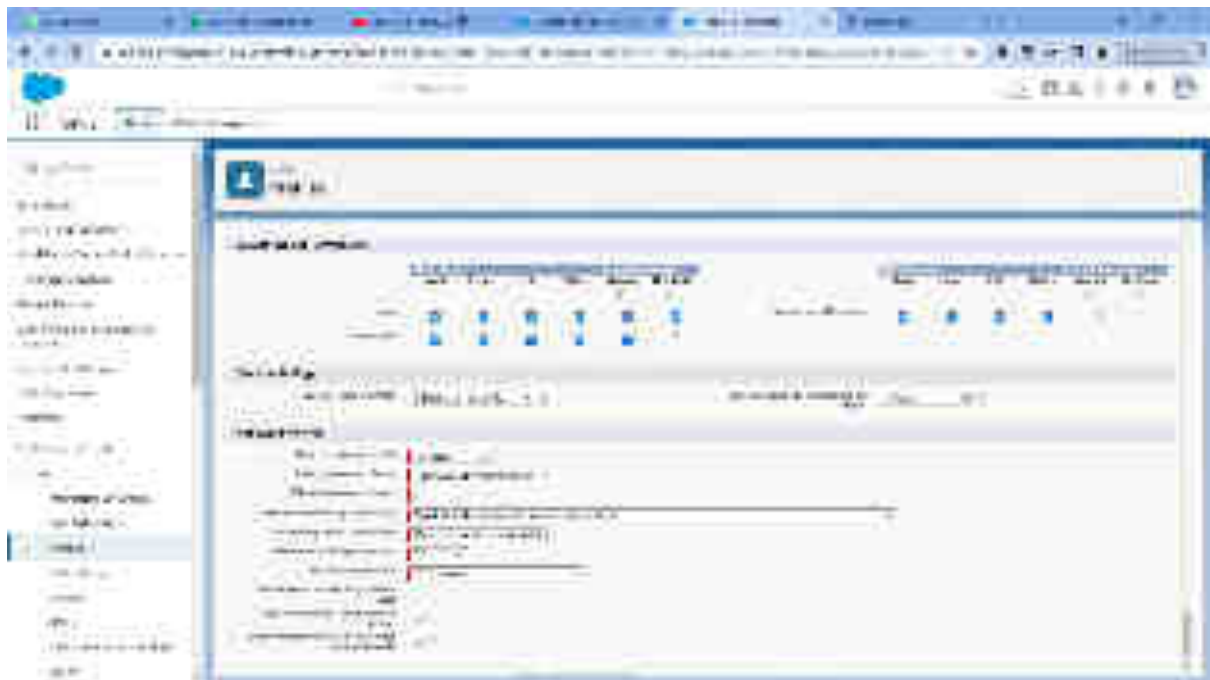
Report Summary: [Report Summary]

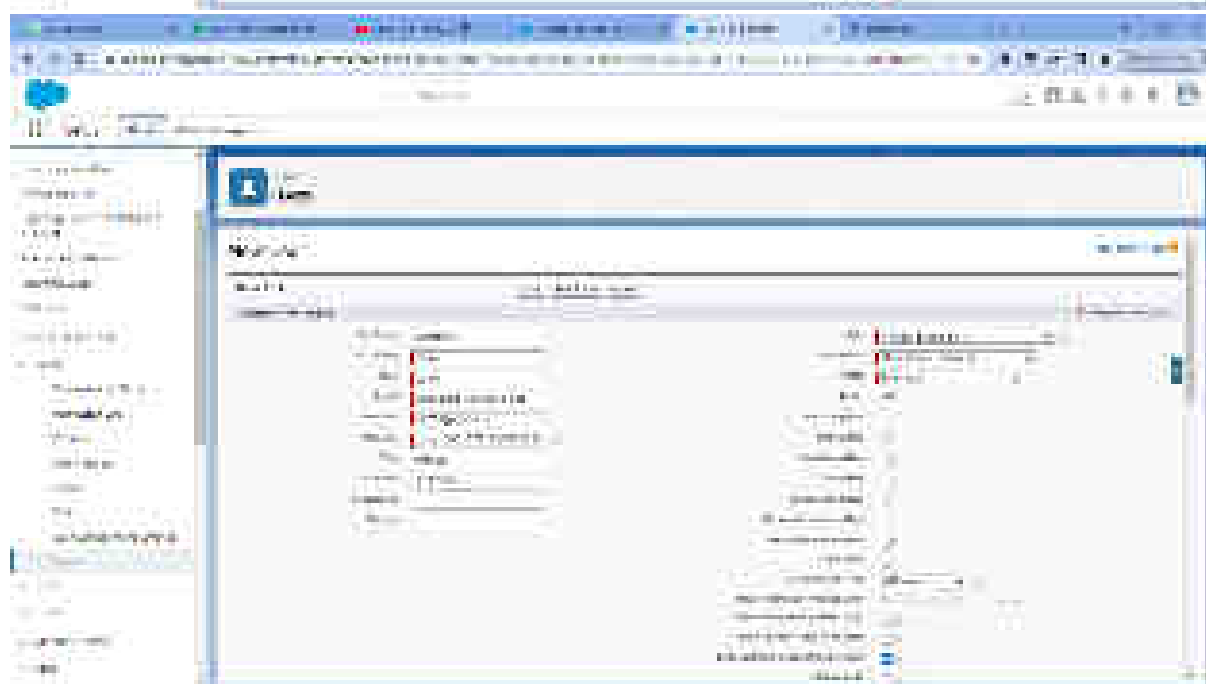
Report Footer: [Report Footer]

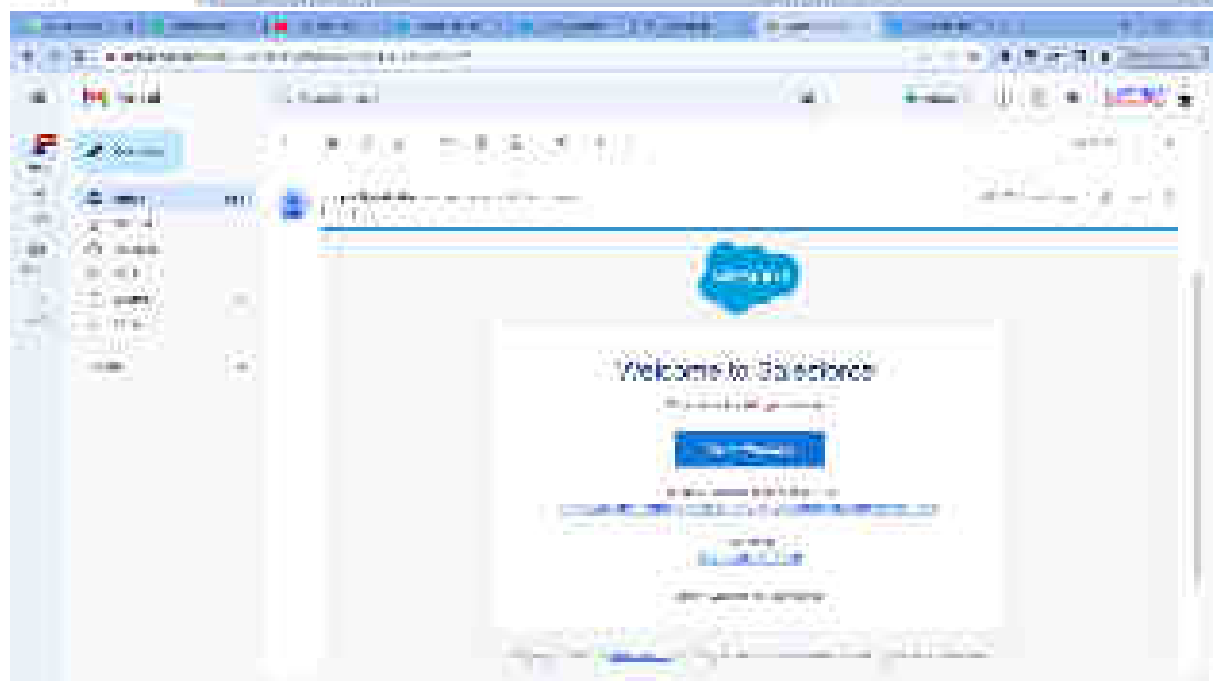


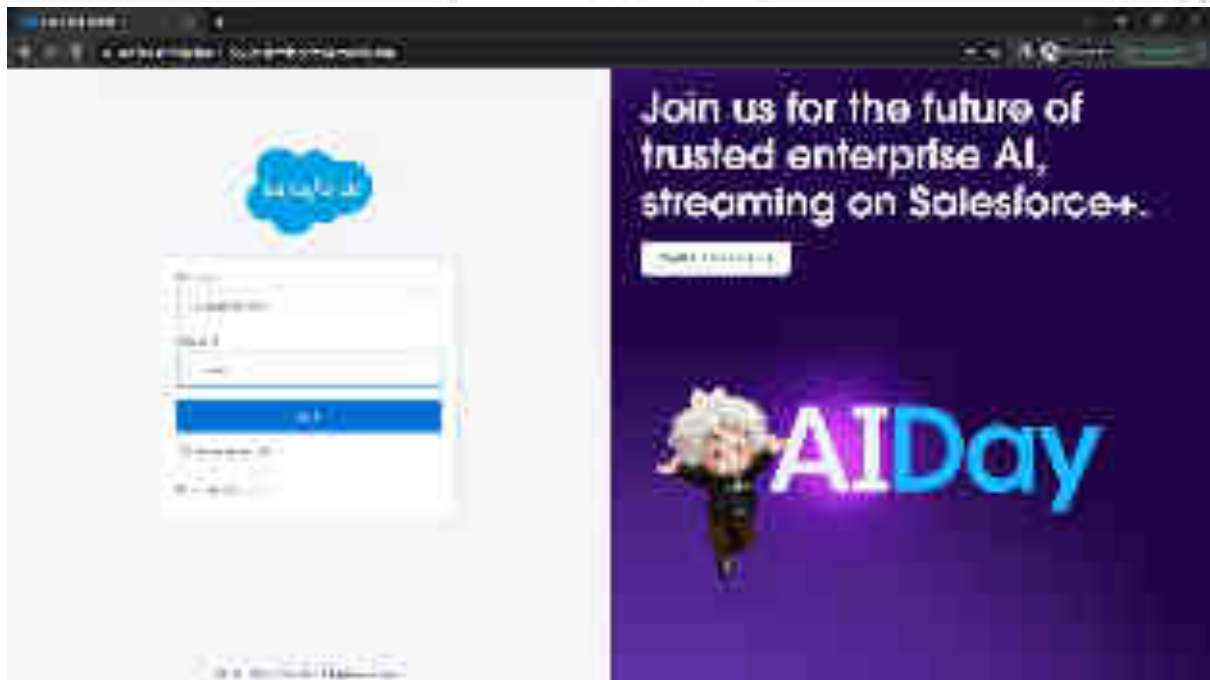
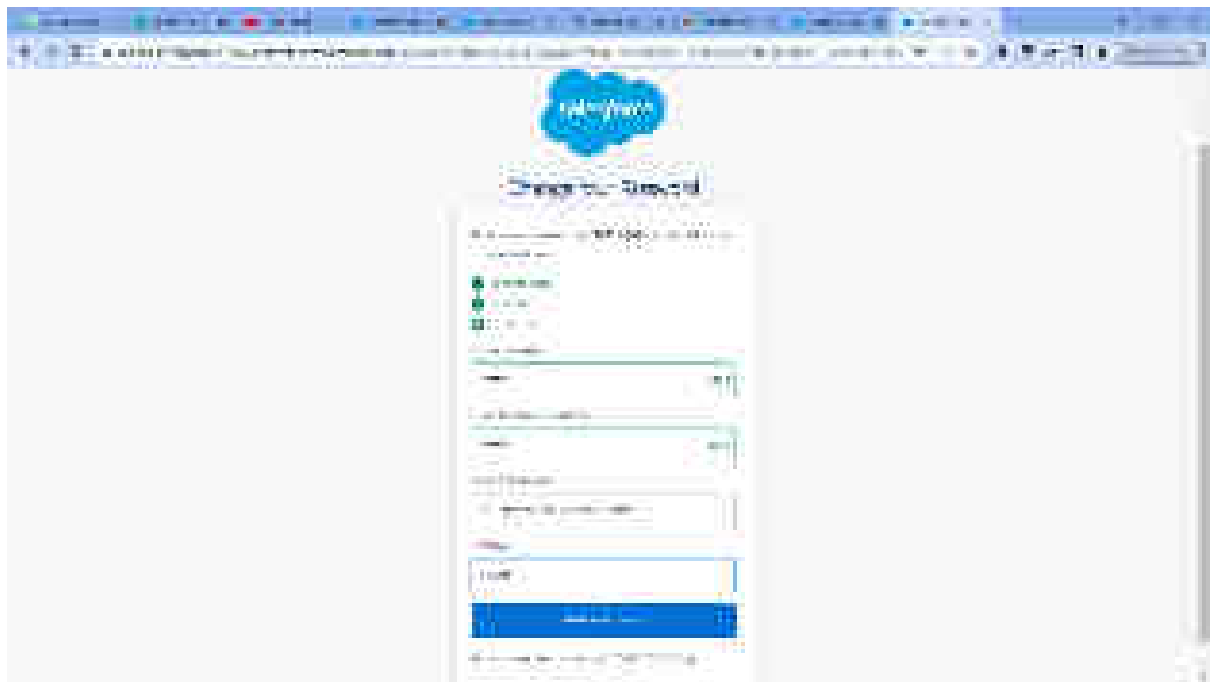


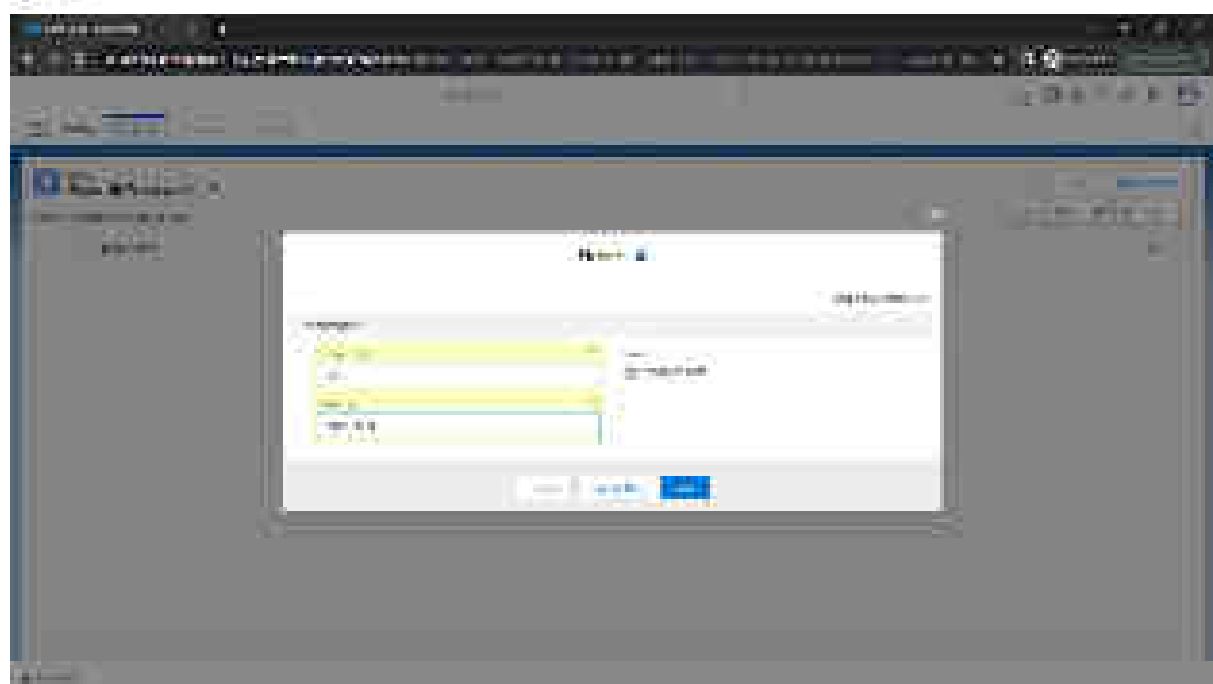
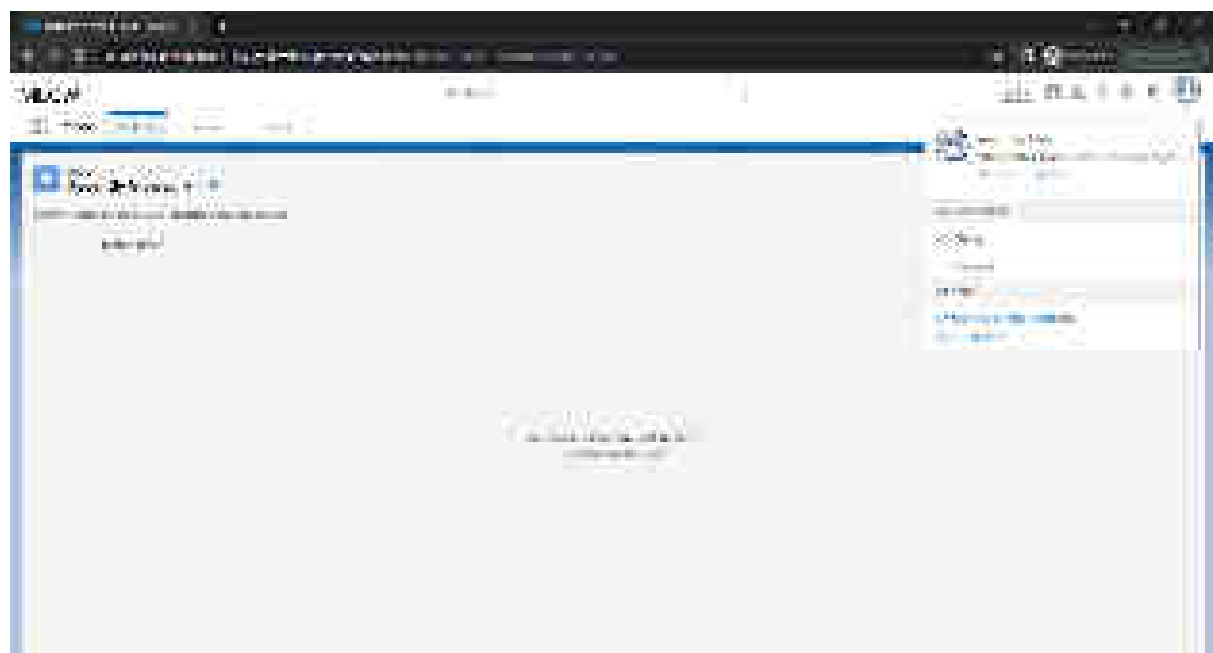


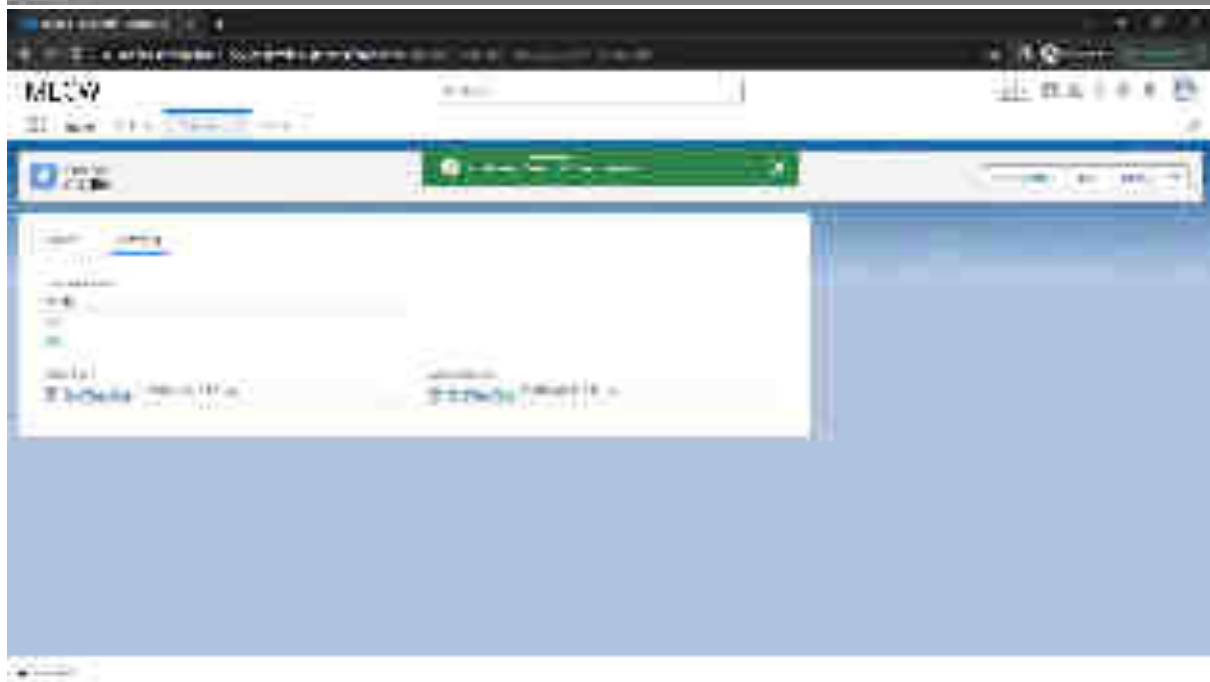
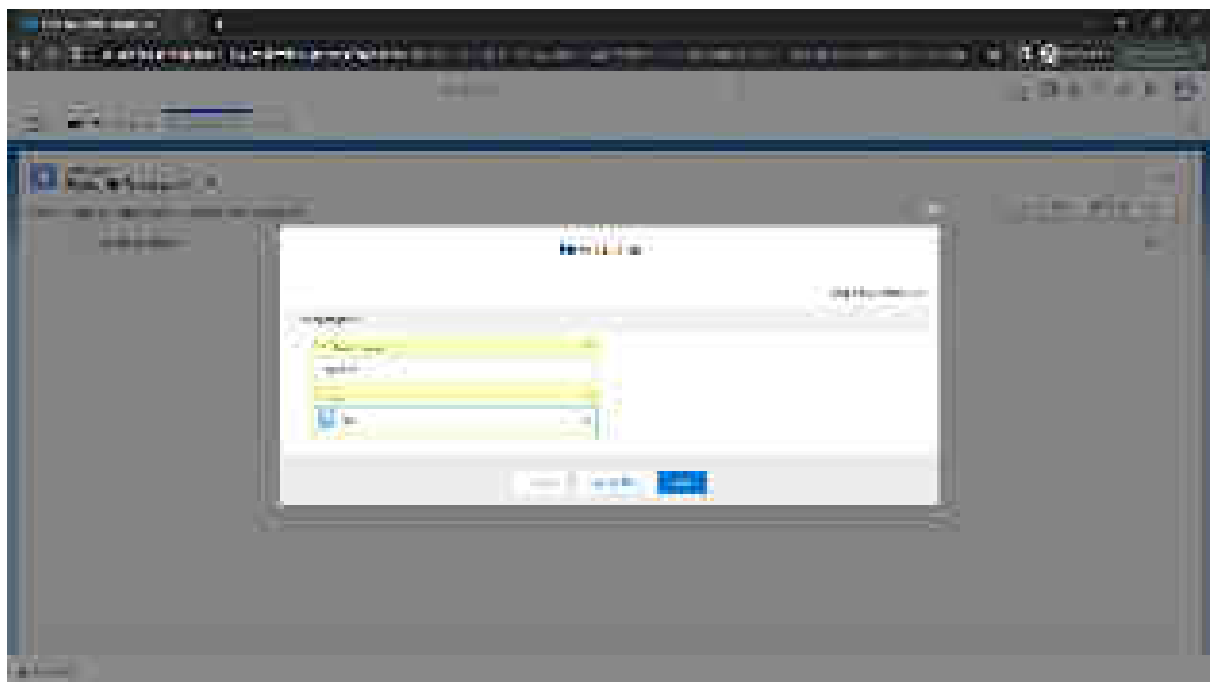












Navigation: Home, Reports, Settings, etc.

Profile

Personal Information

Name: [Name] | Email: [Email] | Phone: [Phone]

Address

Street: [Street] | City: [City] | State: [State] | Zip: [Zip]

Education

University: [University] | Degree: [Degree] | Graduation Year: [Year]

Work Experience

Company: [Company] | Position: [Position] | Start Date: [Date] | End Date: [Date]

Skills

Language: [Language] | Proficiency: [Proficiency] | Other Skills: [Skills]

Navigation: Home, Reports, Settings, etc.

Profile

Personal Information

Name: [Name] | Email: [Email] | Phone: [Phone]

Address

Street: [Street] | City: [City] | State: [State] | Zip: [Zip]

Education

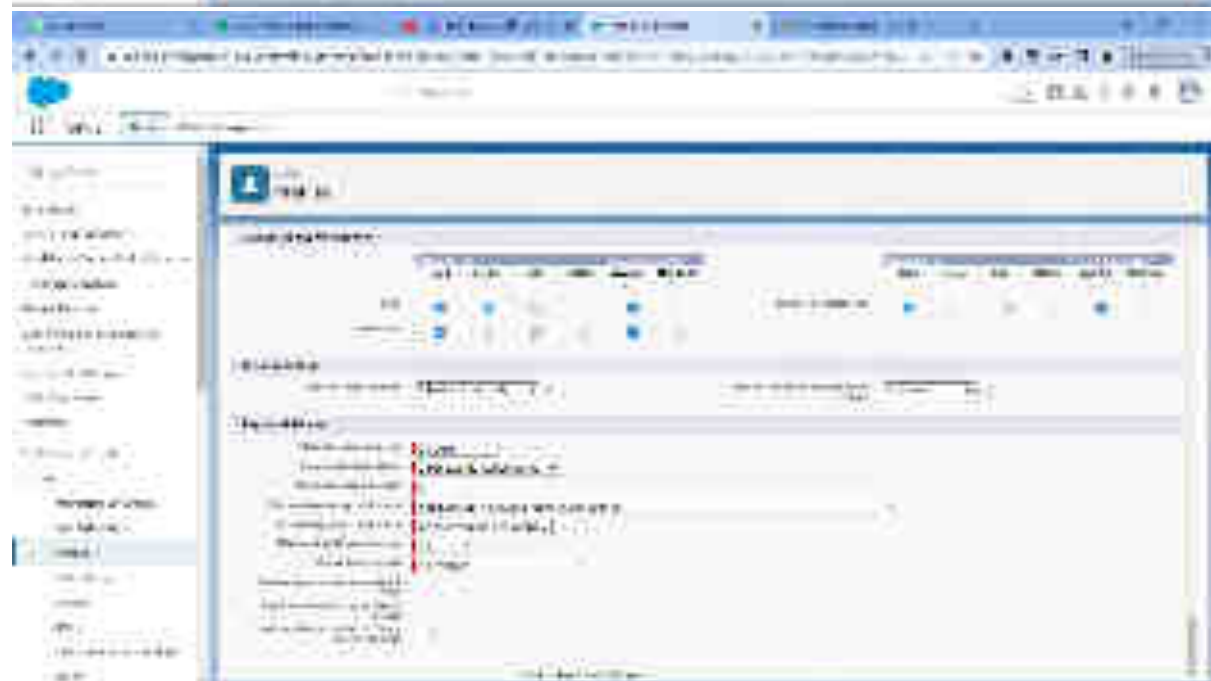
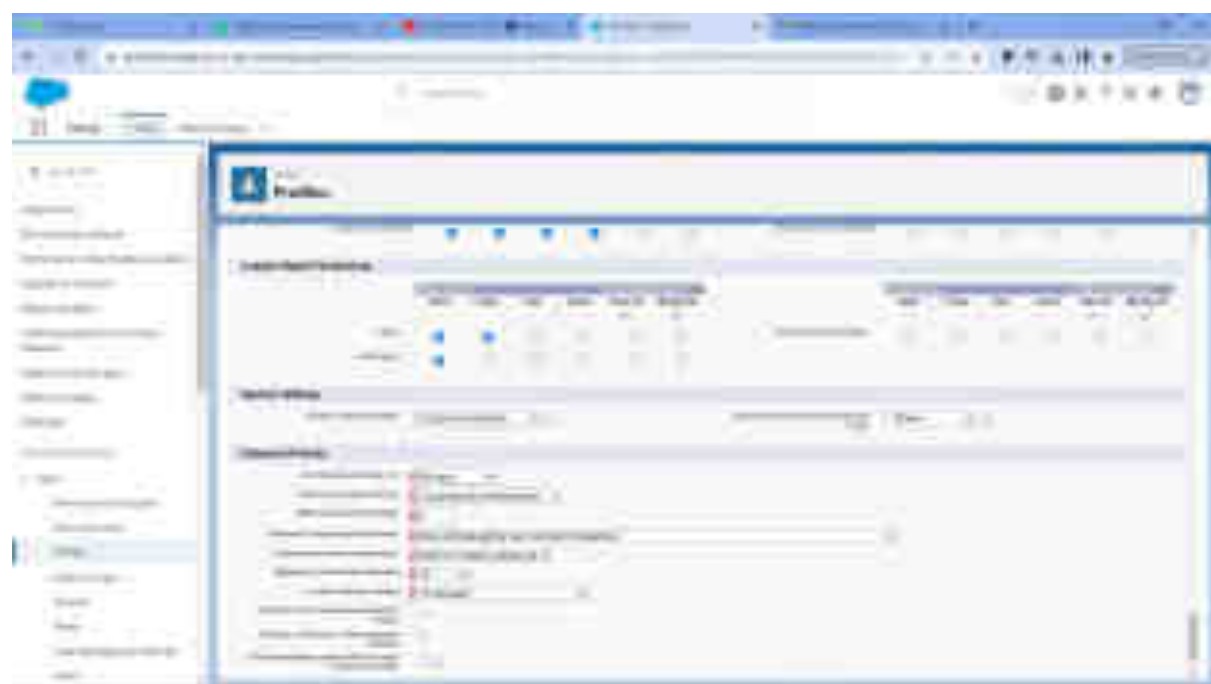
University: [University] | Degree: [Degree] | Graduation Year: [Year]

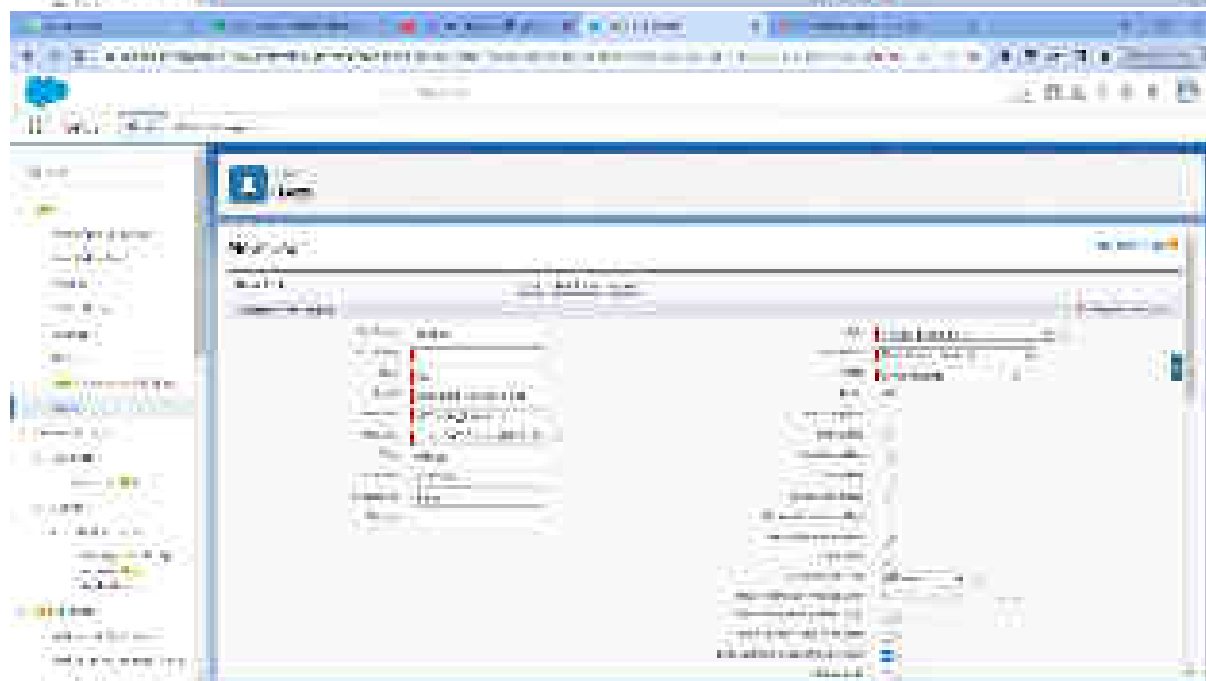
Work Experience

Company: [Company] | Position: [Position] | Start Date: [Date] | End Date: [Date]

Skills

Language: [Language] | Proficiency: [Proficiency] | Other Skills: [Skills]





Microsoft Word - [Document Name].docx

File Home Layouts References Sendungen

1. Einleitung

1.1. Zielsetzung

1.2. Aufgabenstellung

1.3. Methodik

1.4. Ergebnisse

1.5. Diskussion

1.6. Fazit

1.7. Literaturverzeichnis

1.8. Anhang

1.9. Zusammenfassung

1.10. Schlusswort

1.11. Danksagung

1.12. Impressum

1.13. Kontakt

1.14. Impressum

1.15. Impressum

1.16. Impressum

1.17. Impressum

1.18. Impressum

1.19. Impressum

1.20. Impressum

1.21. Impressum

1.22. Impressum

1.23. Impressum

1.24. Impressum

1.25. Impressum

1.26. Impressum

1.27. Impressum

1.28. Impressum

1.29. Impressum

1.30. Impressum

1.31. Impressum

1.32. Impressum

1.33. Impressum

1.34. Impressum

1.35. Impressum

1.36. Impressum

1.37. Impressum

1.38. Impressum

1.39. Impressum

1.40. Impressum

1.41. Impressum

1.42. Impressum

1.43. Impressum

1.44. Impressum

1.45. Impressum

1.46. Impressum

1.47. Impressum

1.48. Impressum

1.49. Impressum

1.50. Impressum

1.51. Impressum

1.52. Impressum

1.53. Impressum

1.54. Impressum

1.55. Impressum

1.56. Impressum

1.57. Impressum

1.58. Impressum

1.59. Impressum

1.60. Impressum

1.61. Impressum

1.62. Impressum

1.63. Impressum

1.64. Impressum

1.65. Impressum

1.66. Impressum

1.67. Impressum

1.68. Impressum

1.69. Impressum

1.70. Impressum

1.71. Impressum

1.72. Impressum

1.73. Impressum

1.74. Impressum

1.75. Impressum

1.76. Impressum

1.77. Impressum

1.78. Impressum

1.79. Impressum

1.80. Impressum

1.81. Impressum

1.82. Impressum

1.83. Impressum

1.84. Impressum

1.85. Impressum

1.86. Impressum

1.87. Impressum

1.88. Impressum

1.89. Impressum

1.90. Impressum

1.91. Impressum

1.92. Impressum

1.93. Impressum

1.94. Impressum

1.95. Impressum

1.96. Impressum

1.97. Impressum

1.98. Impressum

1.99. Impressum

1.100. Impressum

Microsoft Word - [Document Name].docx

File Home Layouts References Sendungen

1. Einleitung

1.1. Zielsetzung

1.2. Aufgabenstellung

1.3. Methodik

1.4. Ergebnisse

1.5. Diskussion

1.6. Fazit

1.7. Literaturverzeichnis

1.8. Anhang

1.9. Zusammenfassung

1.10. Schlusswort

1.11. Danksagung

1.12. Impressum

1.13. Kontakt

1.14. Impressum

1.15. Impressum

1.16. Impressum

1.17. Impressum

1.18. Impressum

1.19. Impressum

1.20. Impressum

1.21. Impressum

1.22. Impressum

1.23. Impressum

1.24. Impressum

1.25. Impressum

1.26. Impressum

1.27. Impressum

1.28. Impressum

1.29. Impressum

1.30. Impressum

1.31. Impressum

1.32. Impressum

1.33. Impressum

1.34. Impressum

1.35. Impressum

1.36. Impressum

1.37. Impressum

1.38. Impressum

1.39. Impressum

1.40. Impressum

1.41. Impressum

1.42. Impressum

1.43. Impressum

1.44. Impressum

1.45. Impressum

1.46. Impressum

1.47. Impressum

1.48. Impressum

1.49. Impressum

1.50. Impressum

1.51. Impressum

1.52. Impressum

1.53. Impressum

1.54. Impressum

1.55. Impressum

1.56. Impressum

1.57. Impressum

1.58. Impressum

1.59. Impressum

1.60. Impressum

1.61. Impressum

1.62. Impressum

1.63. Impressum

1.64. Impressum

1.65. Impressum

1.66. Impressum

1.67. Impressum

1.68. Impressum

1.69. Impressum

1.70. Impressum

1.71. Impressum

1.72. Impressum

1.73. Impressum

1.74. Impressum

1.75. Impressum

1.76. Impressum

1.77. Impressum

1.78. Impressum

1.79. Impressum

1.80. Impressum

1.81. Impressum

1.82. Impressum

1.83. Impressum

1.84. Impressum

1.85. Impressum

1.86. Impressum

1.87. Impressum

1.88. Impressum

1.89. Impressum

1.90. Impressum

1.91. Impressum

1.92. Impressum

1.93. Impressum

1.94. Impressum

1.95. Impressum

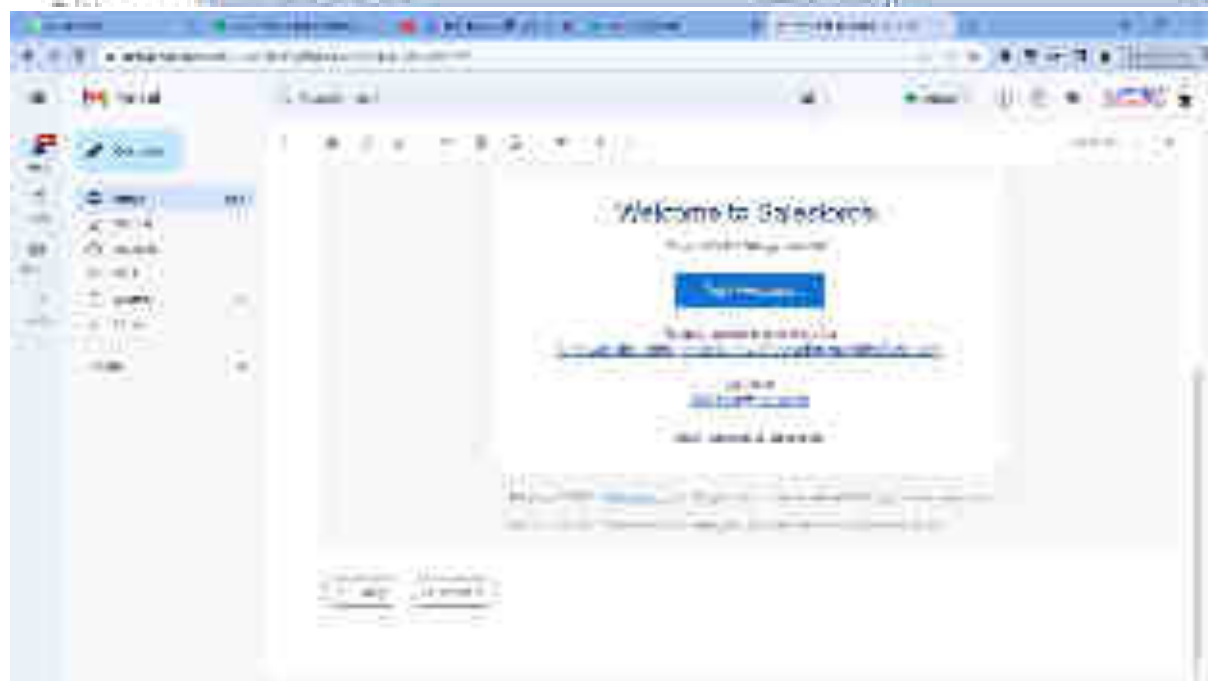
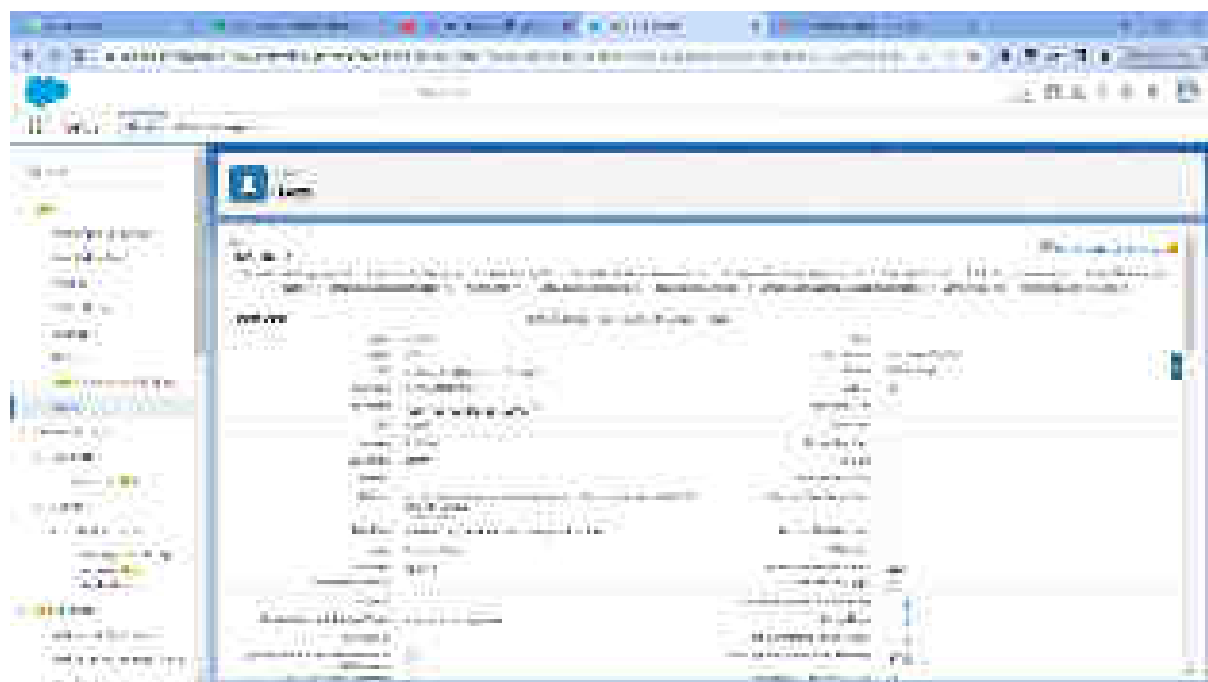
1.96. Impressum

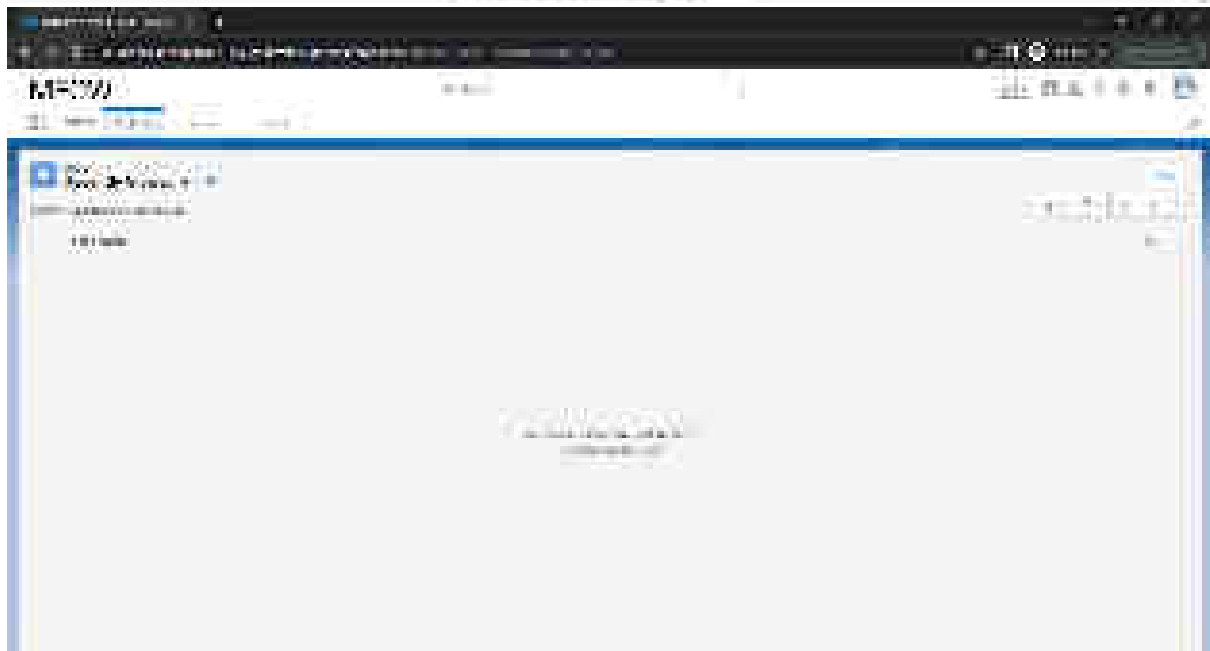
1.97. Impressum

1.98. Impressum

1.99. Impressum

1.100. Impressum







Step 2:

Permission Sets:

- Create two permission sets, one for User A and one for User B.

Object-Level Security:

- In each profile and permission set, set the object-level security for the Account object to "Read" to ensure that both I-Jser A and I-Jser B can view Account records.

Record-Level Security:

- Implement record-level security using Criteria-Based Sharing Rules.
- Create a sharing rule that shares Account records owned by User A with User A and records owned by IJser B with user B.
- For the sharing rule criteria, specify that records owned by User A are shared with user A, and records owned by User B are shared with User B.

Ownership:

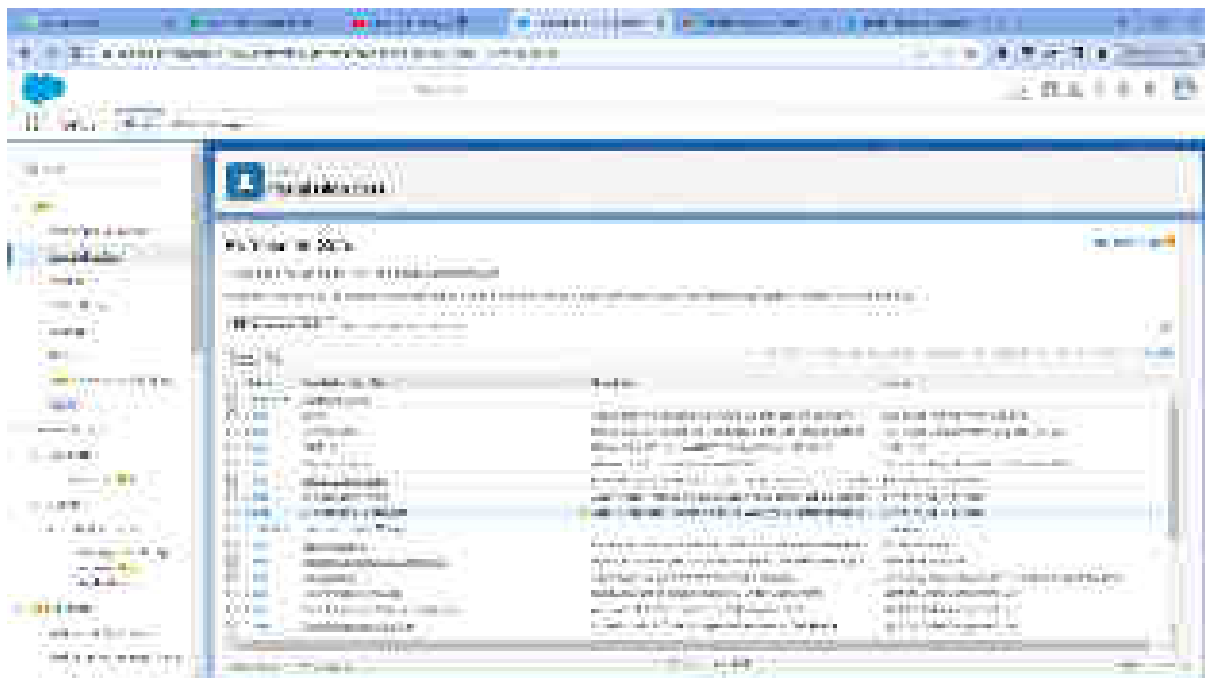
- Ensure that the Account records are owned by the respective users, with User A owning their records and User B owning their records.

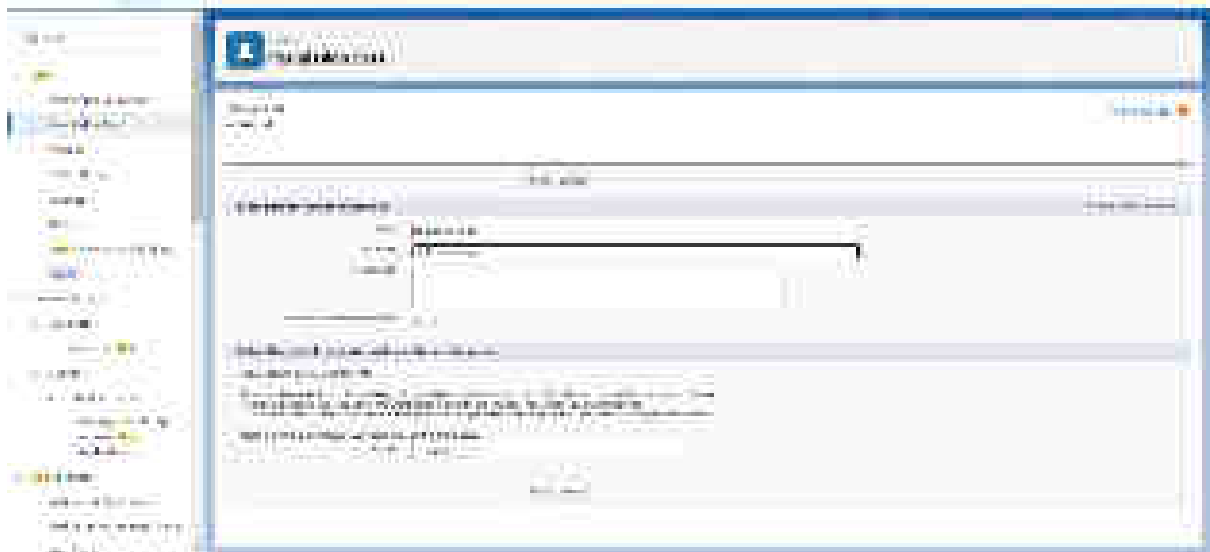
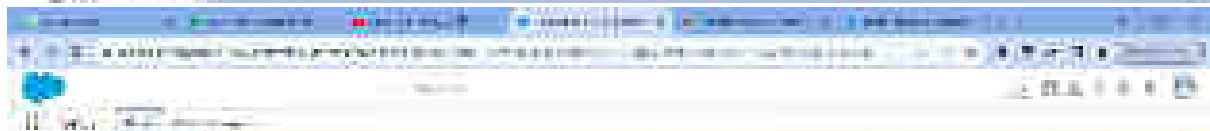
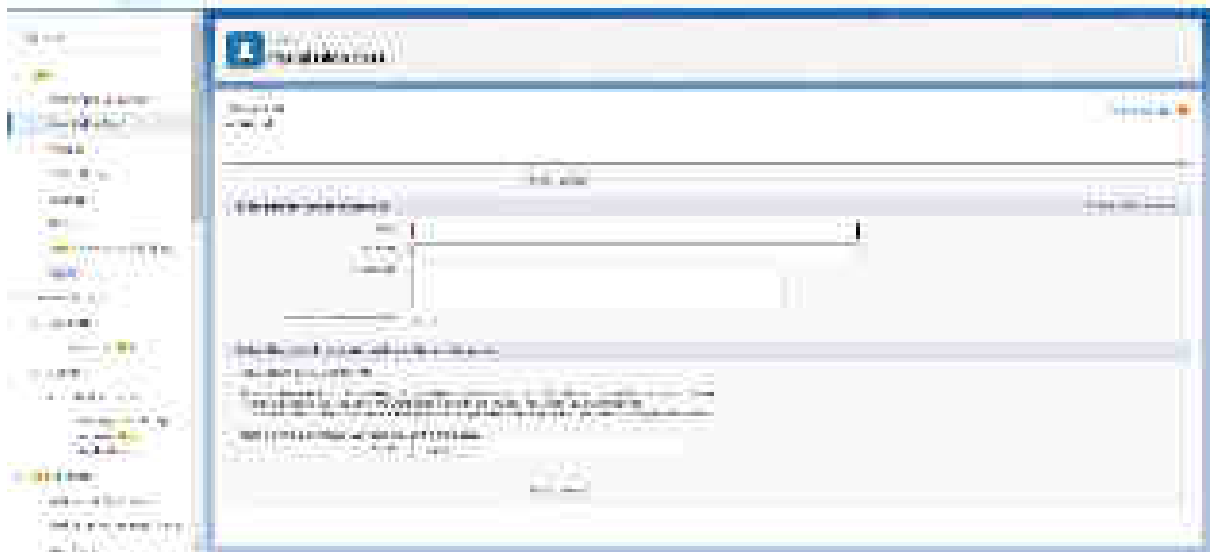
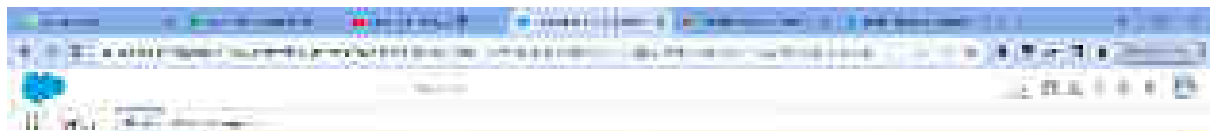
Organization-Wide Defaults:

- Set the organization-wide defaults for the Account object to "Private" to ensure that records are private by default.

Testing:

- Test the setup by logging in as User A and User B separately to verify that they cannot access each other's records.





Navigation: Home > My Account > My Profile > My Profile

My Profile

Profile Information

First Name: John
Last Name: Doe
Email: john.doe@example.com
Phone: +1 123 456 7890
Address: 123 Main St, New York, NY 10001, USA

Account Information

Username: johndoe123
Password: [REDACTED]
Confirm Password: [REDACTED]
Reset Password: [Link]

Security Settings

Two-Factor Authentication: [ON/OFF]
Security Questions: [List of questions and answers]

Privacy Settings

Profile Visibility: [Public/Private]
Post Visibility: [Public/Private]
Comment Visibility: [Public/Private]

Account Settings

Language: English
Currency: USD
Time Zone: Eastern Standard Time

Navigation: Home > My Account > My Profile > My Profile

My Profile

Profile Information

First Name: John
Last Name: Doe
Email: john.doe@example.com
Phone: +1 123 456 7890
Address: 123 Main St, New York, NY 10001, USA

Account Information

Username: johndoe123
Password: [REDACTED]
Confirm Password: [REDACTED]
Reset Password: [Link]

Security Settings

Two-Factor Authentication: [ON/OFF]
Security Questions: [List of questions and answers]

Privacy Settings

Profile Visibility: [Public/Private]
Post Visibility: [Public/Private]
Comment Visibility: [Public/Private]

Account Settings

Language: English
Currency: USD
Time Zone: Eastern Standard Time

Navigation: Home > My Account > My Profile > My Profile

My Profile

Profile Information

First Name: [Text Field]

Last Name: [Text Field]

Email Address: [Text Field]

Phone Number: [Text Field]

Address: [Text Field]

City: [Text Field]

State: [Text Field]

Zip: [Text Field]

Country: [Text Field]

Profile Summary

Profile Information

First Name: [Text Field]

Last Name: [Text Field]

Email Address: [Text Field]

Phone Number: [Text Field]

Address: [Text Field]

City: [Text Field]

State: [Text Field]

Zip: [Text Field]

Country: [Text Field]

Navigation: Home > My Account > My Profile > My Profile

My Profile

Profile Information

First Name: [Text Field]

Last Name: [Text Field]

Email Address: [Text Field]

Phone Number: [Text Field]

Address: [Text Field]

City: [Text Field]

State: [Text Field]

Zip: [Text Field]

Country: [Text Field]

Profile Summary

Profile Information

First Name: [Text Field]

Last Name: [Text Field]

Email Address: [Text Field]

Phone Number: [Text Field]

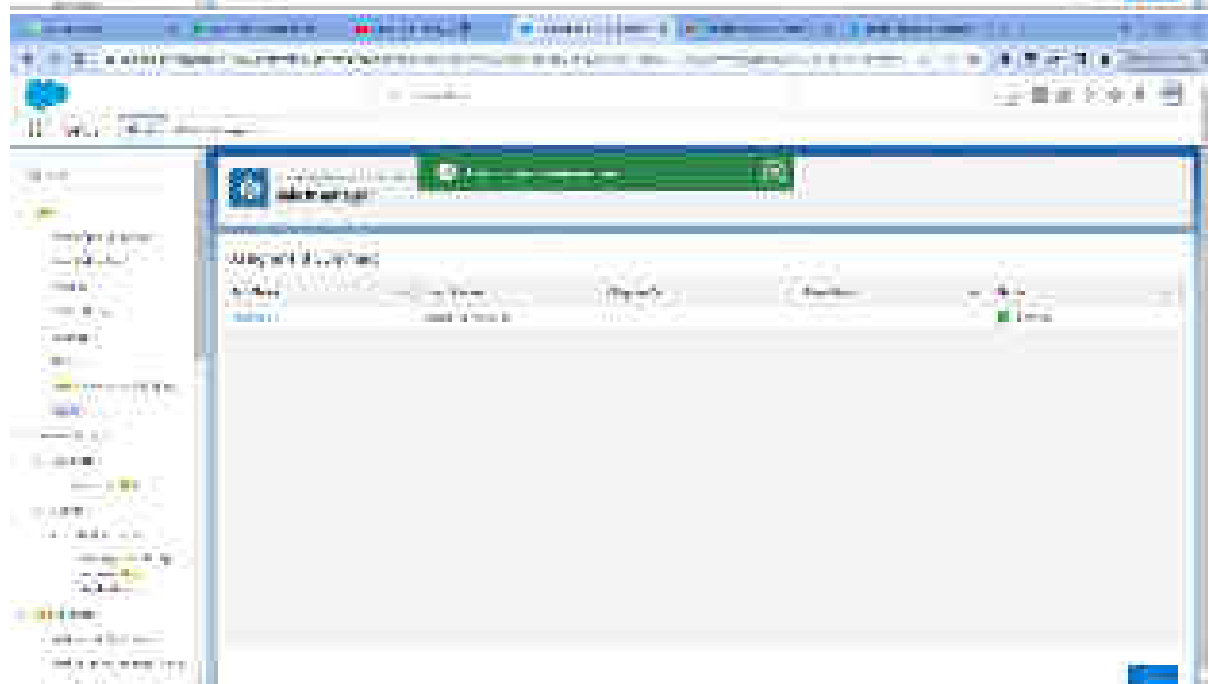
Address: [Text Field]

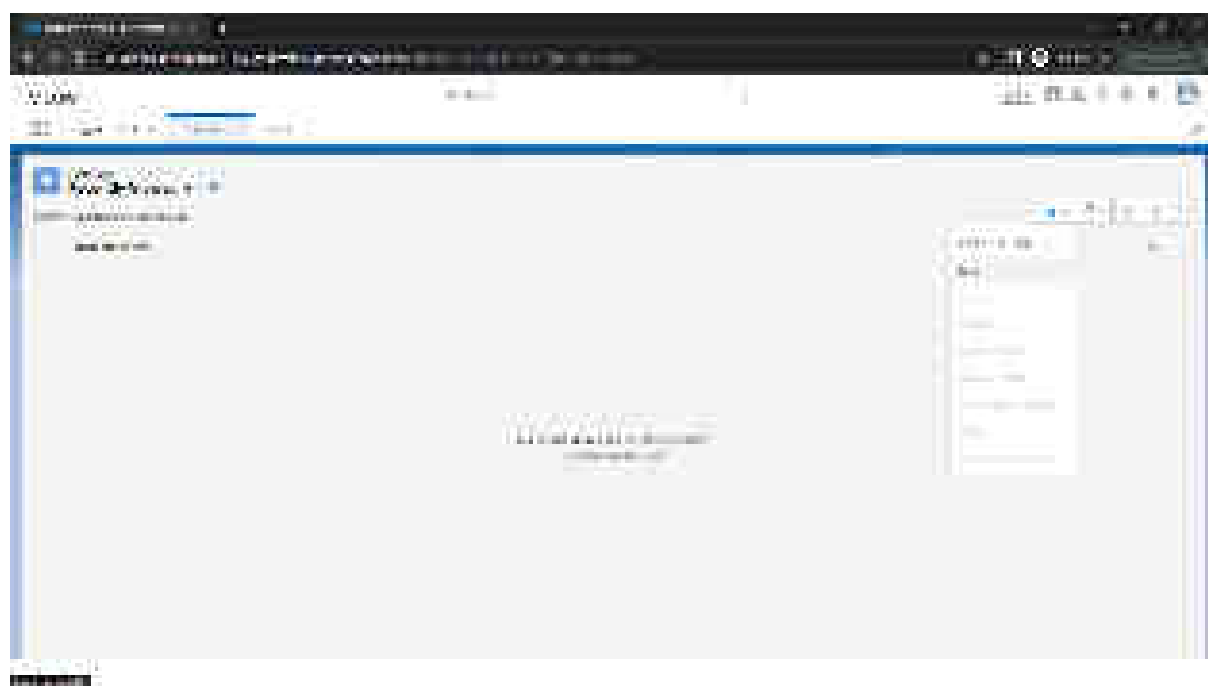
City: [Text Field]

State: [Text Field]

Zip: [Text Field]

Country: [Text Field]



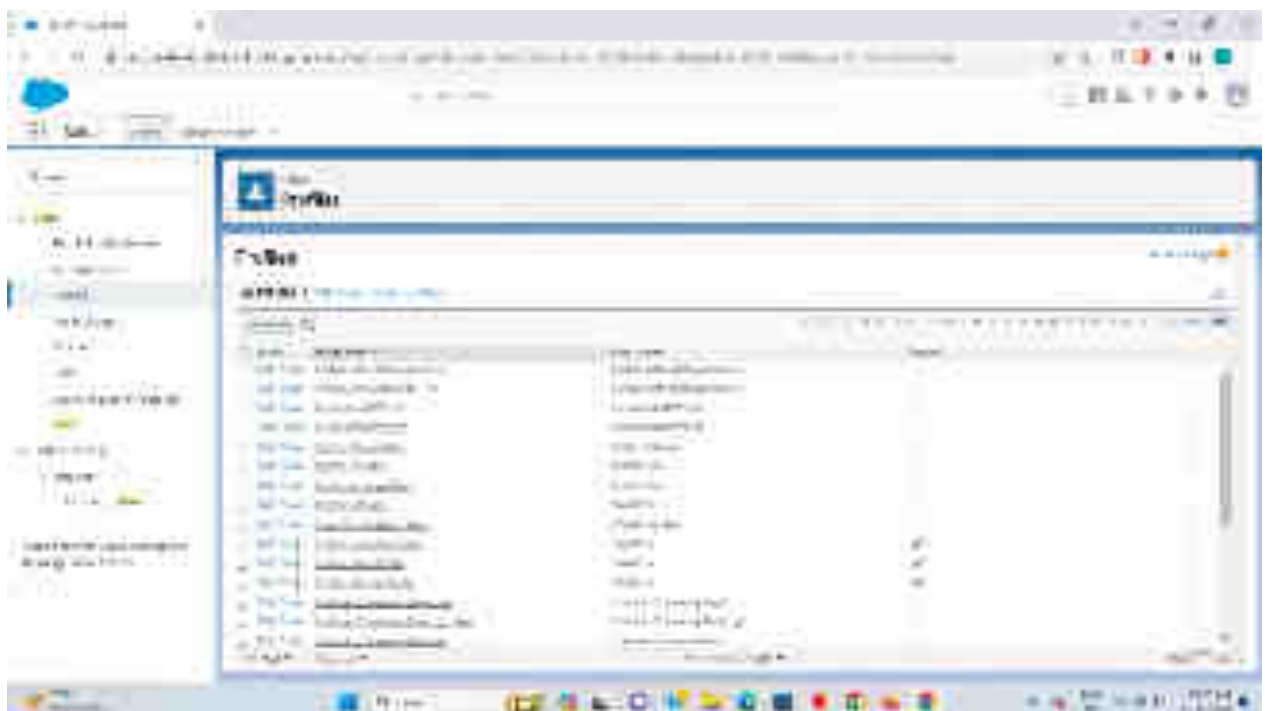


3. . Suppose there are 2 Users and they are having Create, Read, Edit access on Account Object with the same profile but we want to open up the access for one user to delete how will you implement the Security setting.

Solution:

Step 1: we need create a profile for the two user which has the access to Create, Read, Edit for follow as per.

Setup-quick search[profile]



Step 2:

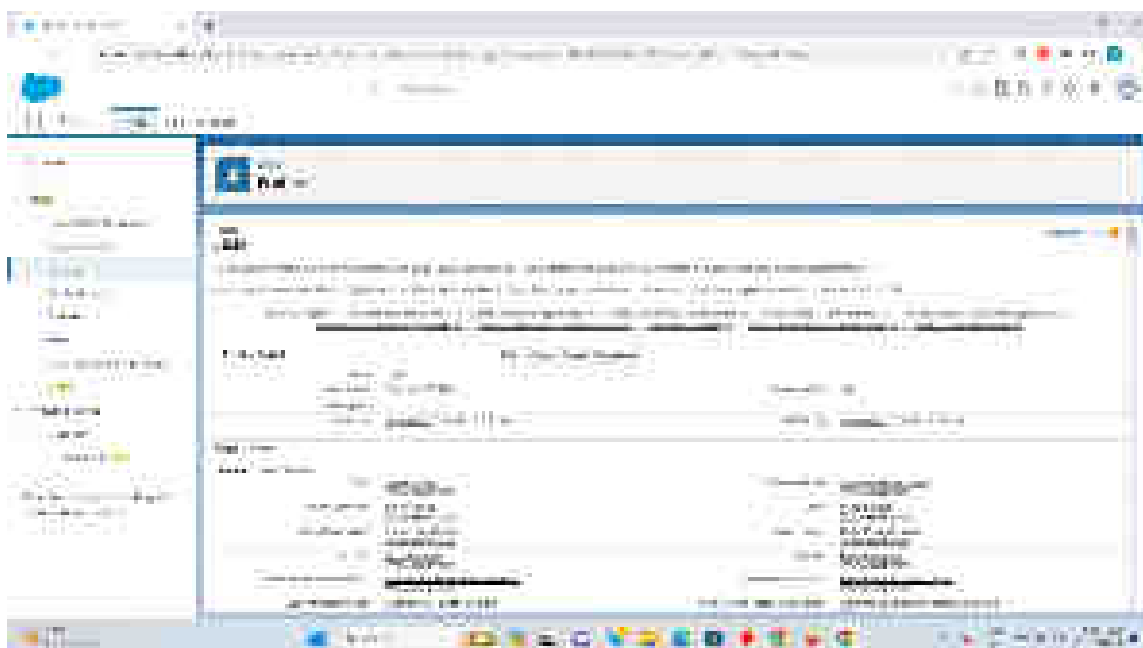
Click on the new to create a new profile along with the label and Api

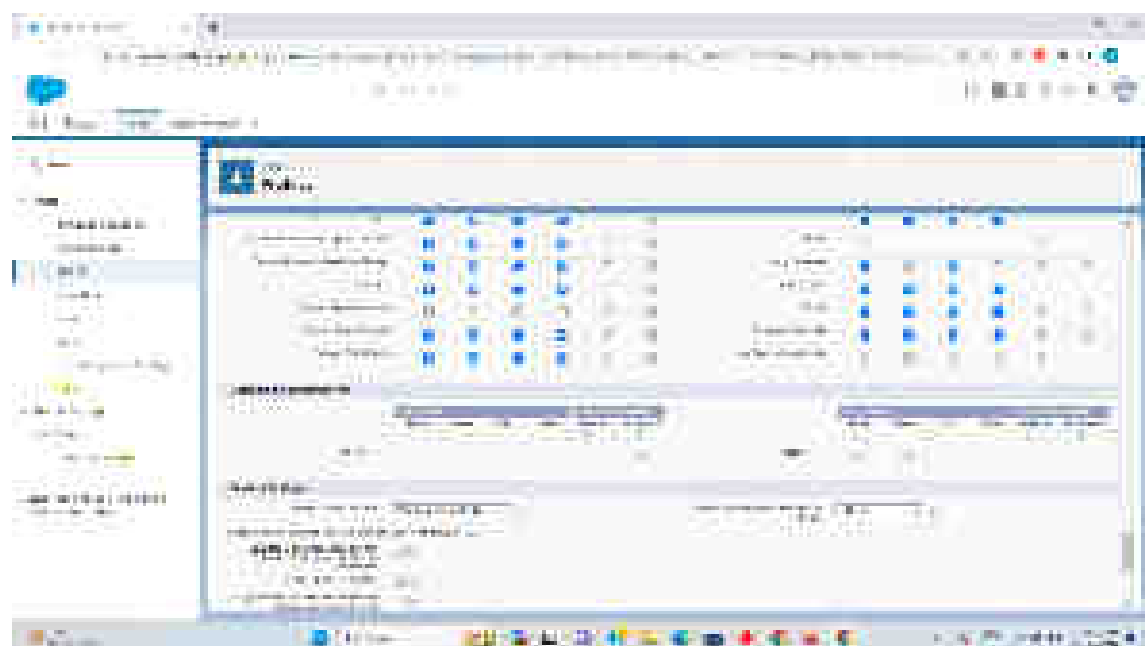


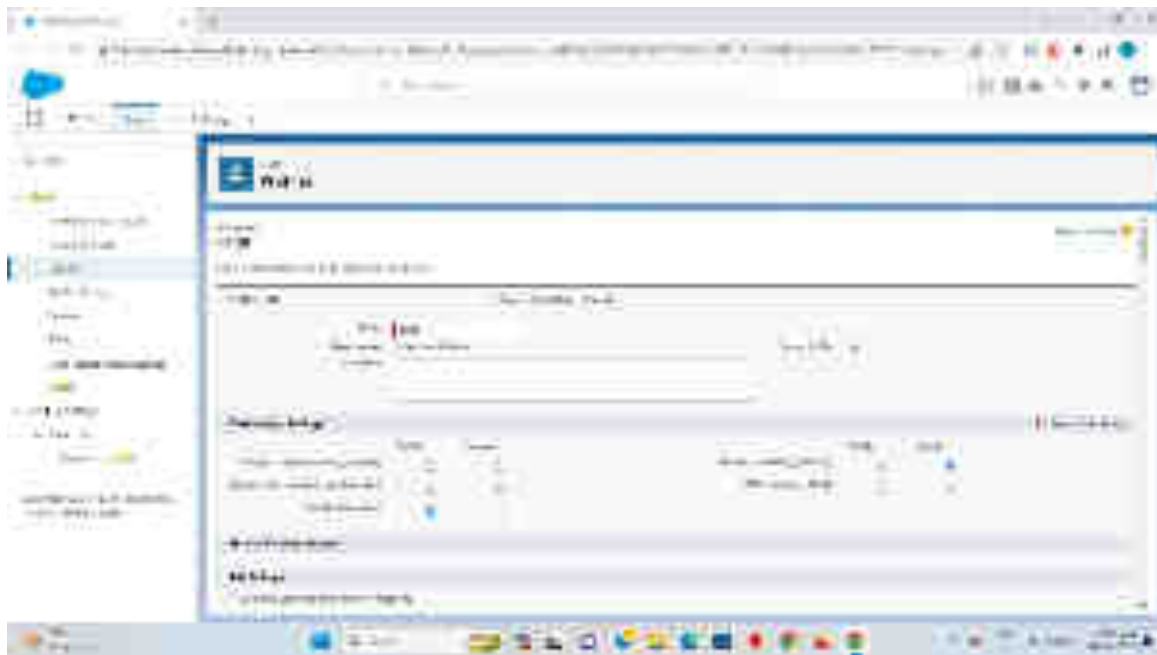
Here I had made it my profile name as Jaga and the existing profile as Standard Platform User.

Step 3:

Now click on the edit and scroll down to custom object settings and enable the read,create,edit and view options. After that click on save.

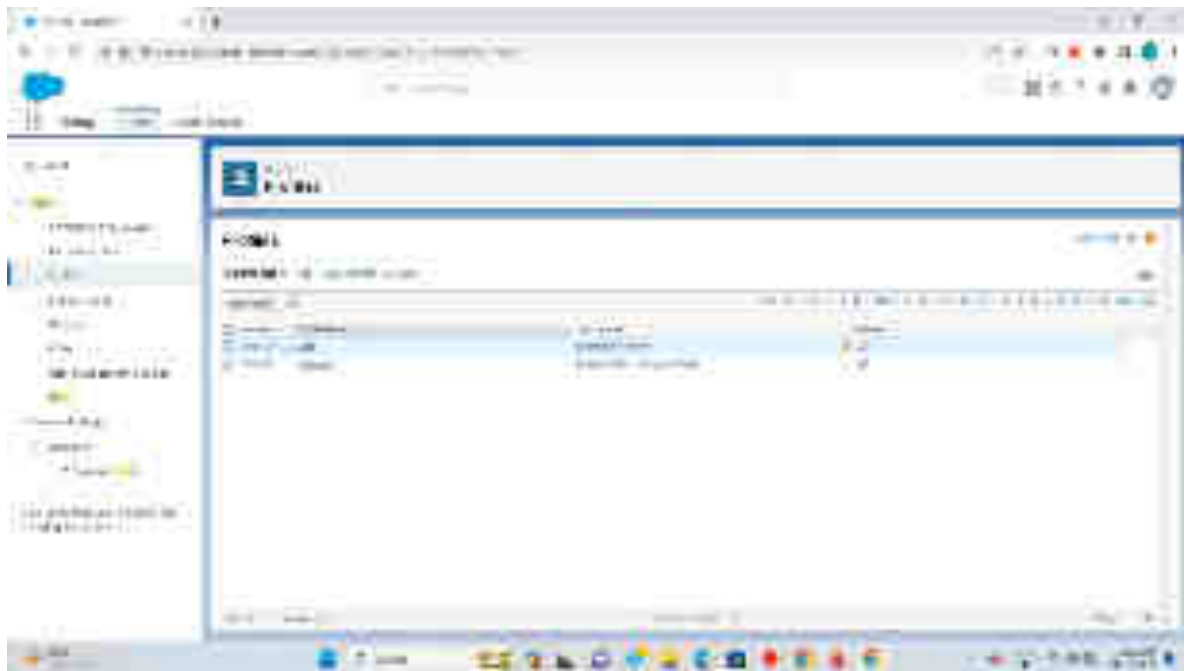






Step 4

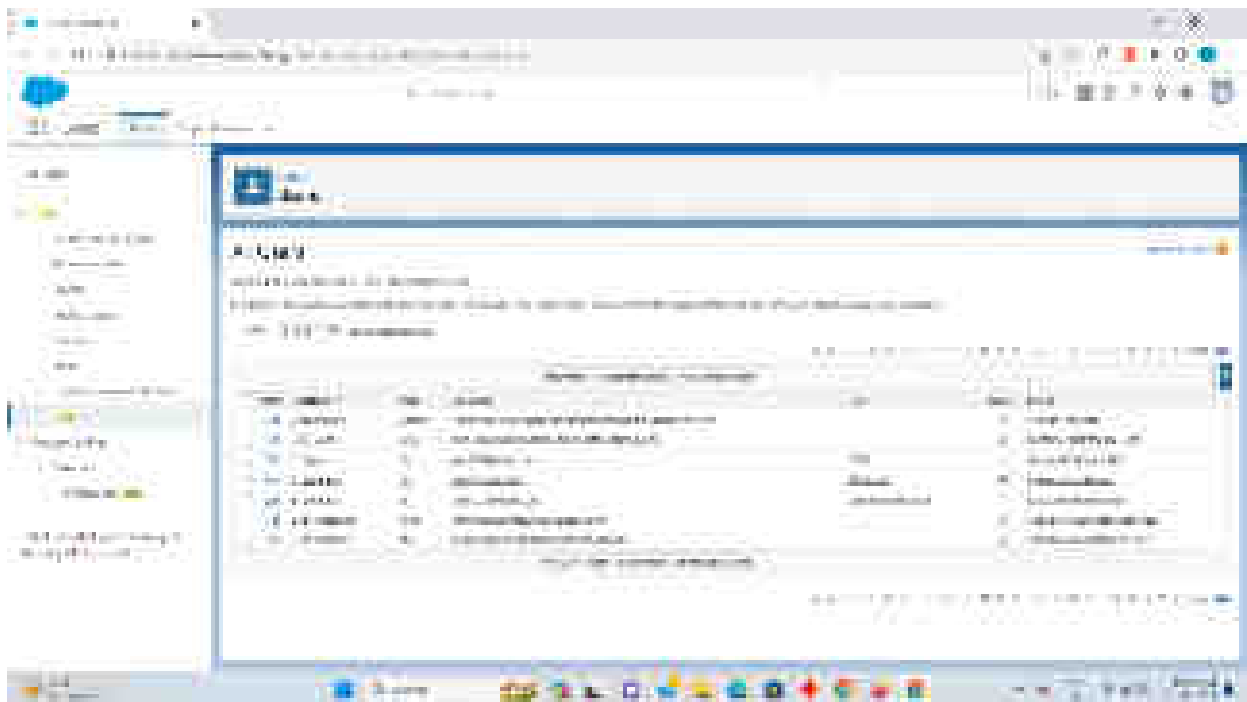
Now you can preview your created profile on the profile option here my profile name jaga has been created with the access of read,create,edit along with view on it

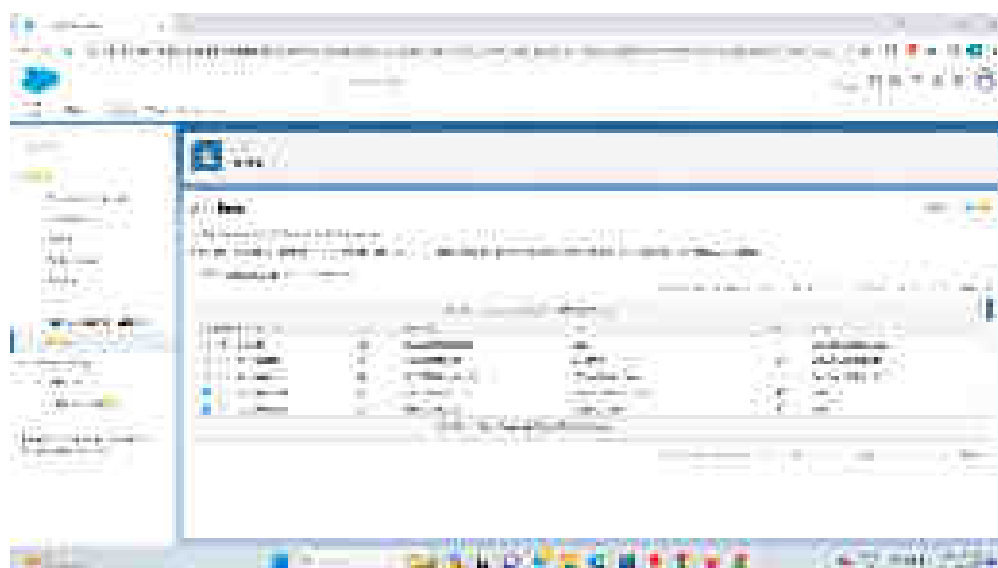
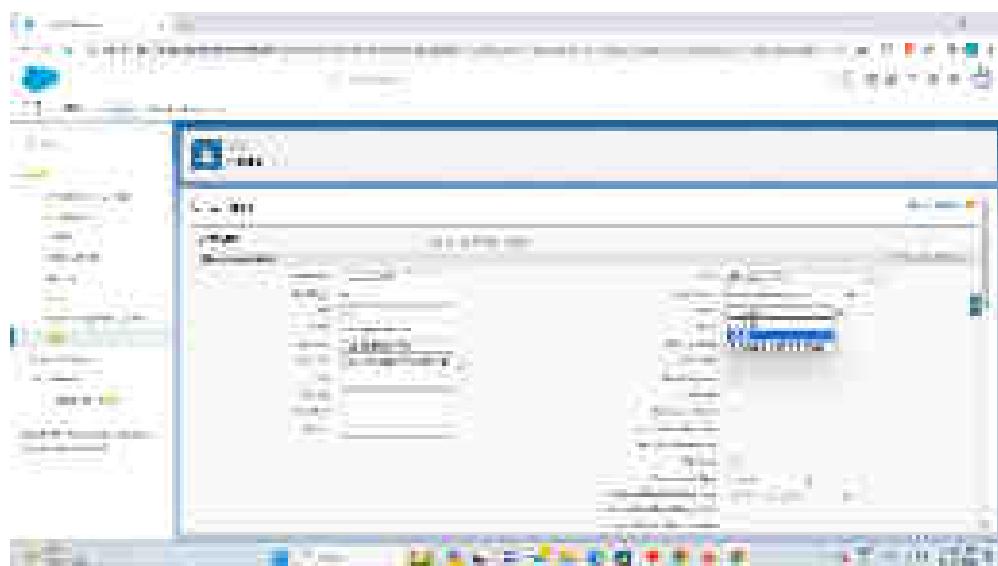
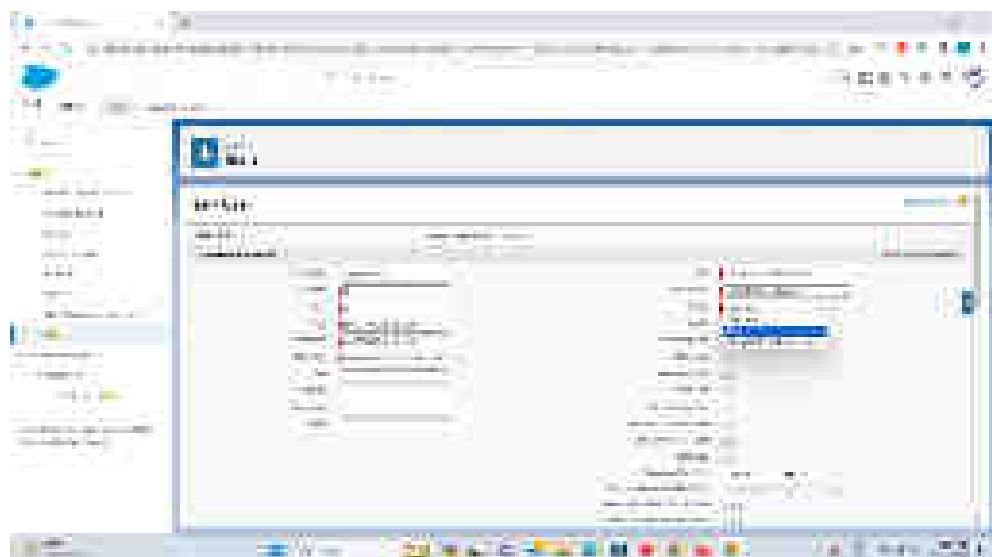


Step 5:

Now create two users by enter into the Setup-quick search[user] and then click on new user after clicking that you need to create two user along

with the profile as Jaga which we have created on the step 2. once the one user has been created click on the save&new so that you can create the second user and there the user name can be created with alternate name but with the same user profile and once the two user are create click on save.



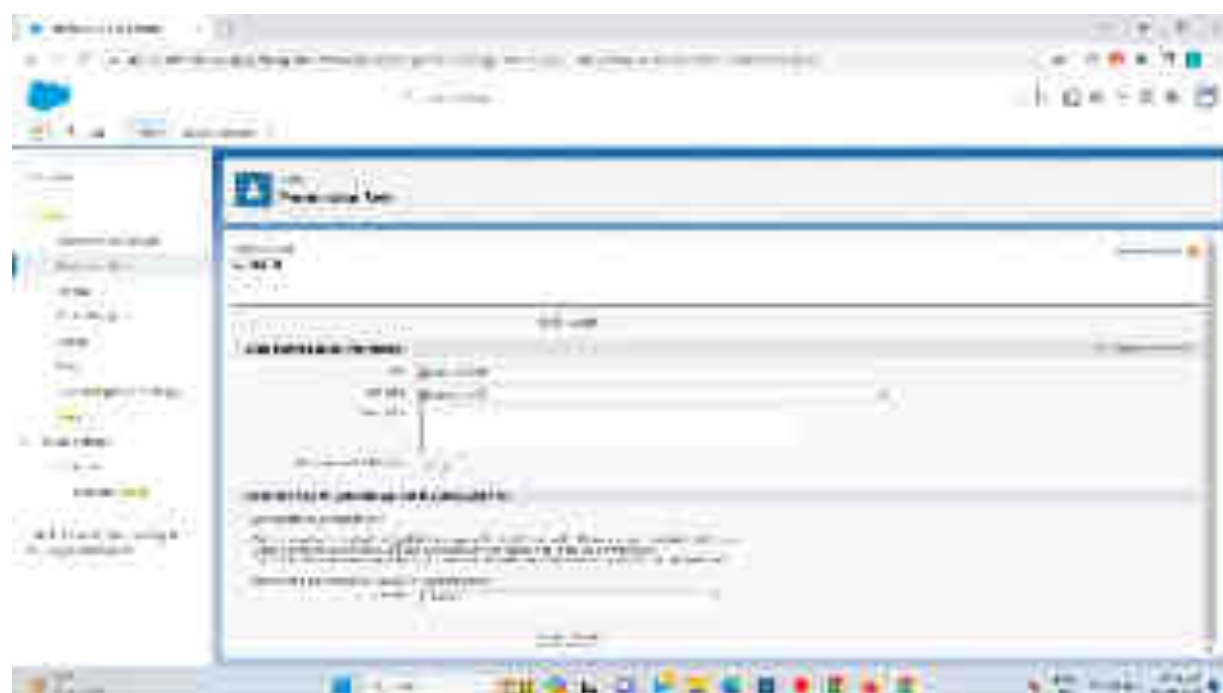


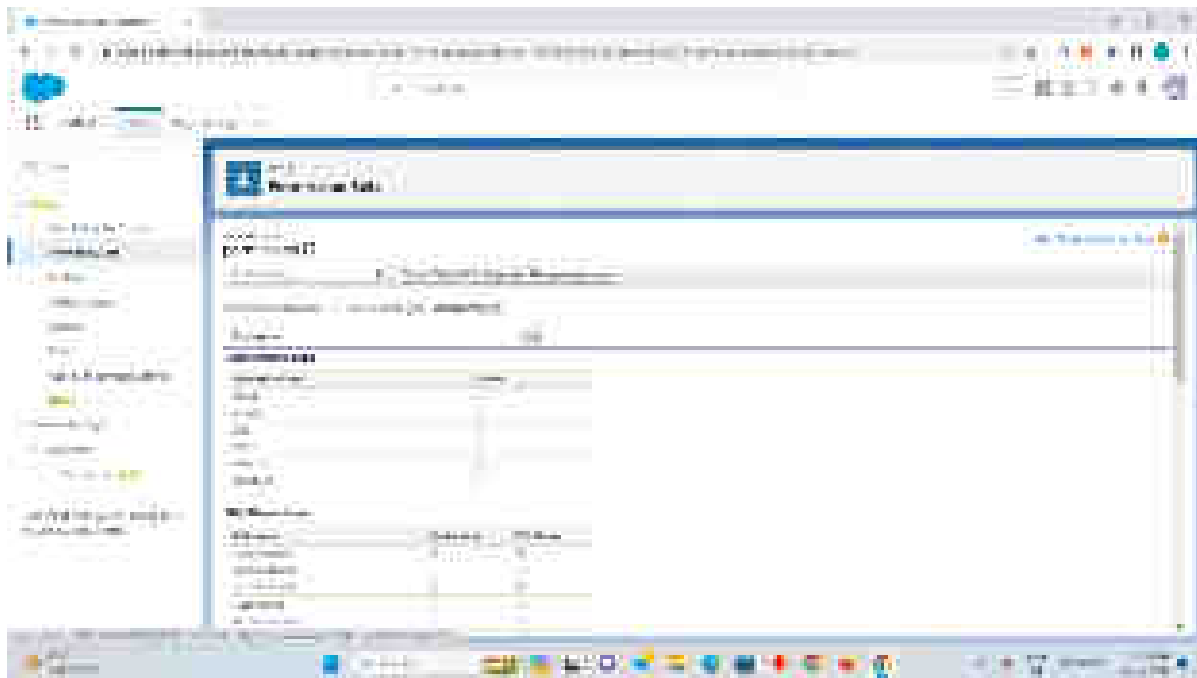
Now you can preview your two user that you have created in my side I had create the two users a Jagadesh11 and Jagadesh22 as a director channel sales with the marketing team.

Step 6:

Now the two user as been created with the profile so that two user can perform the Create, Read, Edit and view on both the user. So as per the given task we need to allocate a specific access as delete on one user for that we need create a permission set for one user so it can created as

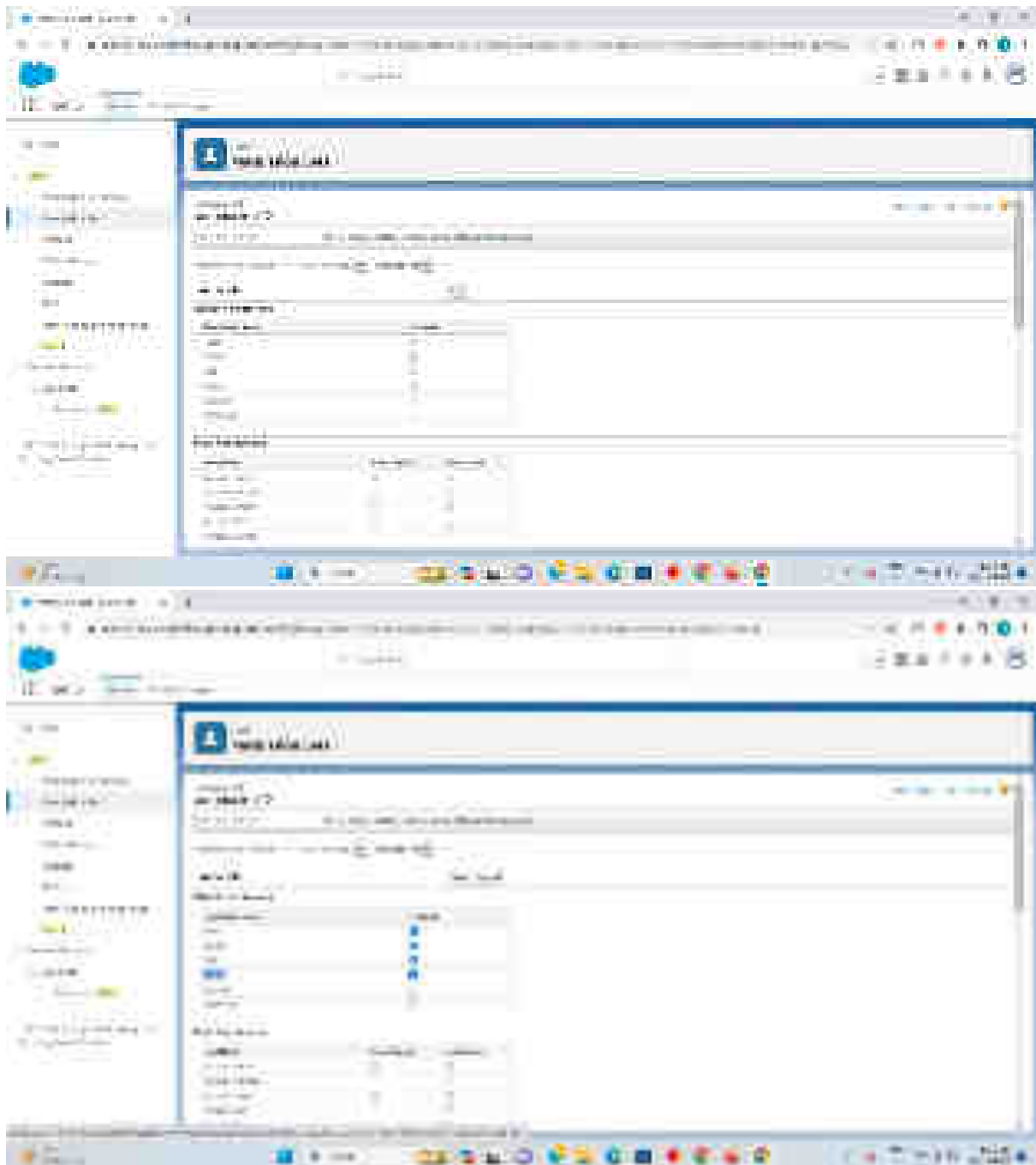
setup-quick search[permission set]-new-fill label name [auto select the API name]-click on save-object settings-accounts.





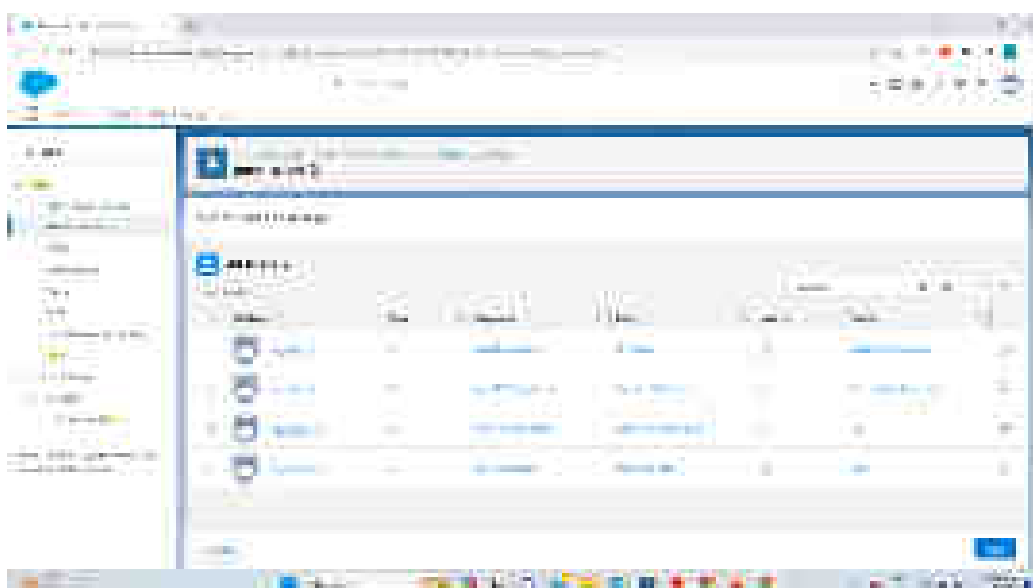
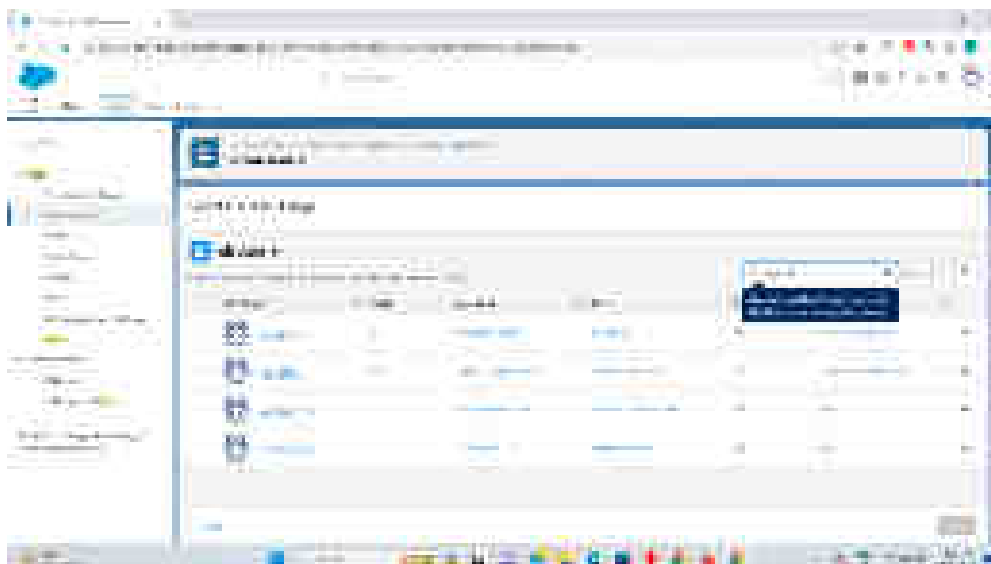
Step 7:

Now to give the specific delete access to the user click on edit on the Account and then enable the read,create,edit and the delete on it so that the permission set will have a specific special access on it. once it has been done click on save and then click on manage assignment.

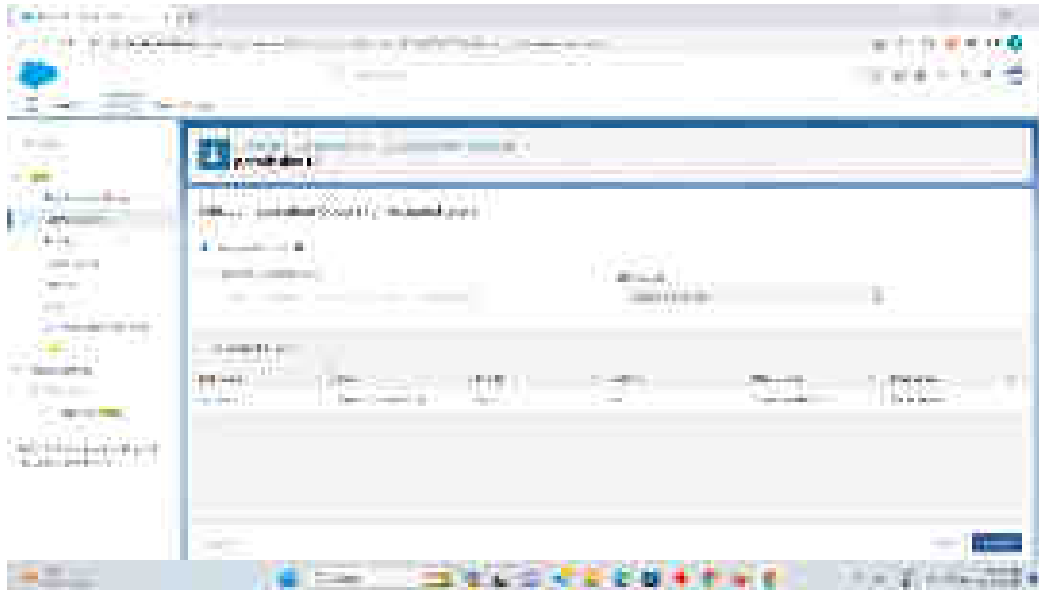


Step 8

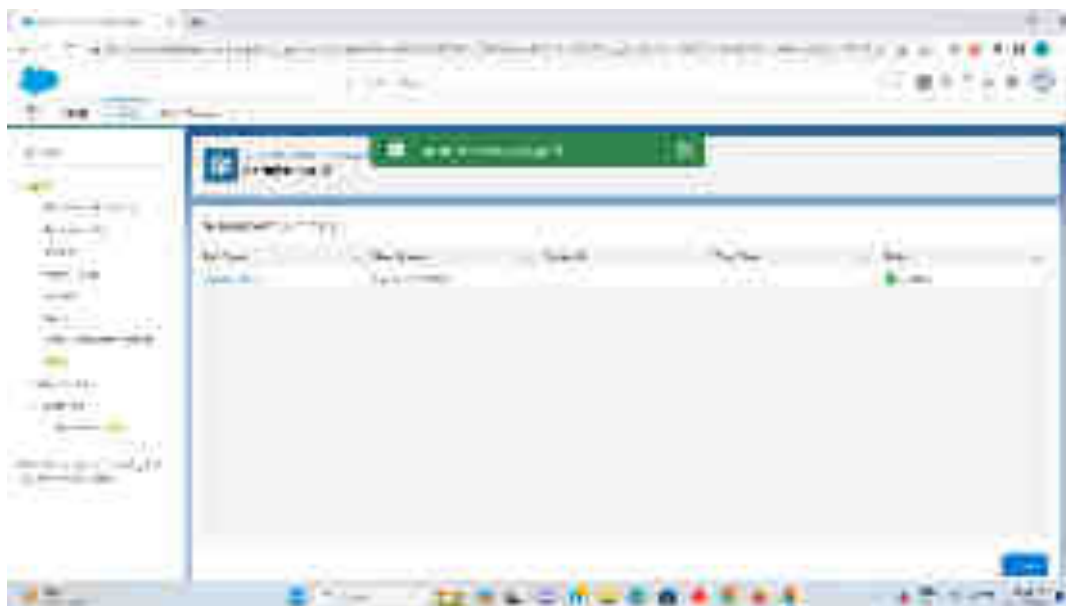
Now click on add assignment there you will find your two created users click on any one user to give a special access as delete on it and then click on assign so that the specific selected user can have a special access as delete on it.



Click on next.



Now click on Assign.



Now the specific access for the Jagadesh11 user has been assigned successfully.

4. Create a screen flow for a basic survey to fill in the details for any form.

Solution:

Step 1: Create a Custom Object

The next step is to create a custom object **Survey Result** and a few custom fields to store survey responses.

1. Click **Setup**.
2. In the Object Manager, click **Create | Custom Object**.
3. Now create a custom object **Survey Result** and fields as shown in the screenshot below:
4. Click **Save**.

SETUP > OBJECT MAPPER

Survey Result

Details

Fields & Relationships

Page Layouts

Uploading Record Pages

Metadata, Links, and Actions

Component Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

Search Layouts for Standard or Classic

Triggers

Validation Rules

Fields & Relationships

Quick Find

8 Columns Selected (by name)

New

Related Fields

Field Dependencies

Get History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Comments	Comments__c	Text Area (255)		
Created By	CreatedBy	Lookup (User)		
Email	Email__c	Email		
Last Modified By	LastModifiedBy	Lookup (User)		
Name	Name__c	Text (255)		
Owner	OwnerId	Lookup (User, Group)		
Rating	Rating__c	Number		
Summary/Detail Name	Name	Auto Number		

Step 2: Create a Thank You For Survey Lightning Email Template

1. Click **App Launcher**.
2. In the Quick Find box, type **Email Templates**.
3. Clicks on the **New Email template** button.
4. Name the **Lightning Email Template** and make sure to store it in the **Public Email Templates** folder.
5. Create a template like the following screenshot.

Email Template Builder

Thank You Email - Survey

View in Builder

Edit

Clone

✕

Details

Related

Information

Email Template Name	Thank You Email - Survey	Related Email Type	Survey Result
Exception		Owner	
Made in Email Template Builder		Public Email Template	

Message Content

Subject	Thank you for completing our survey		
From Name			
	Hi {{Survey_Result__Name__}},		
	<p>Thanks for taking time out to participate in our survey. We are very appreciative of the time you have taken to assist in our analysis, and commit to utilizing the information gained to enhance our services and implement significant improvements. We will share these results with you through your Rickie Ramsey Agency, whom we also thank for their generous participation.</p> <p>Once again, we are extremely grateful for your contributing your valuable time, your honest information, and your thoughtful suggestions.</p> <p>Thanks, Johnston Thompson</p>		

Additional Information

Created By	Created By
Rajesh Gupta, 11/21/2020, 4:22 PM	Rajesh Gupta, 12/21/2020, 4:22 PM

Step 3: Create an Email Alert

1. Click **Setup**.
2. In the Quick Find box, type **Email Alerts**.
3. Select **Email Alerts**, click on the **New Email Alert** button.
4. **Name** the **Email Alert** and click the Tab button. The **Unique Name** will populate.

5. For **Object** select **Survey Result**.
6. For the **Email Template** chooses **Lightning Email Template Thank You Email – Survey**.
7. For **Recipient Type** select **Email Field:**
Email.
8. Click **Save**.

The screenshot shows the 'Edit Email Alert' page in Salesforce. The title is 'Survey - Thank You Email'. Below the title is a description: 'Create an email alert to associate with one or more workflow rules, approval processes, or entitlement processes. When creating an email alert, any workflow rule will apply to all data, approval, or entitlement processes associated with it.' The page has three buttons at the top: 'Save', 'Save & Print', and 'Cancel'.

The main section is titled 'Edit Email Alert' and contains several fields:

- Description:** 'Survey - Thank You Email'
- Unique Name:** 'Survey_Thank_You_Email' (with a dropdown arrow)
- Object:** 'Survey Result'
- Email Template:** 'Thank You Email - Survey' (with a dropdown arrow)
- Protected Component:** ☐
- Recipient Type:** 'Email Field: Email' (with a dropdown arrow and a 'Find' button)
- Recipients:** This section has two columns: 'Available Recipients' and 'Selected Recipients'.
 - Available Recipients:** Lists 'User Information: User', 'User: Role in Queue', and 'User: Security Role'.
 - Selected Recipients:** Lists 'Email Field: Email'.
 - Between the columns are buttons: 'Add', 'Remove', and 'Cancel'.
- Additional Recipients:** A text area with the note 'You can enter up to five (5) email addresses to be notified.' Below it is a 'From Email Address' field with the value 'Current User's email address' and a checkbox to 'Make this address the default From email address for this object's email alerts.' There is a 'Find' button next to the checkbox.

At the bottom of the page are three buttons: 'Save', 'Save & Print', and 'Cancel'.

Step 4.1: Salesforce Flow — Create a Screen that Allow Users to Fill Survey

1. Click **Setup**.

2. In the Quick Find box, type **Flows**.
3. Select **Flows** then click on the **New Flow**.
4. Select the **Screen Flow** option and click on **Next** and configure the flow as follows:
 1. **How do you want to start building: Freeform**
5. We will use the **Screen** element to capture a **Survey response** form. Drag and drop a **Screen** element onto the canvas.

Step 4.2: Salesforce Flow — Add a Record Creates Element to Save Survey Response

1. Drag-and-drop the **Create Records** element onto the Flow designer.
2. Enter a name in the **Label (Save Response)** field; the **API Name** will auto-populate.
3. For **How Many Records to Create** – select **One**.
4. For **How to Set the Record Fields** – select **Use separate resources, and literal values**.
5. Select the **Survey_Result__c** object from the dropdown list.
6. **Set Field Values for the Survey Result**
 1. Row 1:
 1. **Field: Comment__c**

2.Value: {!Comment}

2.Click Add Row

3.Row 2:

1.Field: Email__c

2.Value: {!Email.value}

4.Click Add Row

5.Row 3:

1.Field: Name__c

**2.Value: {!Name.firstName}
 {!Name.lastName}**

6.Click Add Row

7.Row 3:

1.Field: Rating__c

2.Value: {!Rating}

7.Click Done.

Edit Criteria Record

Create Survey Results using values from this Form

First Name: Last Name:

Comments:

Form Setup Instructions for Users

☒ One ☐ Multiple

How to set the record fields

☐ Manually enter field values ☒ Use separate resources and formulas

Create a Record of This Object

Object:

Get Field Values for the Survey Results

Field	Value
Comments__c	<input type="text" value="Comments__c"/>
Image__c	<input type="text" value="Image__c"/>
Field	<input type="text" value="Survey Results (Survey Results)"/>
Rating__c	<input type="text" value="Rating__c"/>

[Add Field](#)

☐ Manually enter values

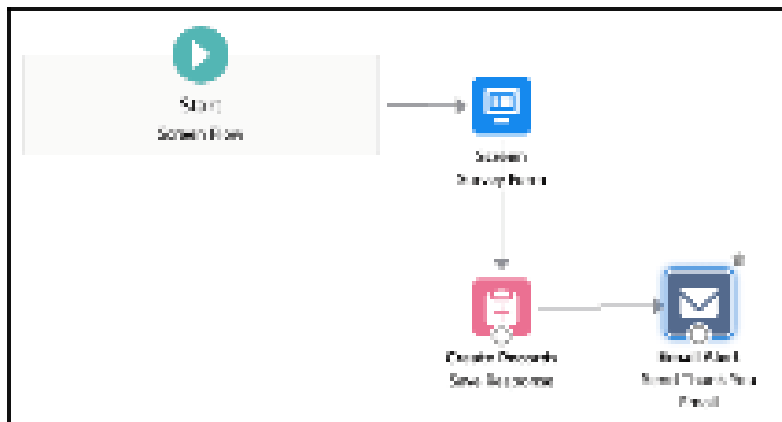
Step 4.3: Salesforce Flow — Call an Action — Email Alert to Send Out Thank You Email

The next step is to call the **Survey – Thank You Email** email alert from flow so that when flow fires it triggers the thank you email to survey participants.

1. Under **Toolbox**, select **Element**.
2. Drag-and-drop **Action** element onto the Flow designer.
3. In the **Action** box, type **Survey – Thank You Email**.

4. Clicks on the **Survey – Thank You Email** email alert.
5. Click **Done**.

In the end, Sergio's **Flow** will look like the following screenshot:



1. Click **Save**.
2. Enter **Flow Label** the API Name will auto-populate.
3. Click **Show Advanced**.

4. **How to Run the Flow: User or System Context—Depends on How Flow is Launched**
5. **Type: Screen Flow**
6. **API Version for Running the Flow: 51**
7. **Interview Label: Survey
{!\$Flow.CurrentDateTime}**
8. **Click Save.**

Save as

A New Version

A New Flow

Flow Label

Survey

Flow API Name

Survey

Description

Hide Advanced

How to Run the Flow ⓘ

User or System Context—Depends on How Flow is Launched

Type

Screen Flow

API Version for Running the Flow

51

Interview Label ⓘ

Insert a resource...

Survey (!\$Flow.CurrentDateTime)

Last Modified

12/21/2020, 4:54 PM by Rabeen Gupta

Status

Active

Type

Screen Flow

Version Number

3

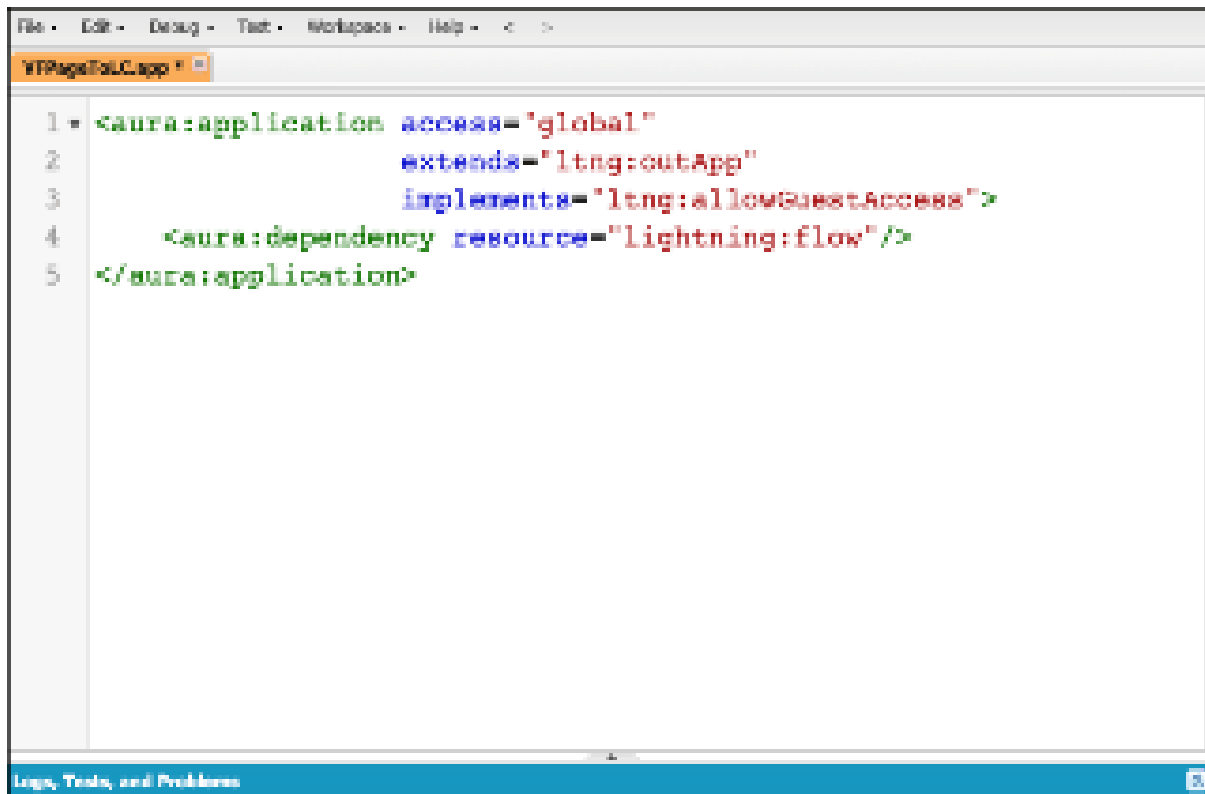
Cancel

Save

Step 5: Create a Lightning Application to Render Lightning Runtime for Flow in a Visualforce Page

Now we will create a Lightning Application that declares a dependency on the **lightning:flow** component.

1. Click **Setup | Developer Console**
2. Navigate to **File | New | Lightning Application**
3. Enter a **Name (VFPageToLC)** field, make sure to select the **Lightning Out Dependency App** checkbox.
4. Click **Submit**.
5. Copy code from **GitHub** and paste it into your Lightning Application.
6. **Save** your code.

A screenshot of an IDE window titled 'VFPPageForCApp'. The editor shows the following XML code:

```
1 <aura:application access='global'
2     extends='ltng:outApp'
3     implements='ltng:allowGuestAccess'>
4     <aura:dependency resource='lightning:flow' />
5 </aura:application>
```

The bottom status bar indicates 'Logs, Tests, and Problems'.

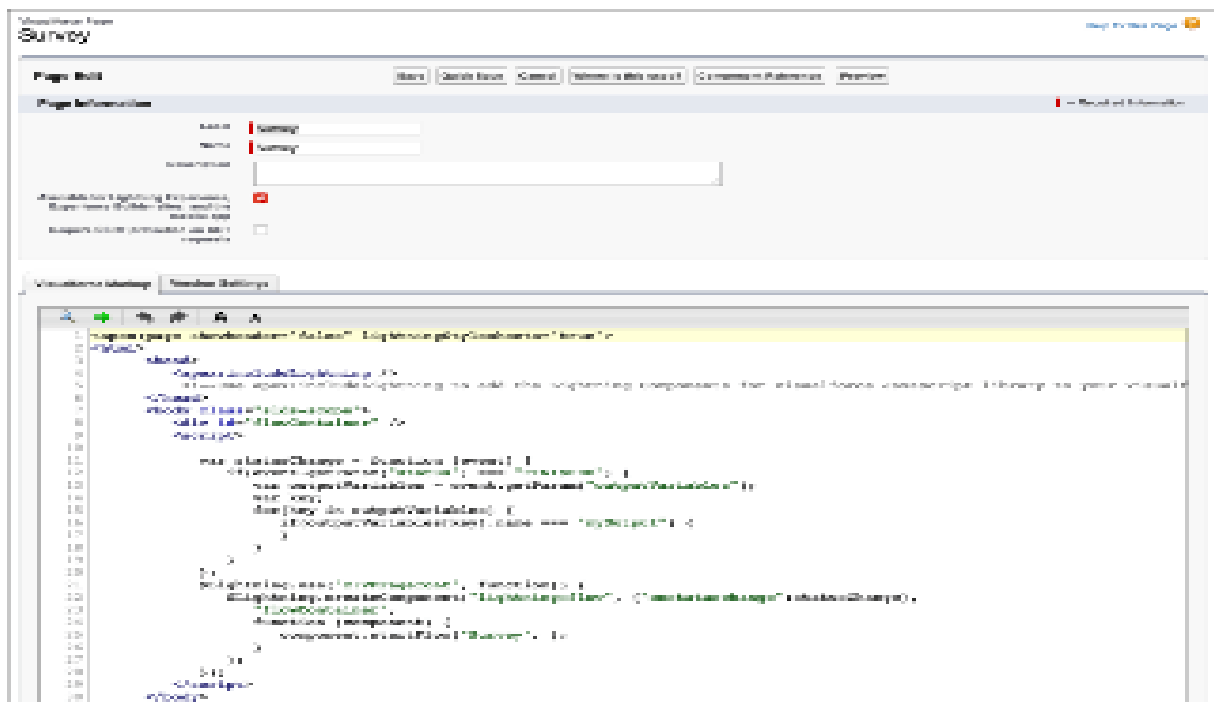
Step 6: Create a Visualforce Page and Embed Your Flow Into It

Now we will create a Lightning Application that declares a dependency on the **lightning:flow** component.

Add the Lightning Components for Visualforce JavaScript library to your Visualforce page using the **<apex:includeLightning/>** component. In the Visualforce page, reference the dependency app. Then write a JavaScript function that creates the

component on the page
using **\$Lightning.createComponent()**

1. Click **Setup**.
2. In the Quick Find box, type **Visualforce Pages**.
3. Clicks on the **New** button.
4. Copy code from **GitHub** and paste it into your visualforce page
5. Click **Save**.



Step 7: Create a Force.com Site to Open Your Flow for Unauthenticated Access

Now we will create a site to open the flow for unauthenticated access.

- ## 1.Click Setup.

After successful submission, he/she will receive an email.