GARAGE MANAGEMENT SYSTEM

COLLEGE NAME : TIRUPPUR KUMARAN COLLEGE FOR WOMEN

COLLEGE CODE : Bru3j

TEAM ID : NM2025TMID28359

TEAM MEMBERS : 4

TEAM LEADER NAME : MEGARAJ BEHUM.M

EMAIL : abdullasheik292@gmail.com

TEAM MEMBER : SWATHI.S

EMAIL : swatihswathi@gmail.com

TEAM MEMBER : INDHUMATHI.M

EMAIL : indhumathim138@gmail.com

TEAM MEMBER : RAMYA.J

EMAIL : ramyajramyaj917@gmail.com

GARAGE MANAGEMENT SYSTEM

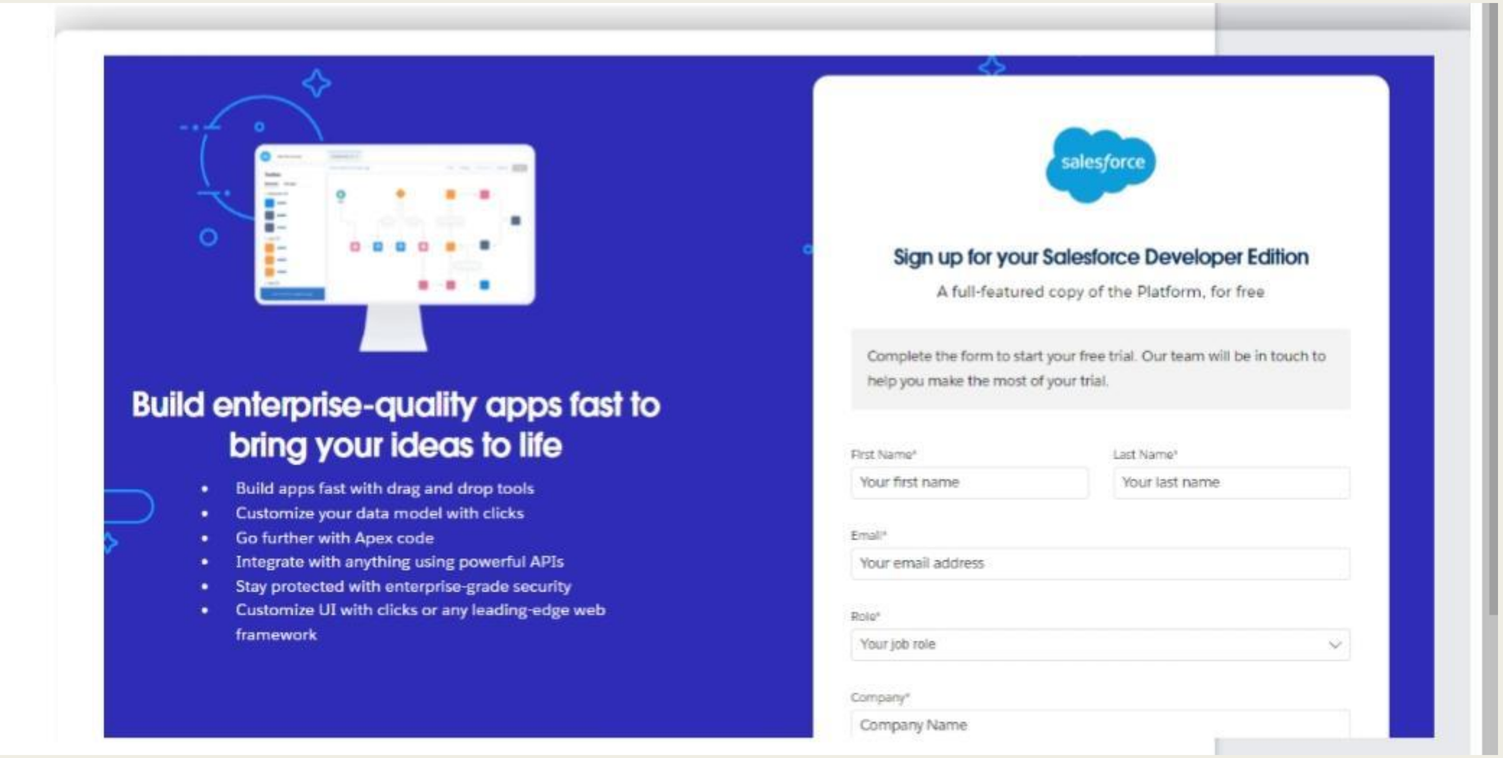
INTRODUCTION

- Garage Management in SalesforceGarage management in Salesforce refers to the process of managing and tracking vehicle maintenance, repairs, and other services within a garage or automotive service center.
- Salesforce can be customized to meet the specific needs of a garage management

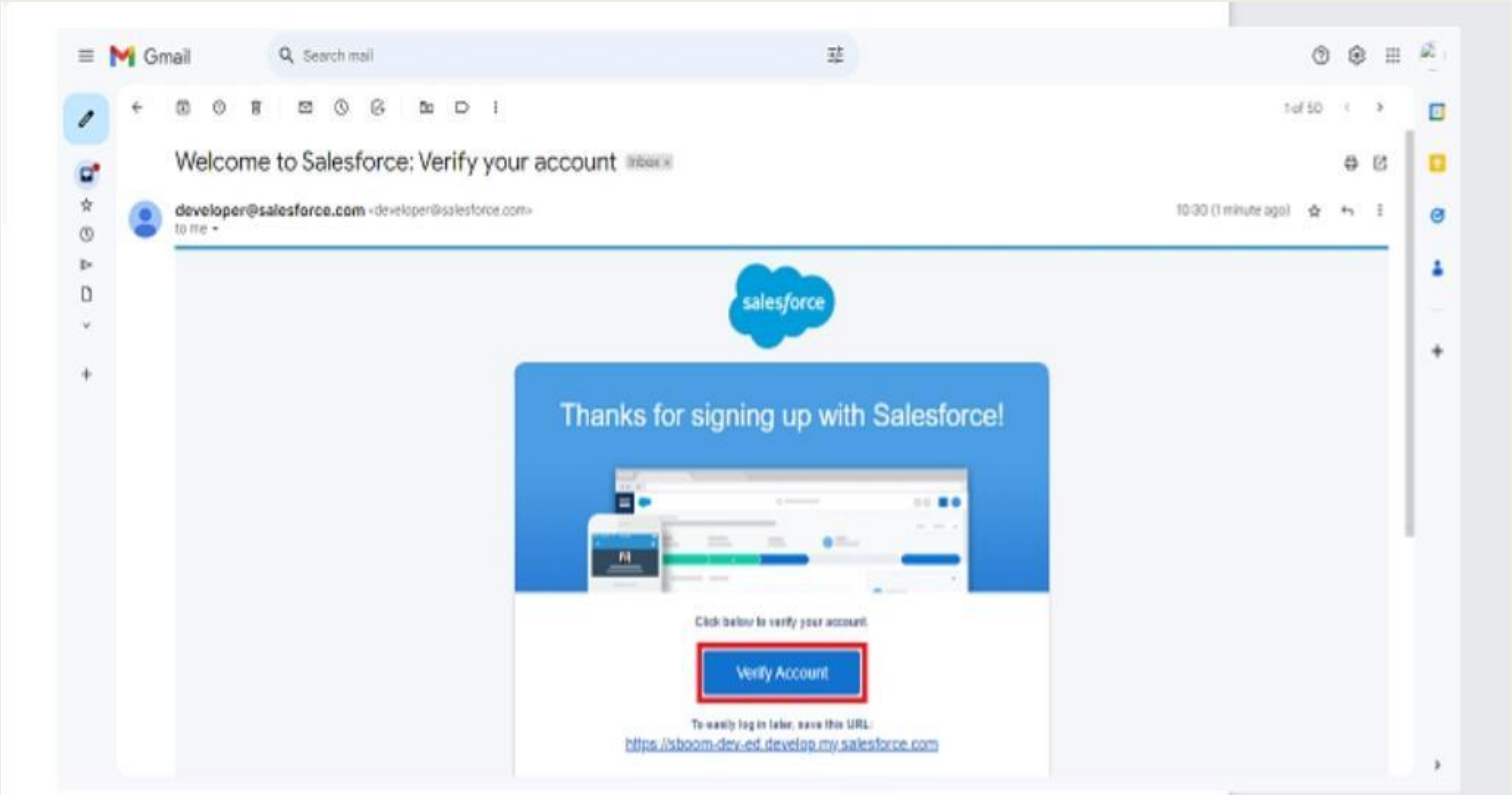
SALESFORCE

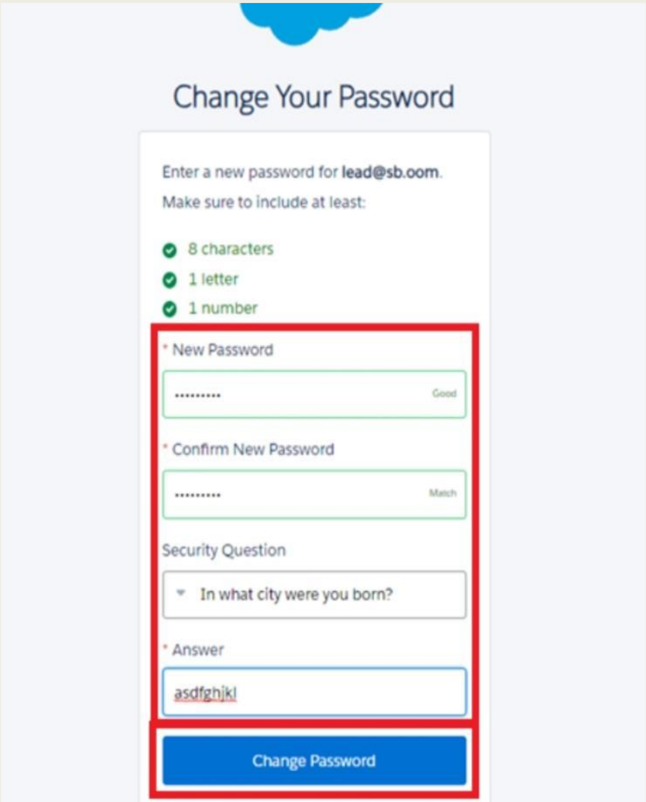
- Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.
- Salesforce has everything you need to run your business from anywhere.
- Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

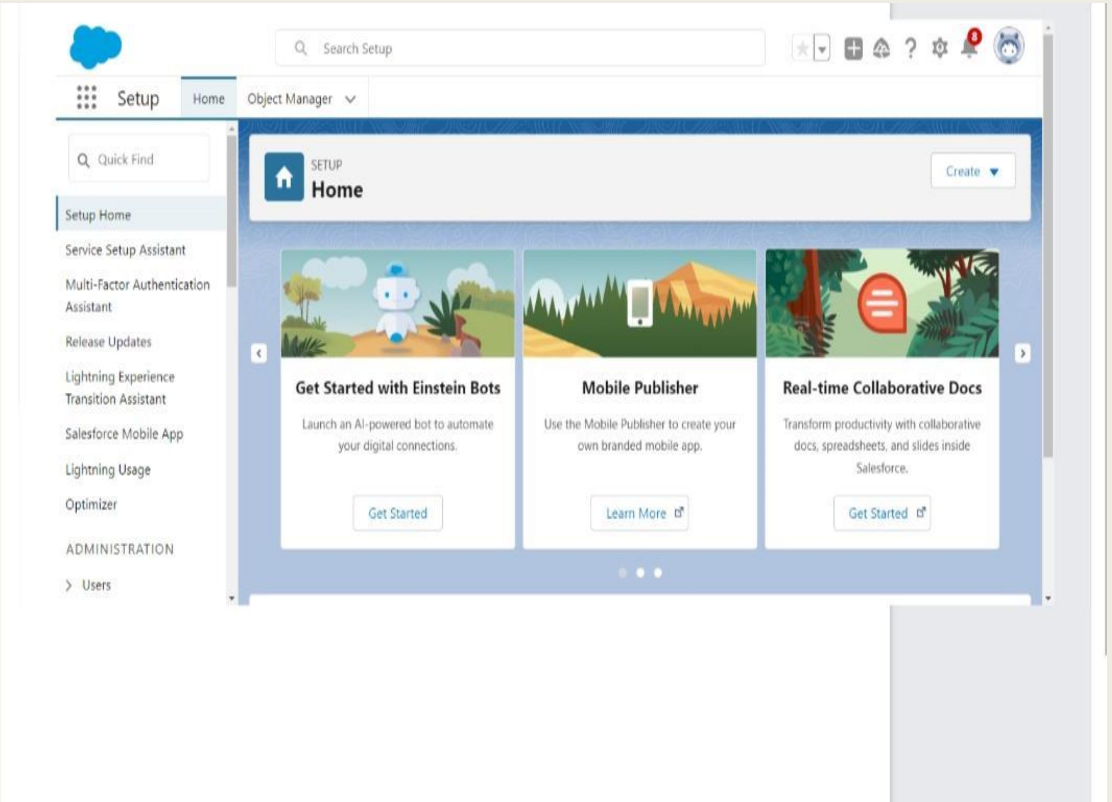
CREATING DEVELOPER ACCOUNT:



ACCOUNT ACTIVATION




Change Your Password
Enter a new password for lead@sb.oom.
Make sure to include at least:
8 characters
1 letter
1 number
* New Password
* Confirm New Password
Security Question
In what city were you born?
* Answer
asdfghjkl
Change Password


Search Setup
Setup
Home
Object Manager
Quick Find
Setup Home
Service Setup Assistant
Multi-Factor Authentication Assistant
Release Updates
Lightning Experience Transition Assistant
Salesforce Mobile App
Lightning Usage
Optimizer
ADMINISTRATION
Users
SETUP
Home
Create
Get Started with Einstein Bots
Launch an AI-powered bot to automate your digital connections.
Get Started
Mobile Publisher
Use the Mobile Publisher to create your own branded mobile app.
Learn More
Real-time Collaborative Docs
Transform productivity with collaborative docs, spreadsheets, and slides inside Salesforce.
Get Started

OBJECT

- Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects
- Salesforce objects are of two types:
- Standard Objects: Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
- Custom Objects: Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure for sharing data.

TABS

- A tab is like a user interface that is used to build records for objects and to view the records in the objects.

Types of Tabs:

1.Custom Tabs

Custom object tabs are the user interface for custom applications that you build in salesforce.com. They look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

2.Web Tabs

Web Tabs are custom tabs that display web content or applications embedded in the salesforce.com window.

3.Visualforce Tabs

Visualforce Tabs are custom tabs that display a Visualforce page. Visualforce tabs look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

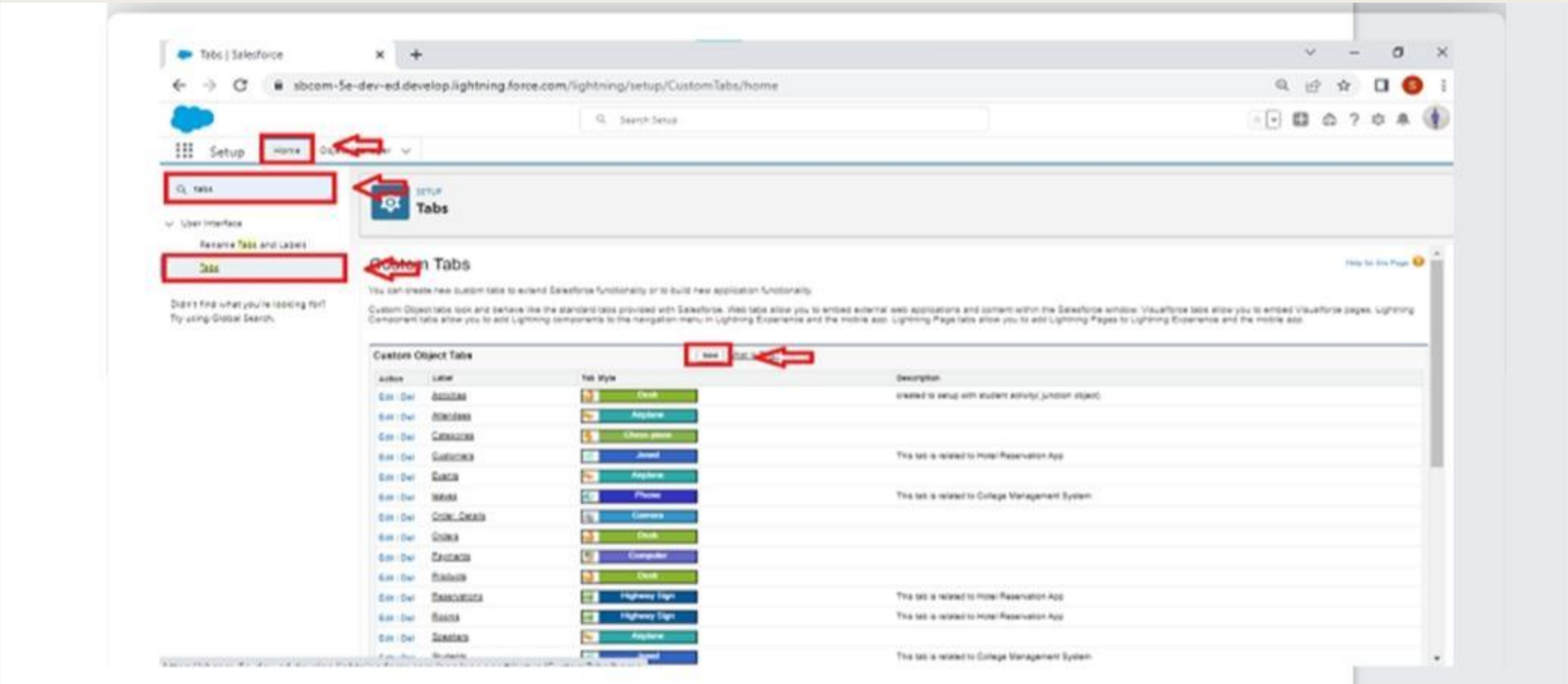
4.Lightning Component Tabs

Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app.

5.Lightning Page Tabs

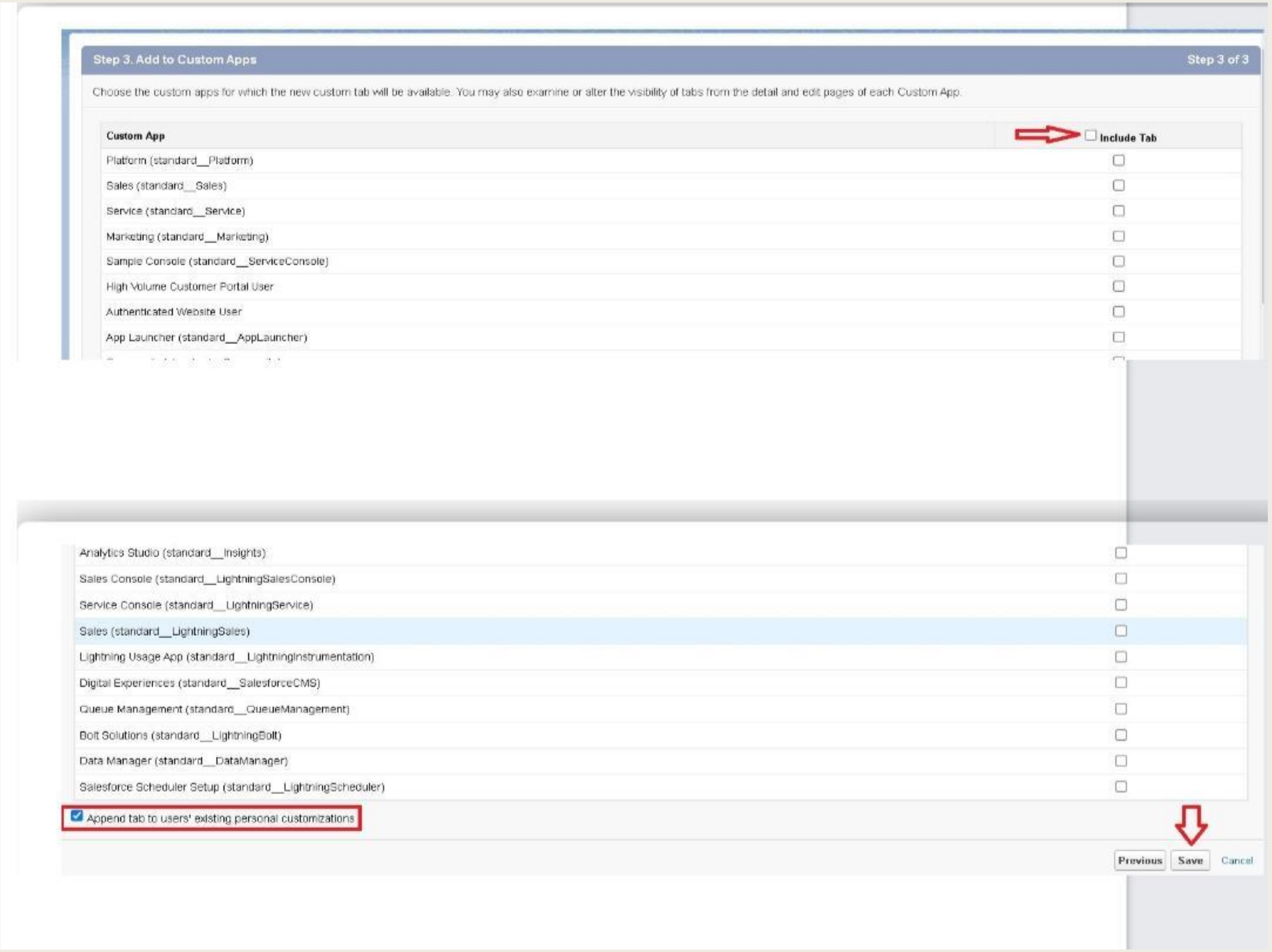
Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu.
Lightning Page tabs don't work like other custom tabs.

Creating A Custom Tab




New Custom Object Tab
Help for this Page
Step 1. Enter the Details
Step 1 of 3
Choose the custom object for this new custom tab. Fill in other details.
Select an existing custom object or create a new custom object now.
Object
Customer Details
Tab Style
(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.
Splash Page Custom Link
--None--
Enter a short description.
Description
Next
Cancel


Tab Style Selector
Create your own style
Hide styles which are used on other tabs
Airplane
Alarm clock
Apple
Balls
Bank[1]
Bell
Big top
Boat[1]
Books
Bottle
Box
Bridge
Building
Building Block
Caduceus
Camera
Can
Car
Castle
CD/DVD
Cell phone
Chalkboard
Chess piece
Chip
Circle
Compass
Computer
Credit card
CRT TV
Cup
Desk[1]
Diamond
Dice
Factory
Fan
Flag
Form
Gears
Globe
Guitar
Hammer
Hands
Handsaw
Headset
Heart[1]
Helicopter
Hexagon
Highway Sign
Hot Air Balloon
Insect
IP Phone
Jewel
Keys
Laptop
Leaf
Lightning
Save
Cancel


Step 3. Add to Custom Apps
Step 3 of 3
Choose the custom apps for which the new custom tab will be available. You may also examine or alter the visibility of tabs from the detail and edit pages of each Custom App.
Custom App
Include Tab
Platform (standard__Platform)
Sales (standard__Sales)
Service (standard__Service)
Marketing (standard__Marketing)
Sample Console (standard__ServiceConsole)
High Volume Customer Portal User
Authenticated Website User
App Launcher (standard__AppLauncher)
Analytics Studio (standard__Insights)
Sales Console (standard__LightningSalesConsole)
Service Console (standard__LightningService)
Sales (standard__LightningSales)
Lightning Usage App (standard__LightningInstrumentation)
Digital Experiences (standard__SalesforceCMS)
Queue Management (standard__QueueManagement)
Bolt Solutions (standard__LightningBolt)
Data Manager (standard__DataManager)
Salesforce Scheduler Setup (standard__LightningScheduler)
Append tab to users' existing personal customizations
Previous
Save
Cancel

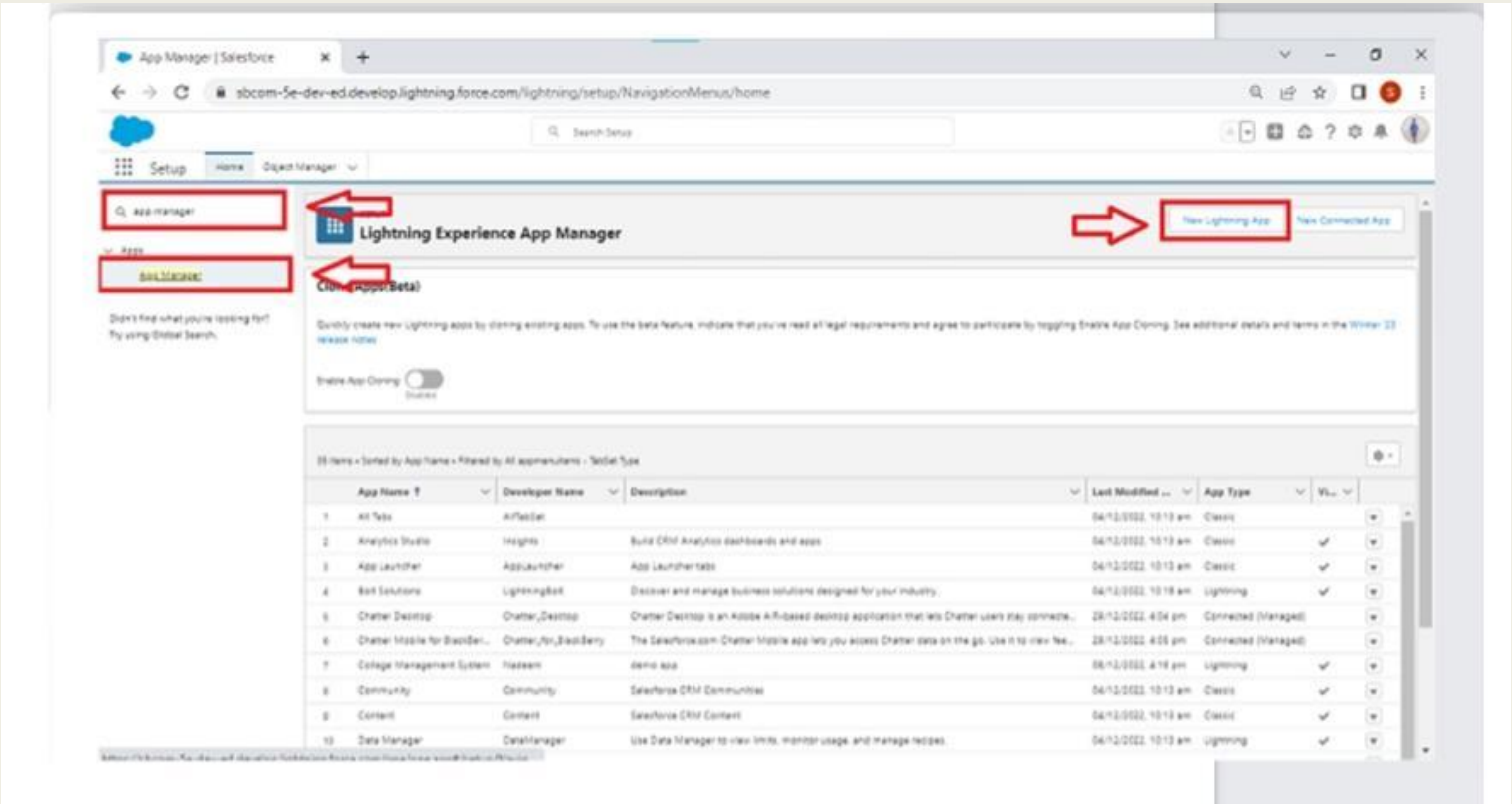
THE LIGHTNING APP

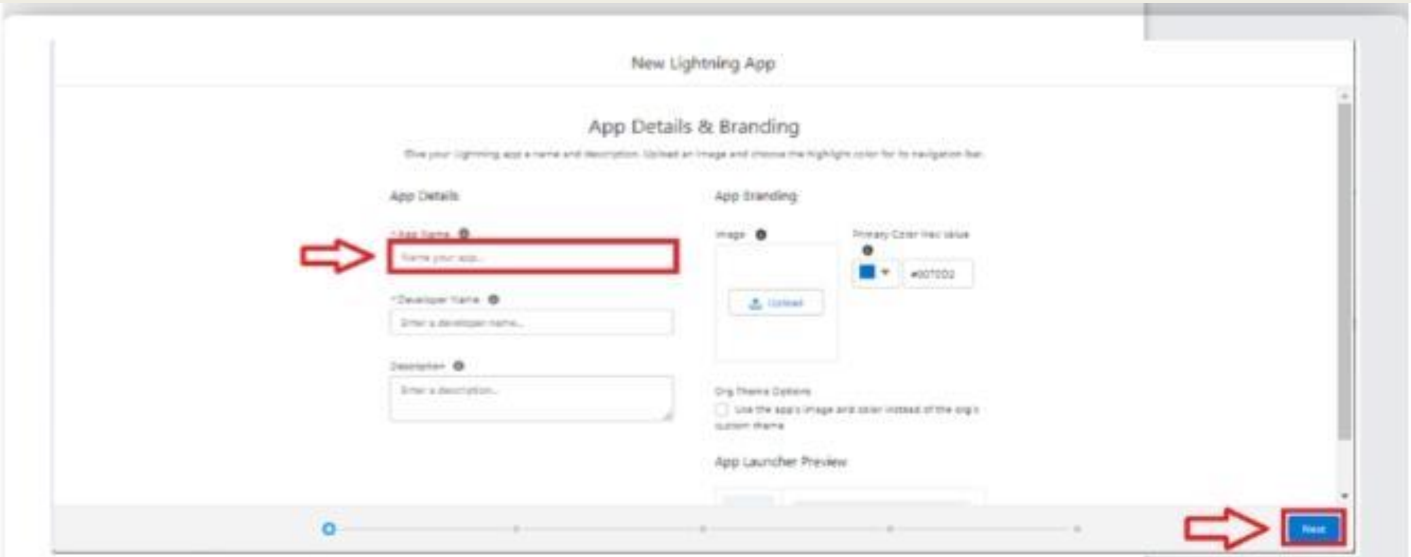
***An app is a collection of items that work together to serve a particular function.**

***In Lightning Experience, Lightning apps give your users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar.**

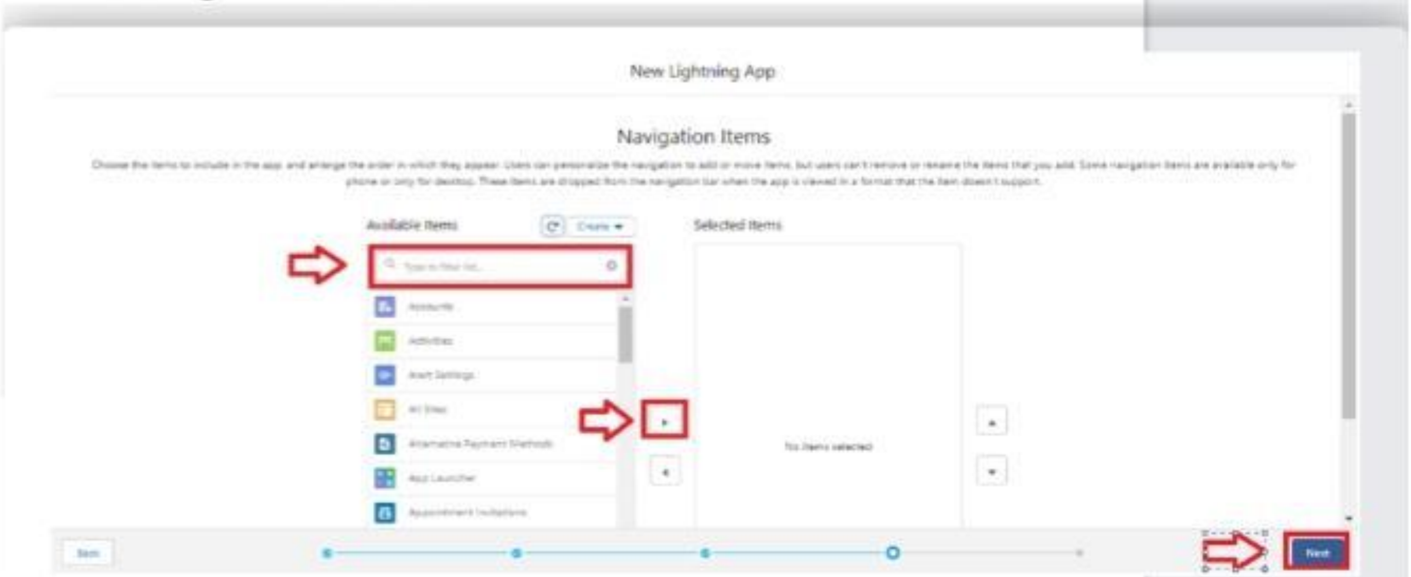
***Lightning apps let you brand your apps with a custom colour and logo. You can even include a utility bar and Lightning page tabs in your Lightning app.**

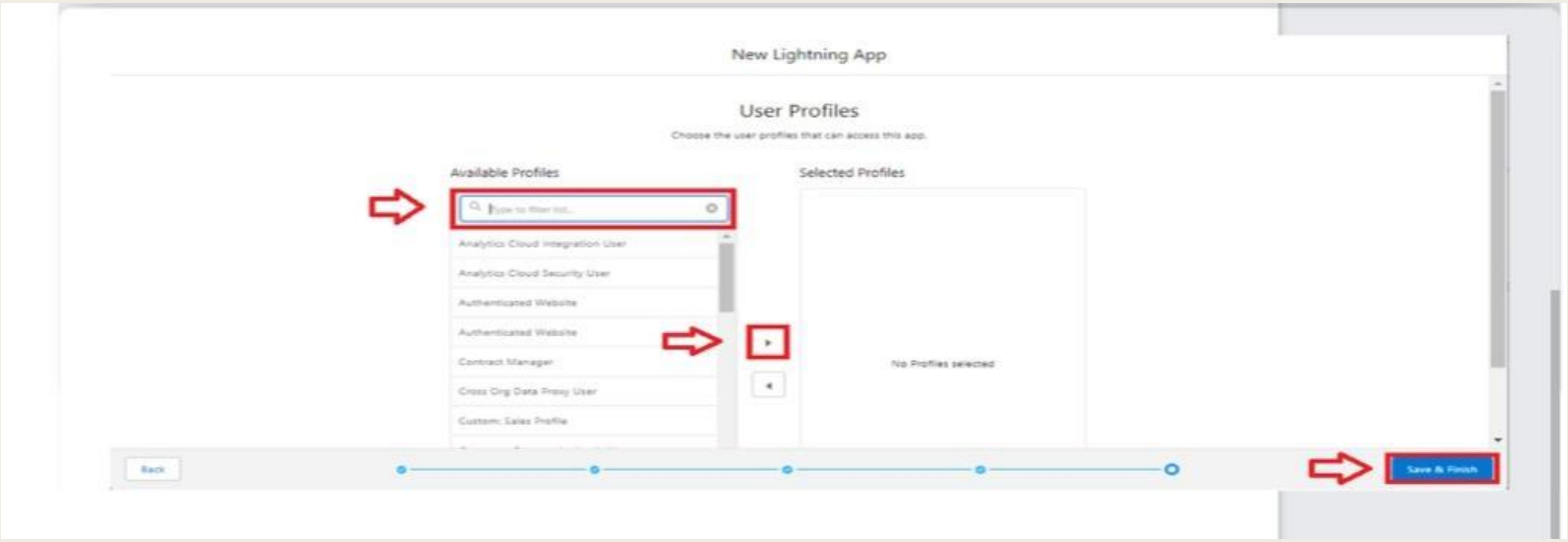
CREATE A LIGHTNING APP





3. To Add Navigation Items:




New Lightning App
User Profiles
Choose the user profiles that can access this app.
Available Profiles
Selected Profiles
Analytics Cloud Integration User
Analytics Cloud Security User
Authenticated Website
Authenticated Website
Contract Manager
Cross Org Data Proxy User
Custom: Sales Profile
No Profiles selected
Back
Save & Finish

FIELDS

- When we talk about Salesforce, Fields represent the data stored in the columns of a relational database.
- It can also hold any valuable information that you require for a specific object. Hence, the overall searching, deletion, and editing of the records become simpler and quicker.

Types of Fields

1. Standard Fields
2. Custom Fields

STANDARD FIELDS:

As the name suggests, the Standard Fields are the predefined fields in Salesforce that perform a standard task.

- The main point is that you can't simply delete a Standard Field until it is a non-required standard field.
- Moreover, we have some fields that you will find common in every Salesforce application.

  They are,

  * Created By
  * Owner
  * Last Modified
  * Field Made During object Creation

CUSTOM FIELDS:

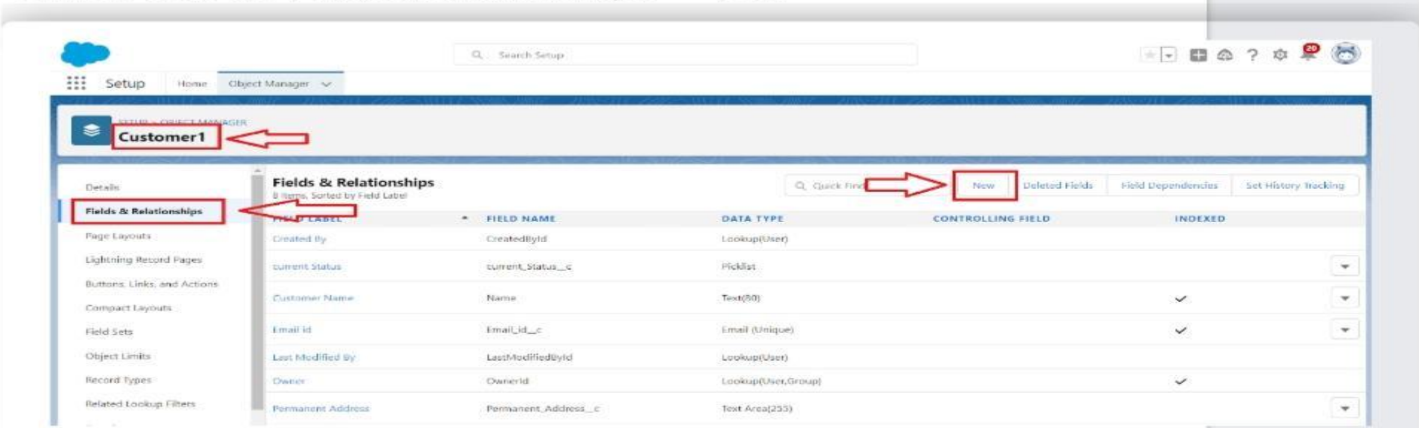
- **On the other side of the coin, Custom Fields are highly flexible, and users can change them according to requirements.**
- **Hence, the final decision depends on the user, and he can add/remove Custom Fields of any given form.**

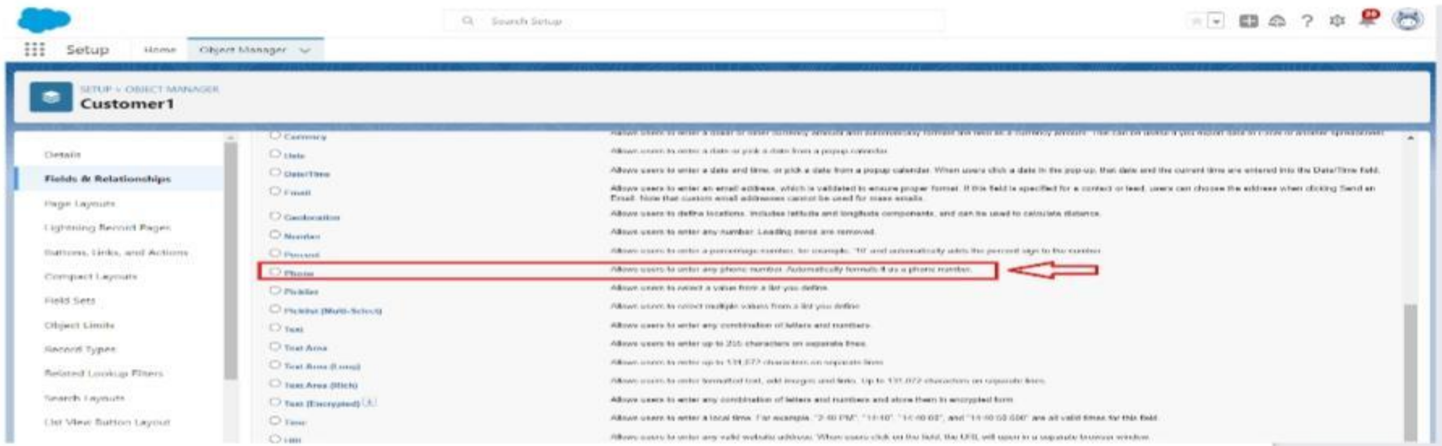
CREATION OF FIELDS FOR THE CUSTOMER DETAILS OBJECT

1. To create fields in an object:

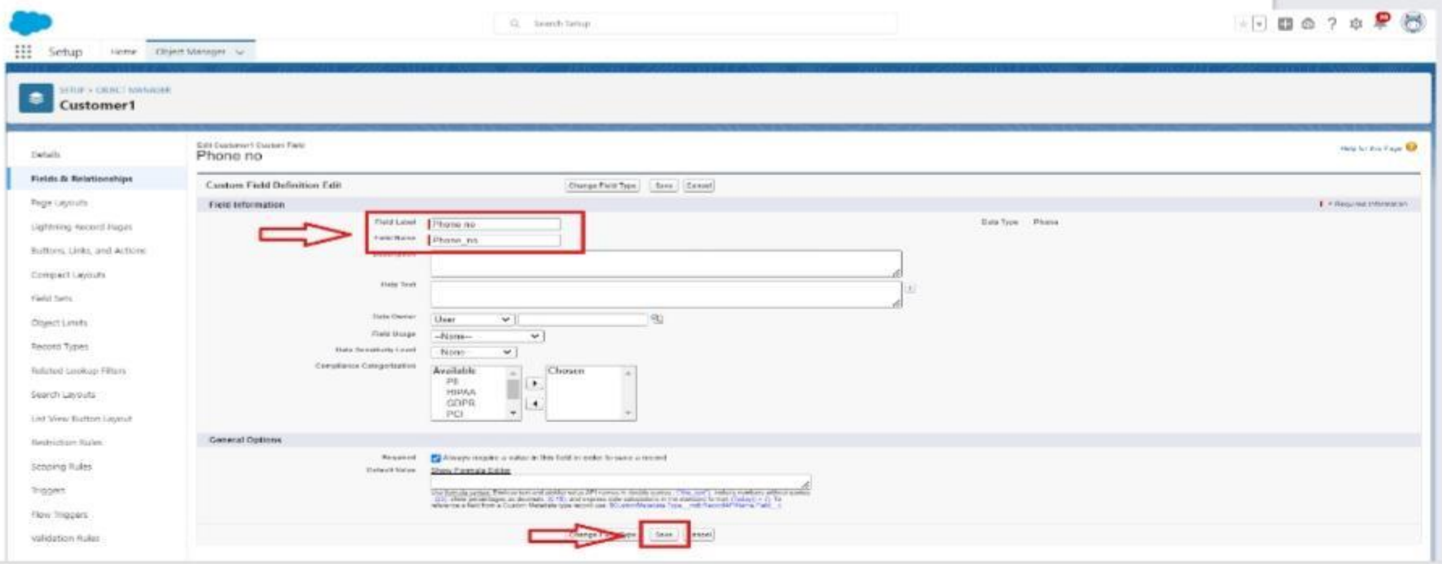
1. Go to setup >> click on Object Manager >> type object name(Customer Details) in search bar >> click on the object.

2. Now click on “Fields & Relationships” >> New

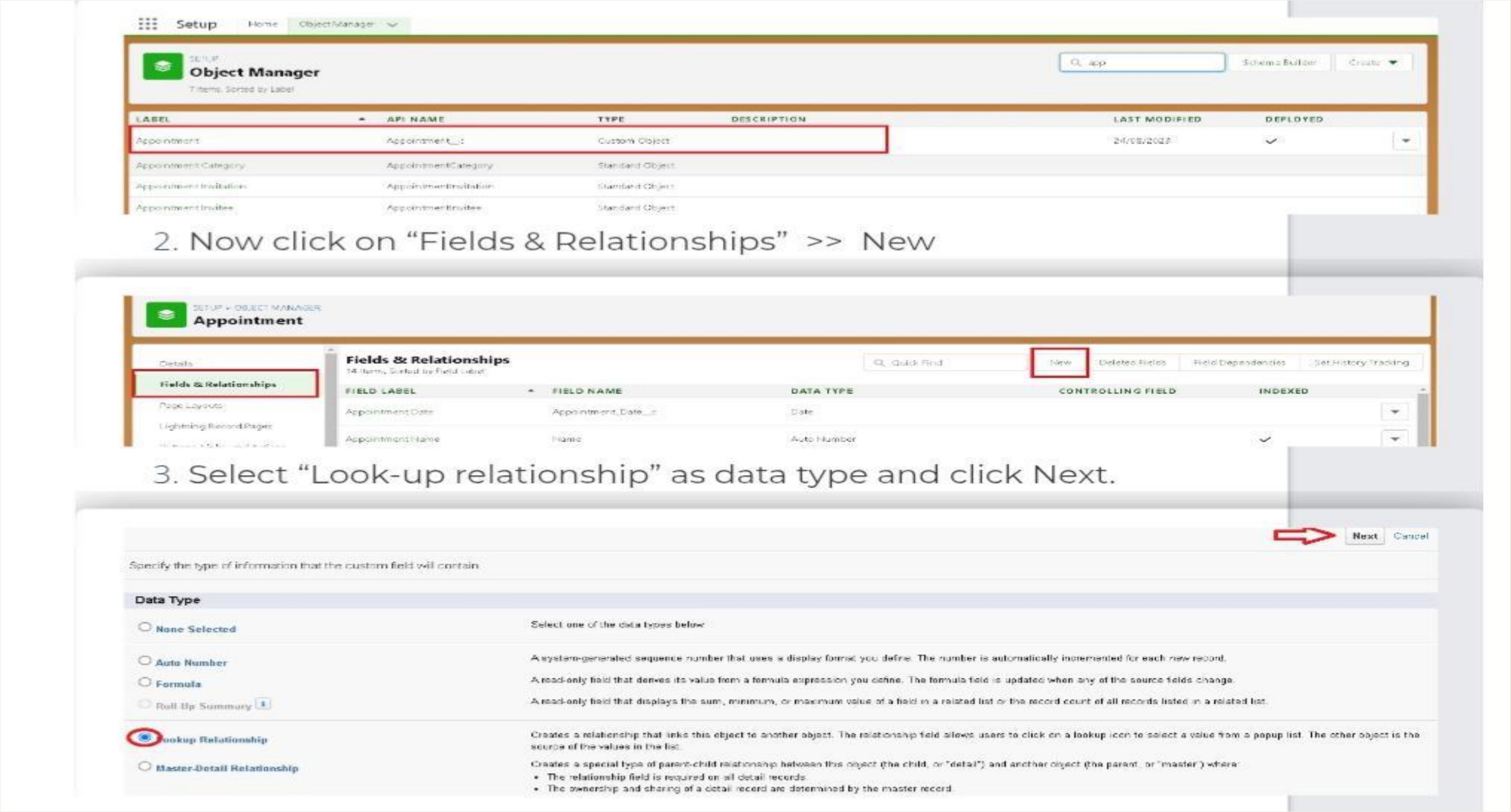


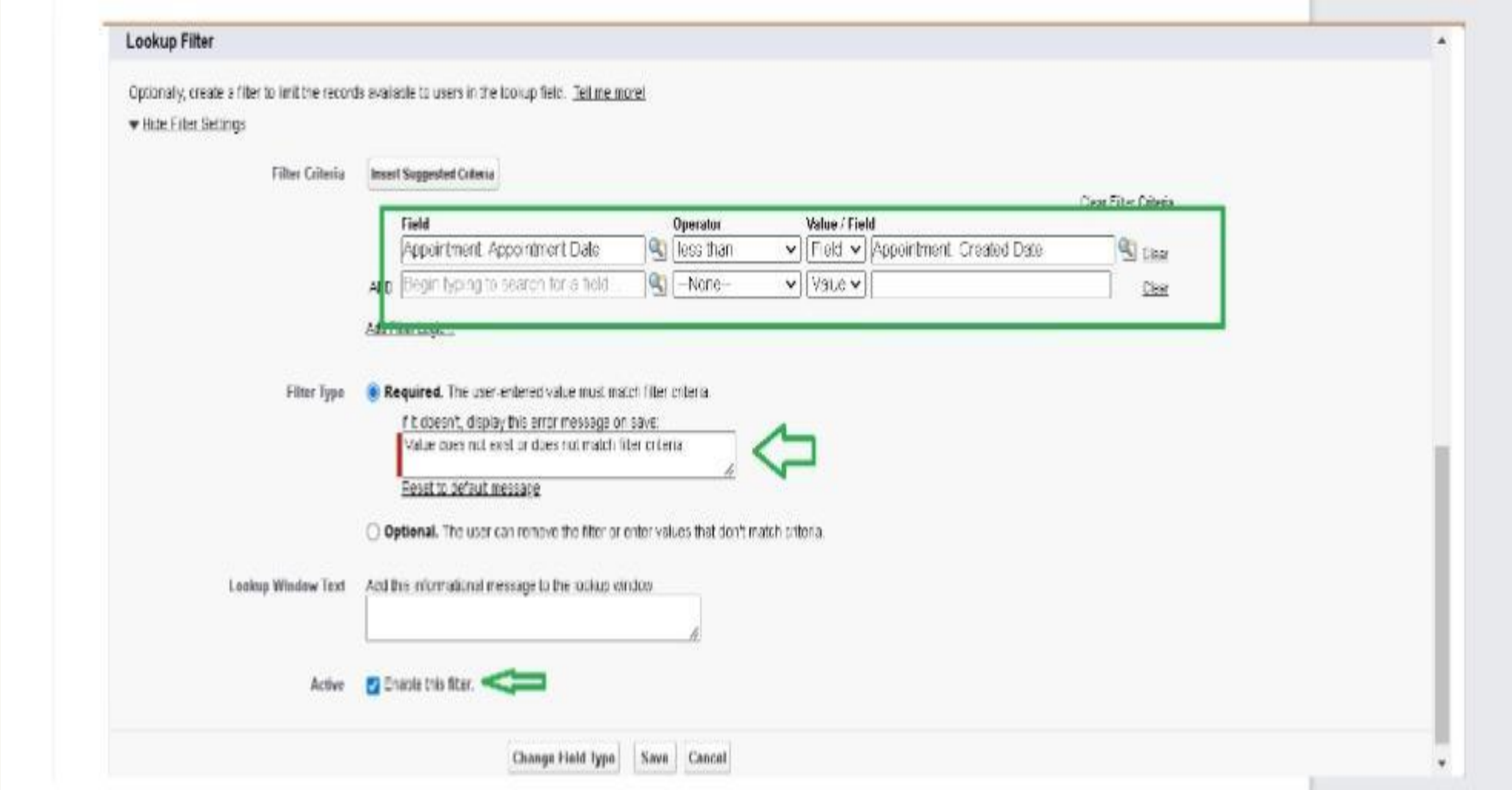


4. Click on next.



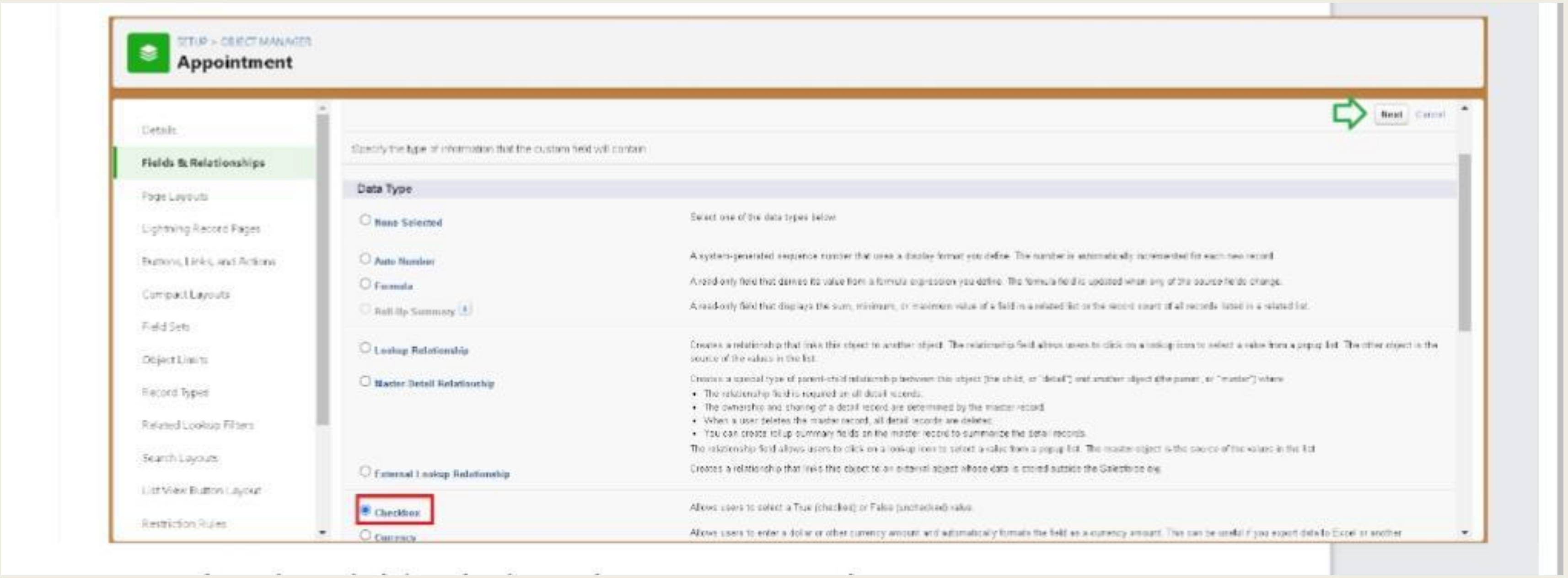
CREATION OF LOOKUP FIELDS

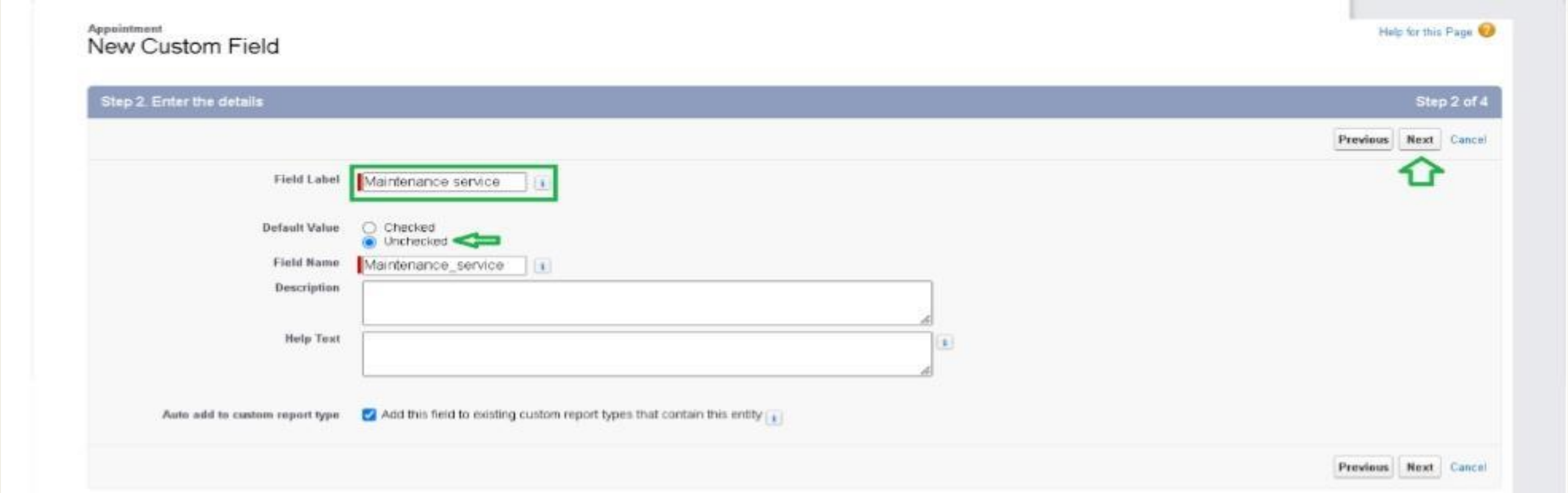




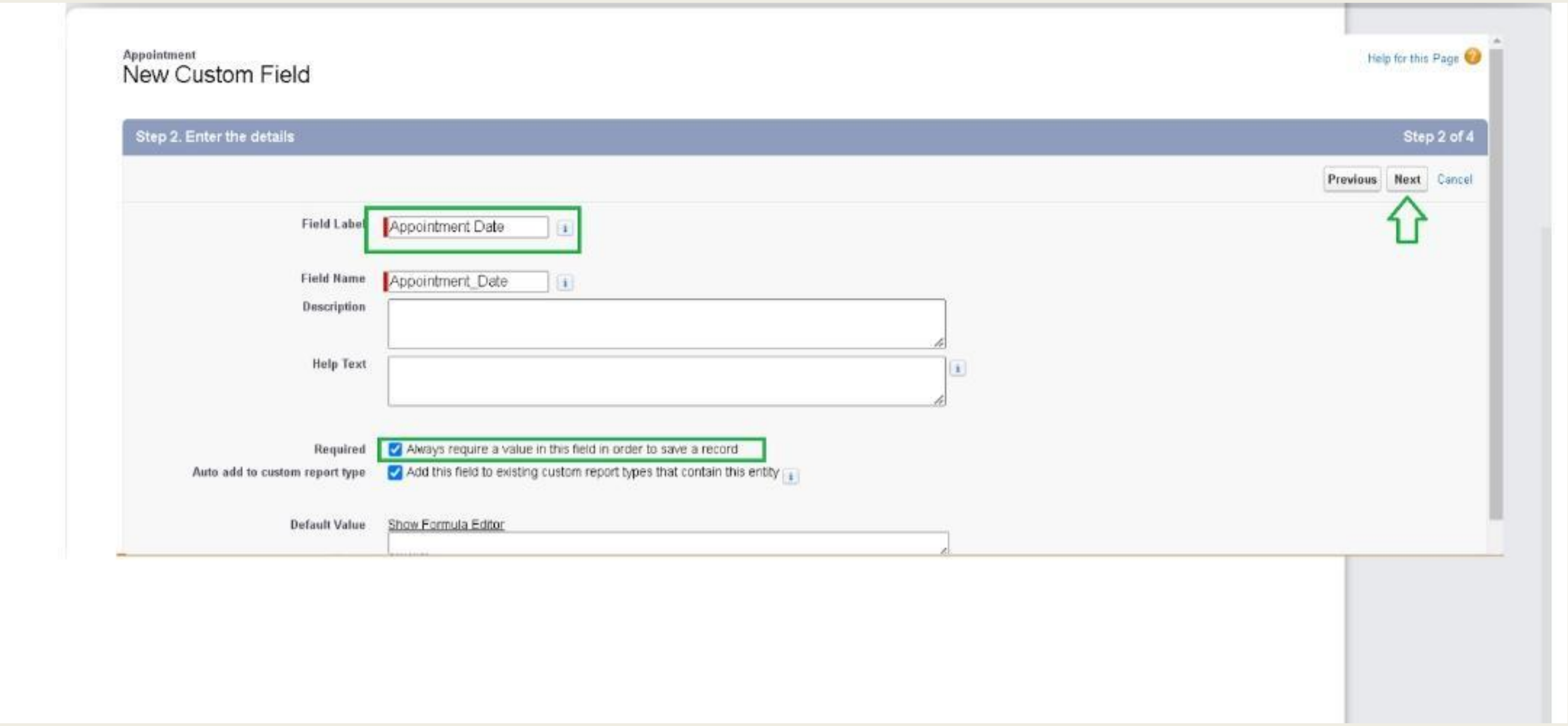
Lookup Filter
Optionally, create a filter to limit the records available to users in the lookup field. Tell me more!
Hide Filter Settings
Filter Criteria
Insert Suggested Criteria
Field
Operator
Value / Field
Appointment: Appointment Date
less than
Field
Appointment: Created Date
Clear
AND
Begin typing to search for a field...
--None--
Value
Clear
Filter Type
Required. The user-entered value must match filter criteria.
If it doesn't, display this error message on save:
Value does not exist or does not match filter criteria.
Reset to default message
Optional. The user can remove the filter or enter values that don't match criteria.
Lookup Window Text
Add this informational message to the lookup window.
Active
Enable this filter.
Change Field Type
Save
Cancel

CREATION OF CHECK BOX FIELDS

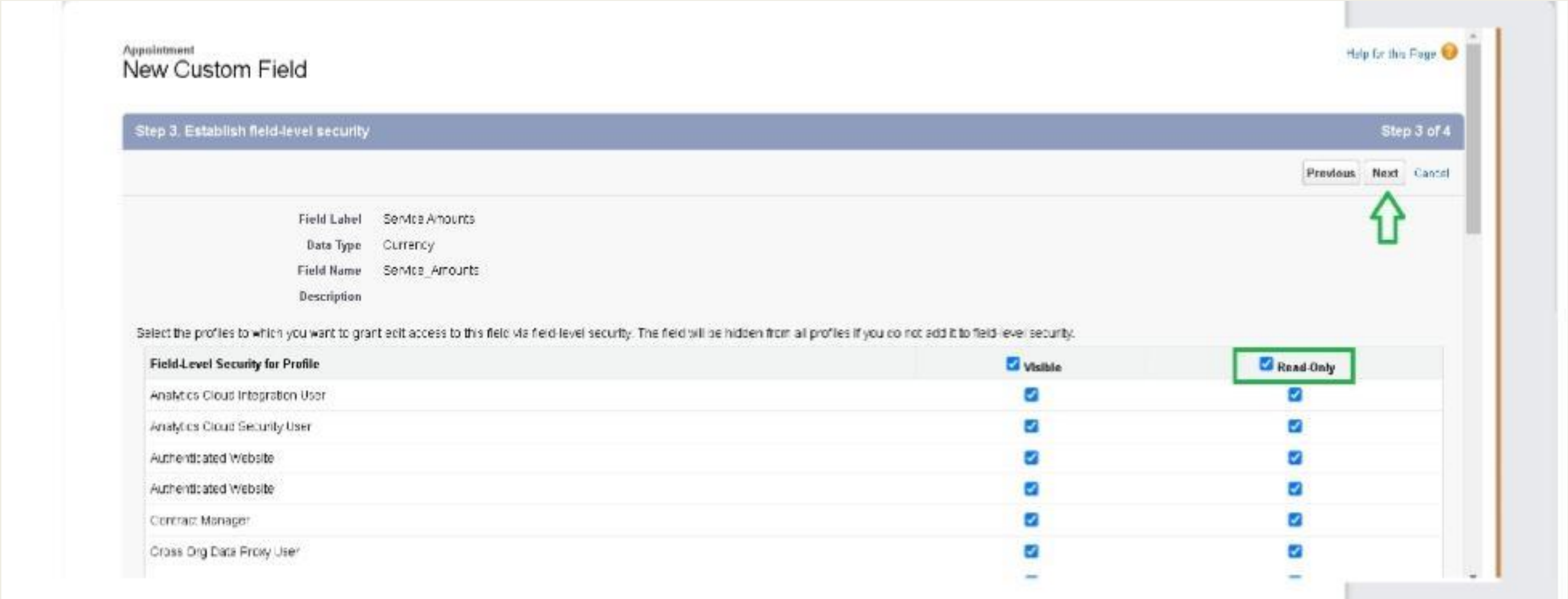




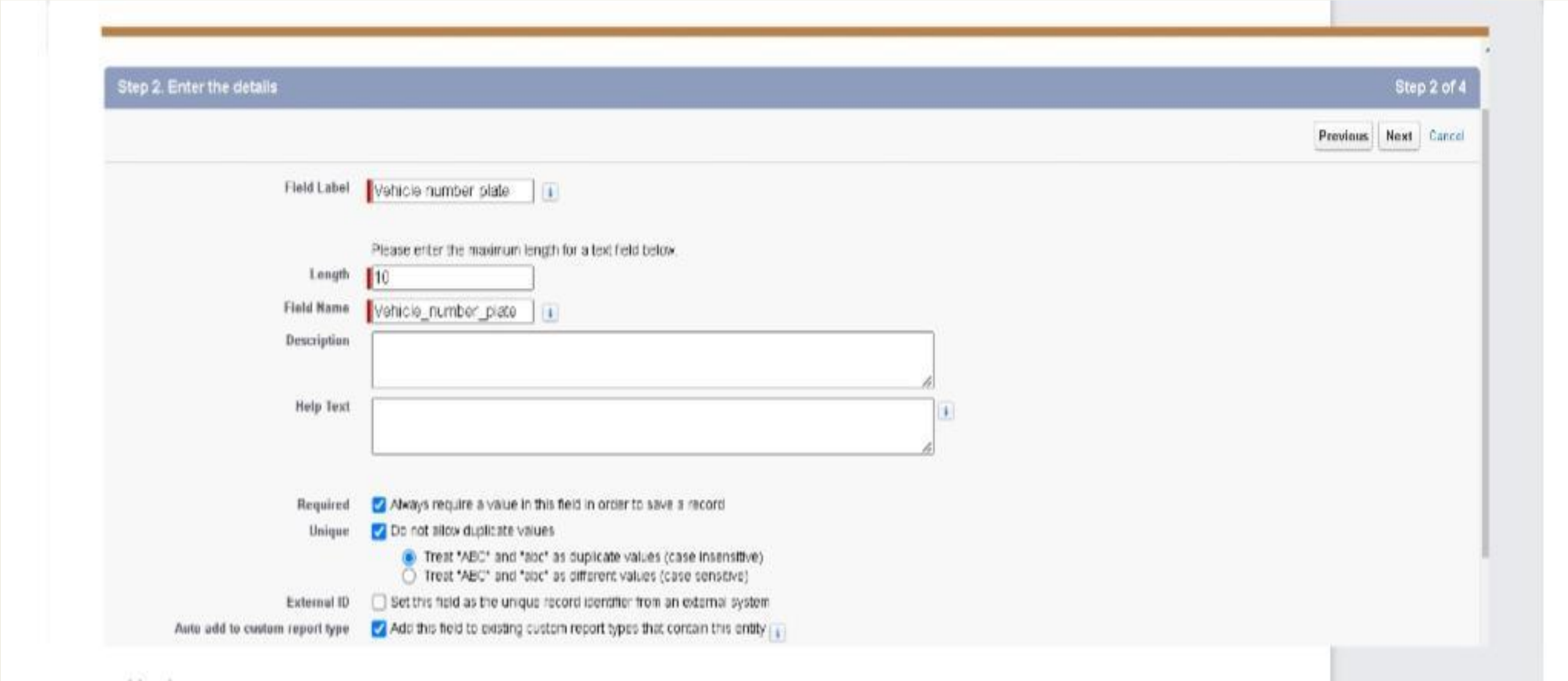
CREATION OF DATE FIELDS



CREATION OF CURRENCY FIELDS

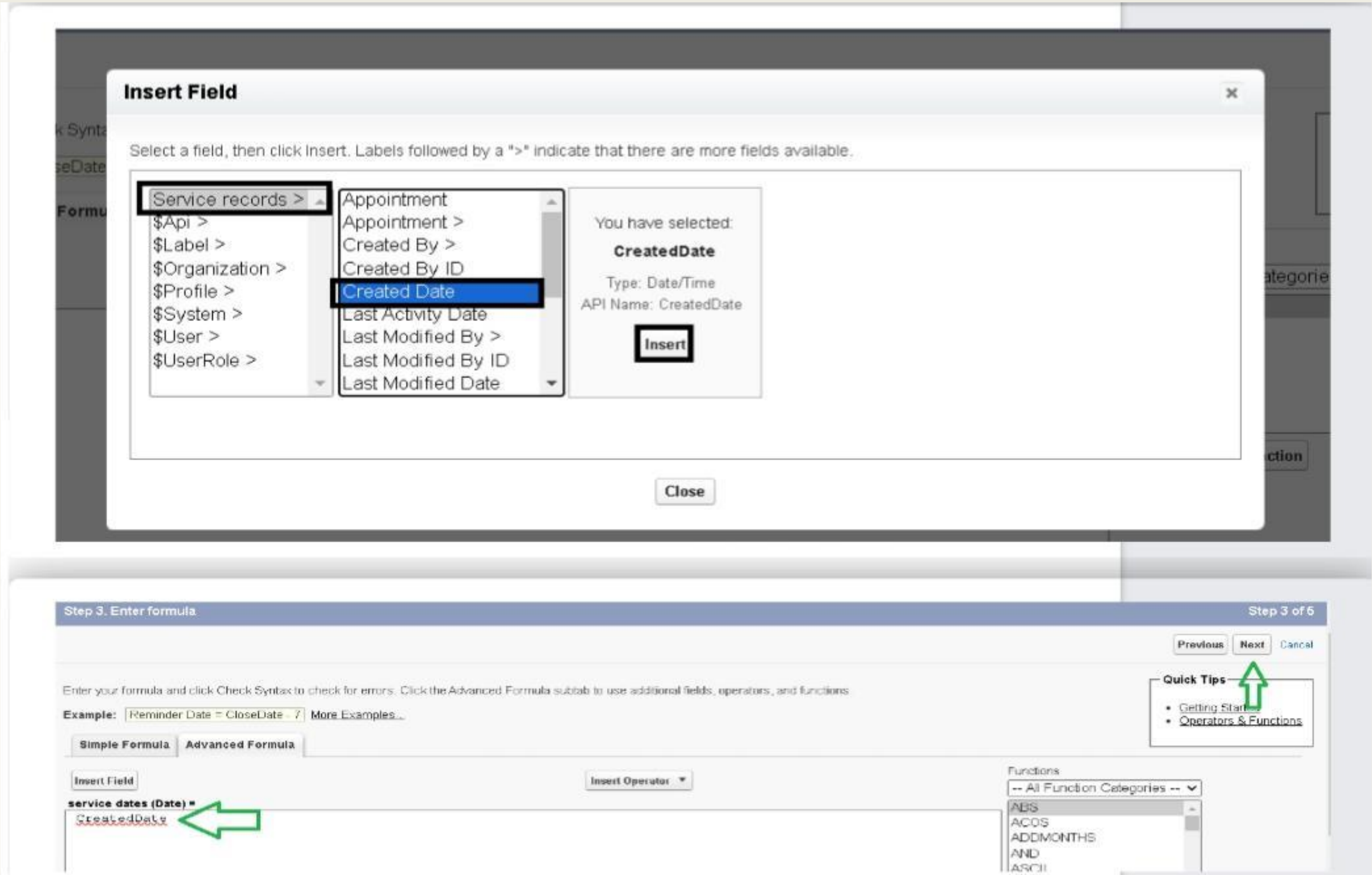


CREATION OF TEXT FIELDS



CREATION OF PICKLIST FIELDS

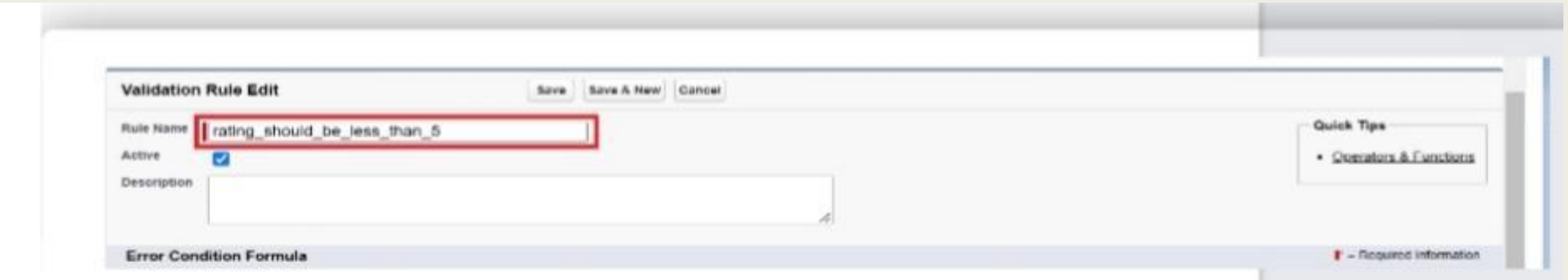
CREATING FORMULA FIELD IN SERVICE RECORDS OBJECT

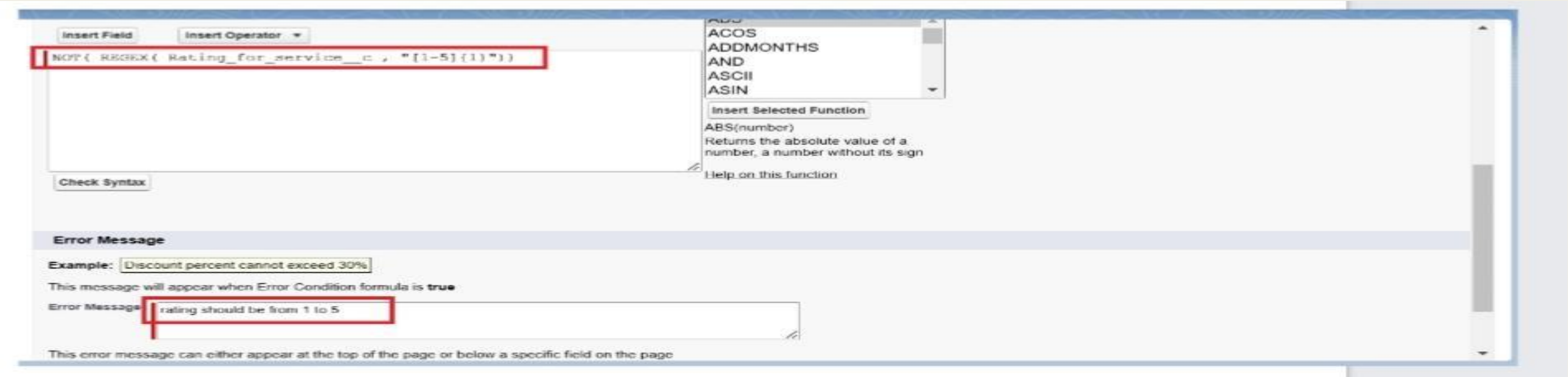

Insert Field
Select a field, then click Insert. Labels followed by a ">" indicate that there are more fields available.
Service records >
$Api >
$Label >
$Organization >
$Profile >
$System >
$User >
$UserRole >
Appointment
Appointment >
Created By >
Created By ID
Created Date
Last Activity Date
Last Modified By >
Last Modified By ID
Last Modified Date
You have selected:
CreatedDate
Type: Date/Time
API Name: CreatedDate
Insert
Close
Step 3. Enter formula
Step 3 of 6
Previous
Next
Cancel
Quick Tips
Operators & Functions
Example:
More Examples...
Simple Formula
Advanced Formula
Insert Field
Insert Operator
Functions
-- All Function Categories --
service dates (Date) =
CreatedDate
ABS
ACOS
ADDMONTHS
AND

VALIDATION RULE

Validation rules are applied when a user tries to save a record and are used to check if the data meets specified criteria.

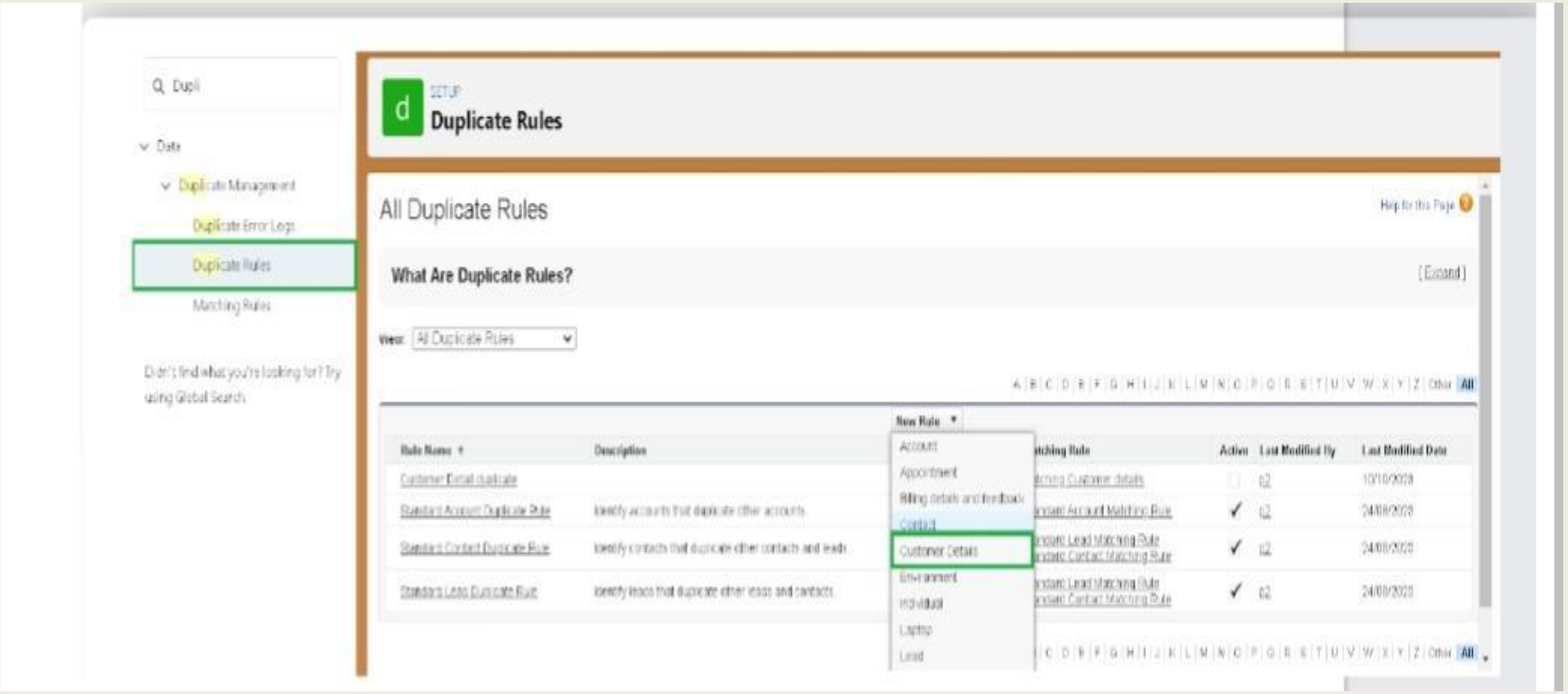
- To create a validation rule to an Billing details and feedback Object

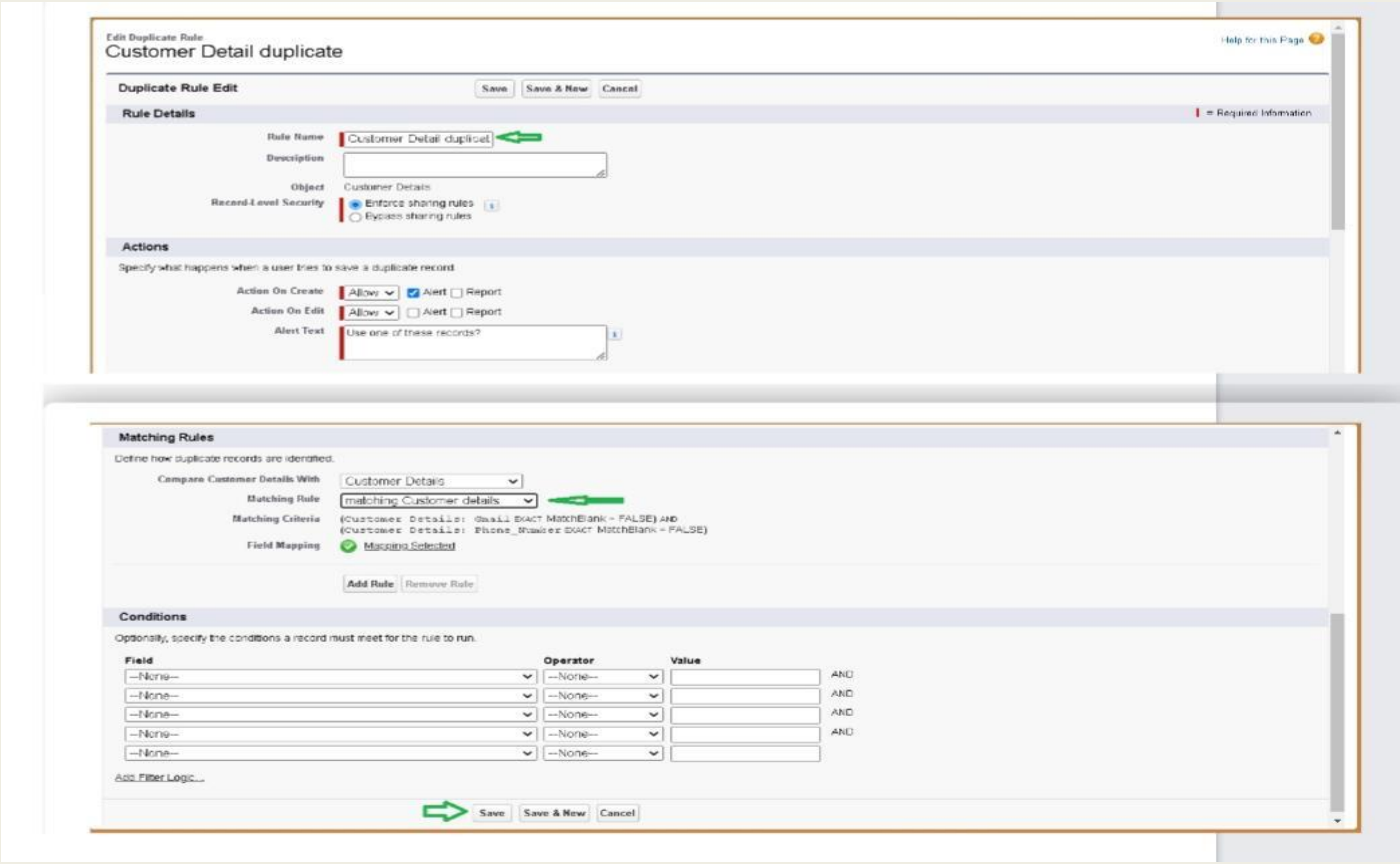




DUPLICATE RULE

TO CREATE A DUPLICATE RULE TO AN CUSTOMER DETAILS OBJECT




Edit Duplicate Rule
Customer Detail duplicate
Help for this Page
Duplicate Rule Edit
Save
Save & New
Cancel
Rule Details
= Required Information
Rule Name
Customer Detail duplicat
Description
Object
Customer Details
Record-Level Security
Enforce sharing rules
Bypass sharing rules
Actions
Specify what happens when a user tries to save a duplicate record
Action On Create
Allow
Alert
Report
Action On Edit
Allow
Alert
Report
Alert Text
Use one of these records?
Matching Rules
Define how duplicate records are identified.
Compare Customer Details With
Customer Details
Matching Rule
matching Customer details
Matching Criteria
(Customer Details: Gmail EXACT MatchBlank = FALSE) AND
(Customer Details: Phone_Number EXACT MatchBlank = FALSE)
Field Mapping
Mapping Selected
Add Rule
Remove Rule
Conditions
Optionally, specify the conditions a record must meet for the rule to run.
Field
Operator
Value
--None--
AND
Add Filter Logic...
Save
Save & New
Cancel

PROFILES

A profile is a group/collection of settings and permissions that define what a user can do in salesforce.

Types of profiles in salesforce

Standard profiles:

*By default salesforce provides below standard profiles.

*Contract Manager

*Read Only

*Marketing User

*Solutions Manager

*Standard User

*System Administrator.

*We cannot deleted standard ones

Custom Profiles:

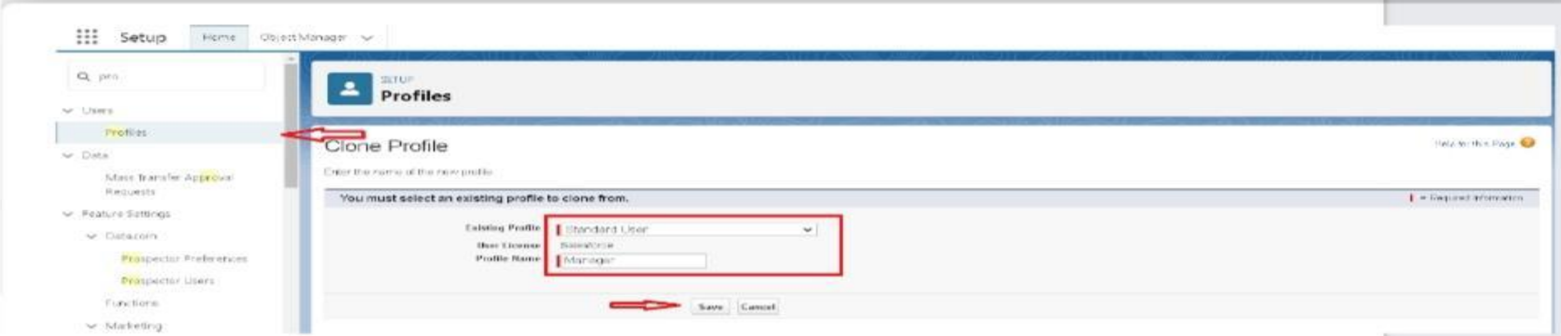
*Custom ones defined by us.

*They can be deleted if there are no users assigned with that particular one.

MANAGER PROFILE

To create a new profile:

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Manager) >> Save.



2. While still on the profile page, then click Edit.

3. Select the Custom App settings as default for the Garage management.

4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.

5. Changing the session times out after should be " 8 hours of inactivity".
6. Change the password policies as mentioned :
7. User passwords expire in should be " never expires ".
8. Minimum password length should be " 8 ", and click save.

SALES PERSON PROFILE

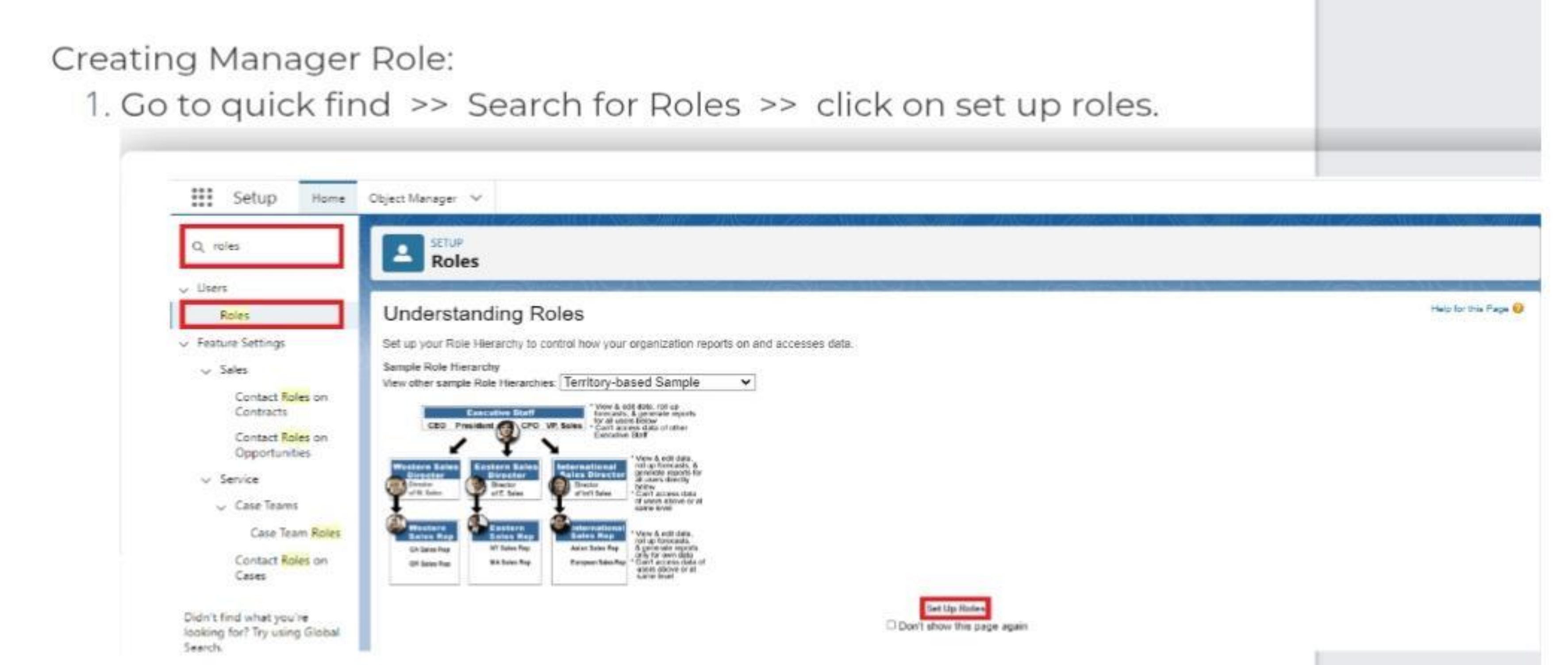
1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Salesforce Platform User) >> enter profile name (sales person) >> Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the GArage management.
4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.

5. And click save.

ROLE & ROLE HIERARCHY

A role in Salesforce defines a user's visibility access at the record level.

Creating Manager Role



2. Click on Expand All and click on add role under whom this role works.

3. Give Label as "Manager" and Role name gets auto populated. Then click on Save.

CREATING ANOTHER ROLES

Creating another two roles under manager

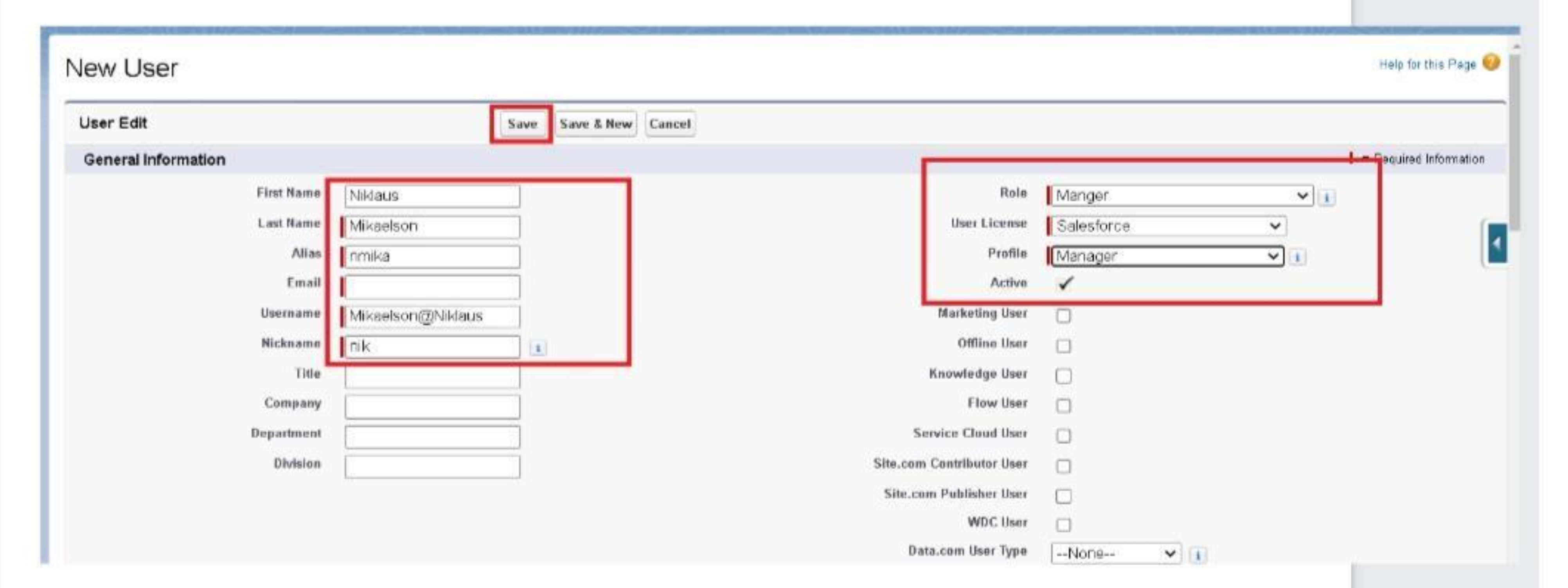
1. Go to quick find >> Search for Roles >> click on set up roles.
2. Click plus on CEO role, and click add role under manager.

3. Give Label as “sales person” and Role name gets auto populated. Then click on Save.

USERS

A user is anyone who logs in to Salesforce. Users are employees at your company, such as sales reps, managers, and IT specialists, who need access to the company's records.

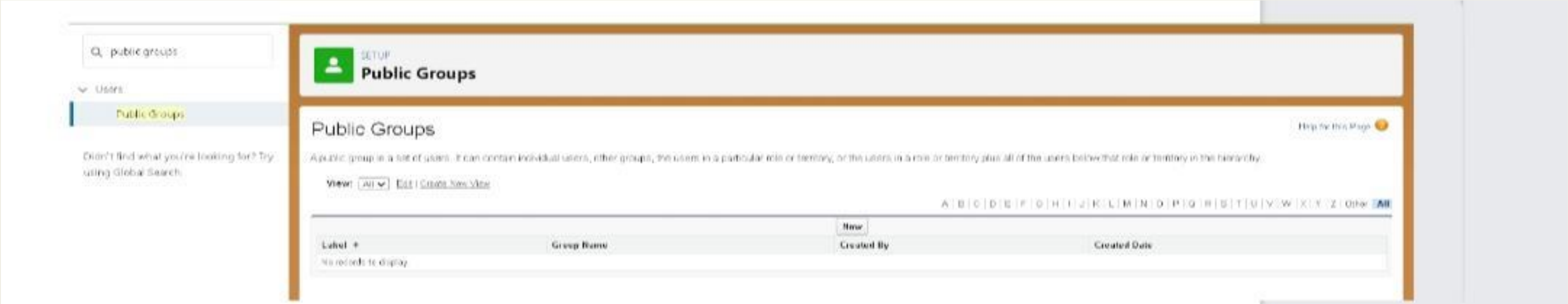
CREATE USER

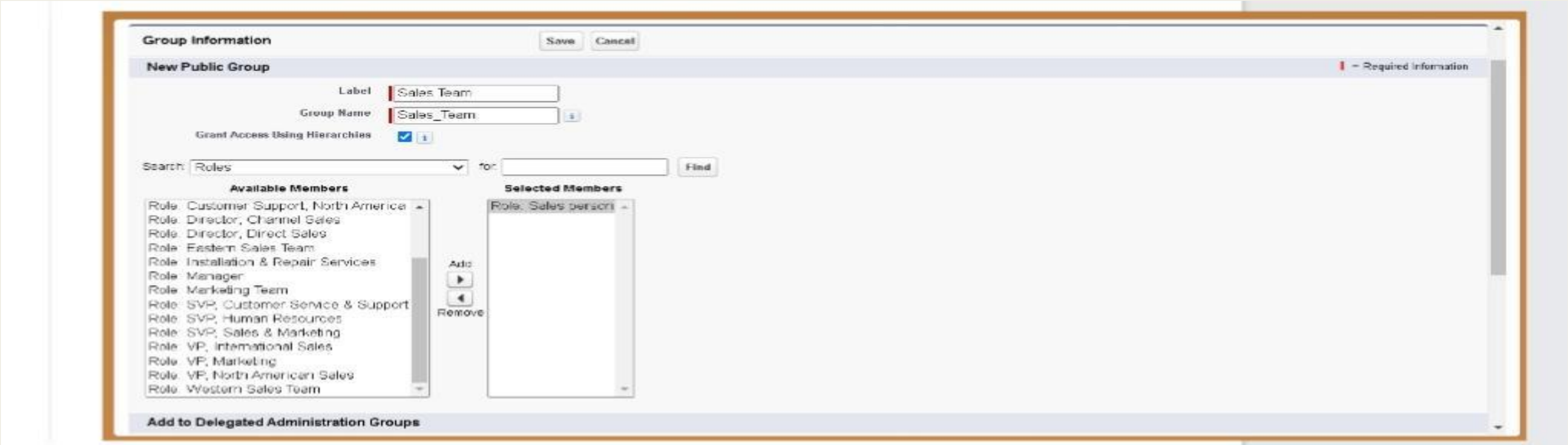


PUBLIC GROUPS

- **Public groups are a valuable tool for Salesforce administrators and developers to streamline user management, data access, and security settings.**

CREATING NEW PUBLIC GROUP

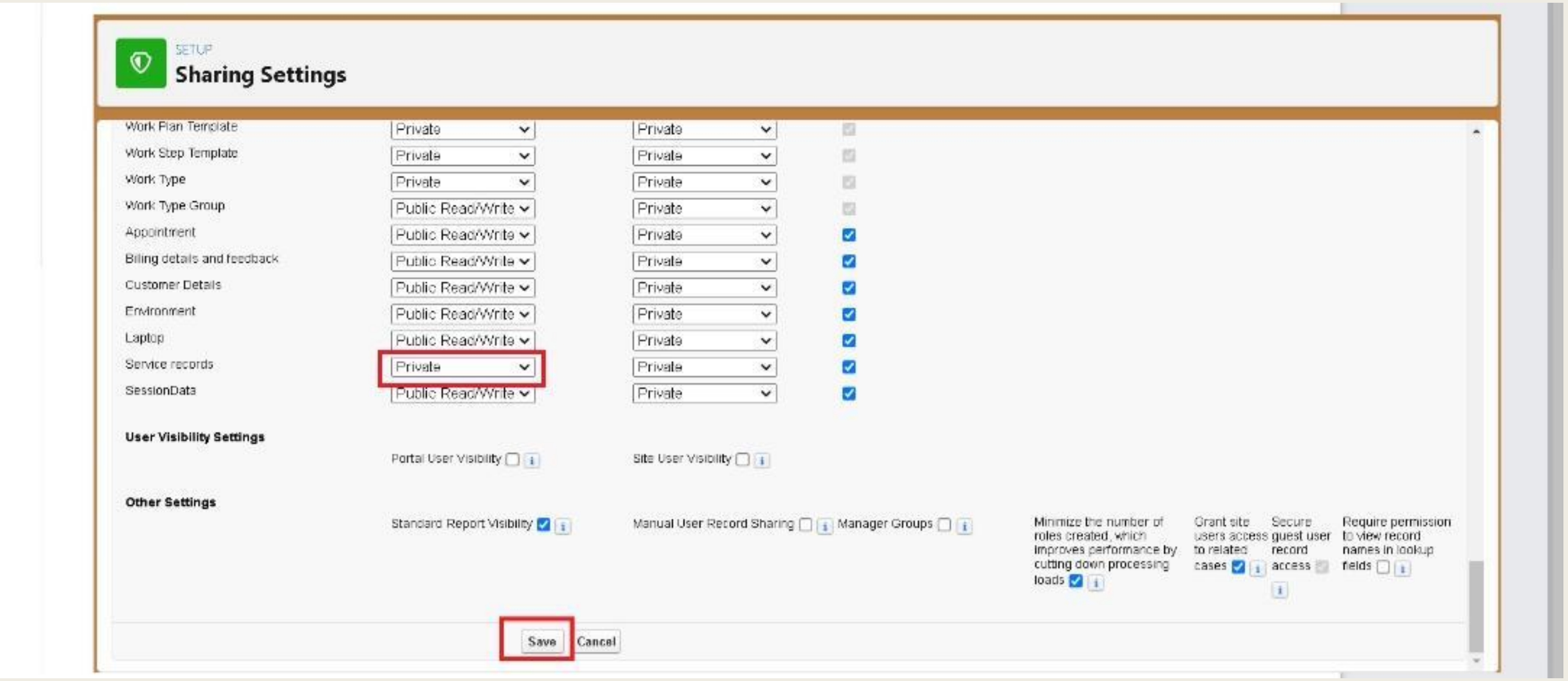


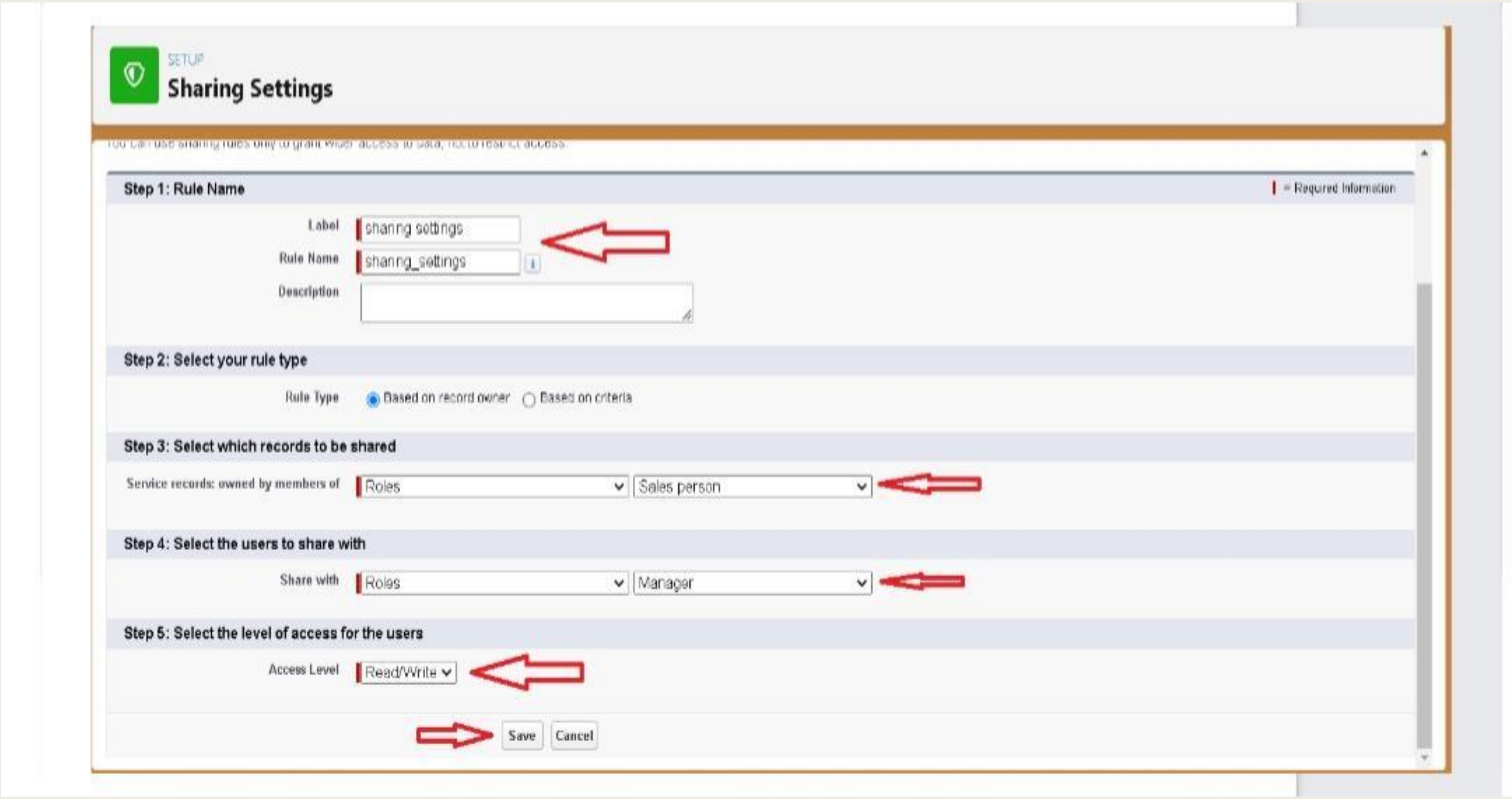


SHARING SETTING

Salesforce allows you to configure sharing settings to control how records are accessed and shared within your organization.

CREATING SHARING SETTINGS

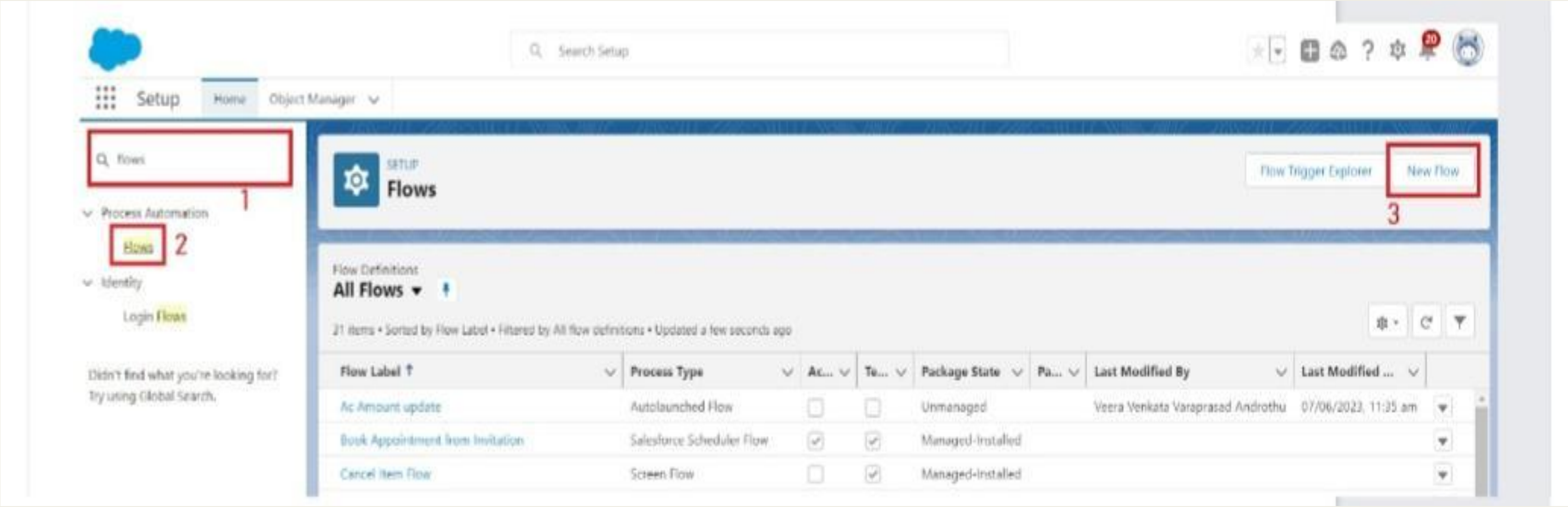


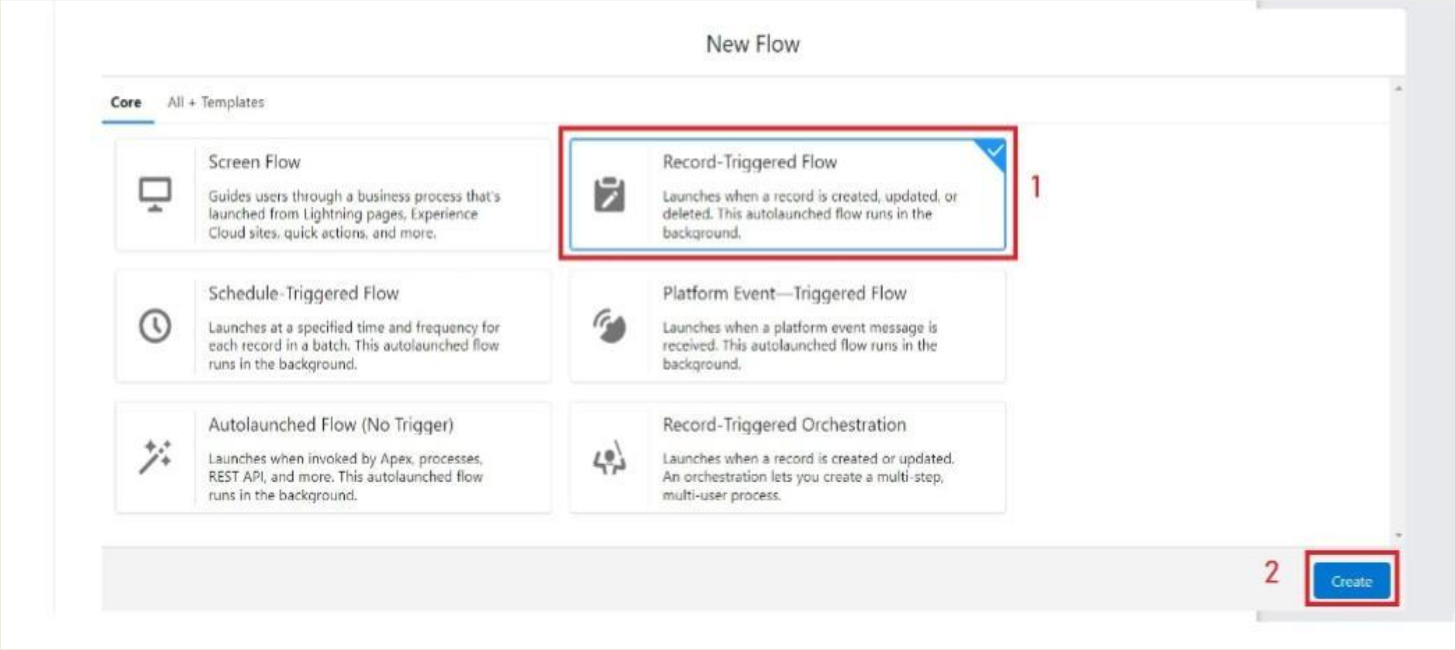

SETUP
Sharing Settings
Step 1: Rule Name
= Required Information
Label
sharing settings
Rule Name
sharing_settings
Description
Step 2: Select your rule type
Rule Type
Based on record owner
Based on criteria
Step 3: Select which records to be shared
Service records: owned by members of
Roles
Sales person
Step 4: Select the users to share with
Share with
Roles
Manager
Step 5: Select the level of access for the users
Access Level
Read/Write
Save
Cancel

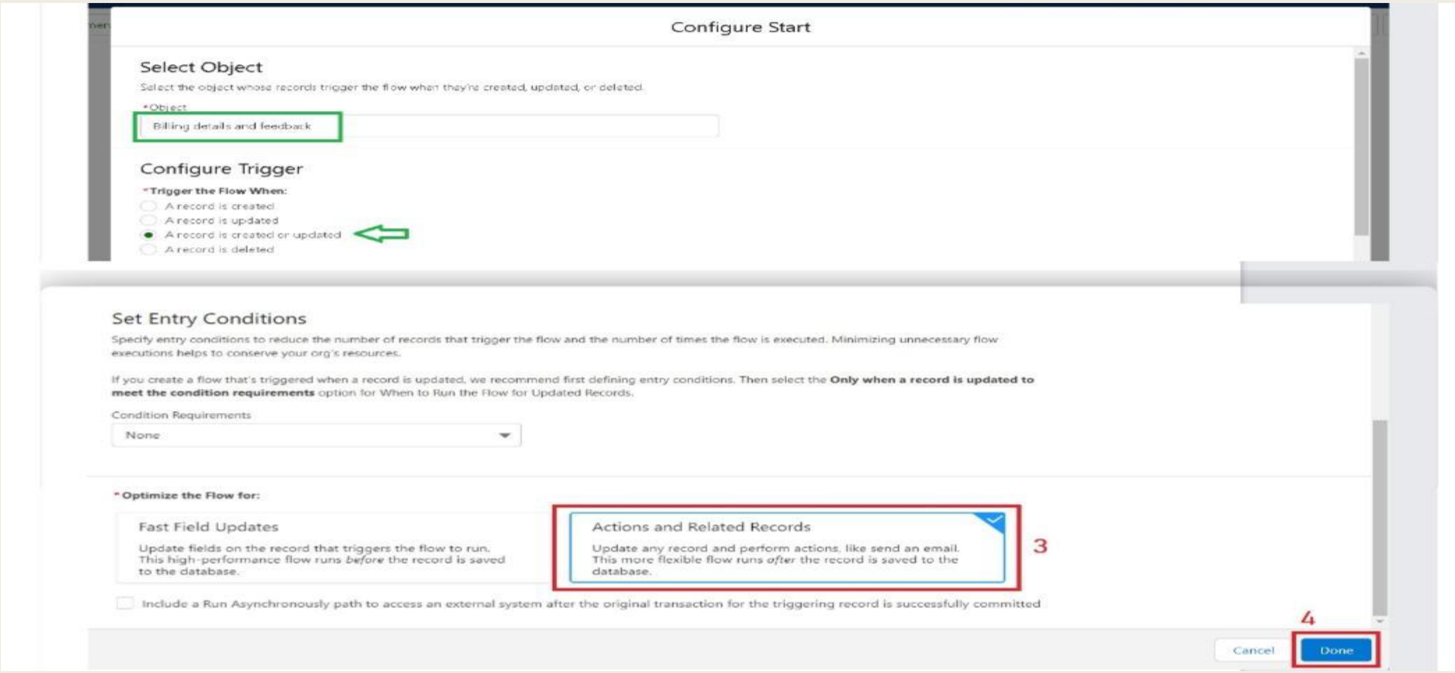
FLOWS

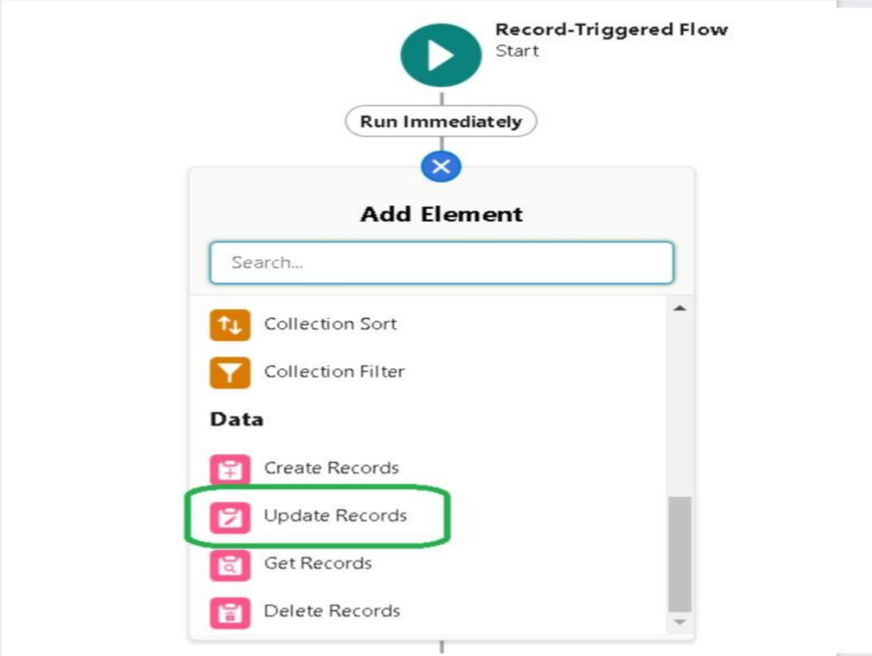
In Salesforce, a flow is a powerful tool that allows you to automate business processes, collect and update data, and guide users through a series of screens or steps.

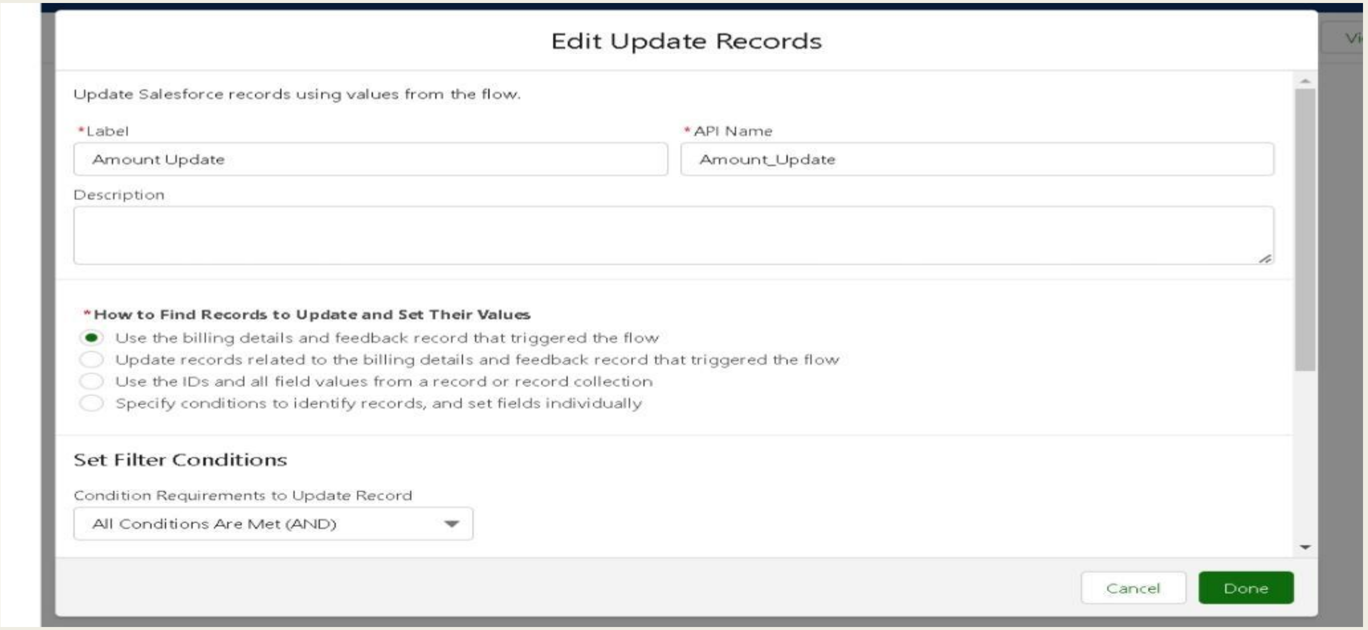
CREATE A FLOW

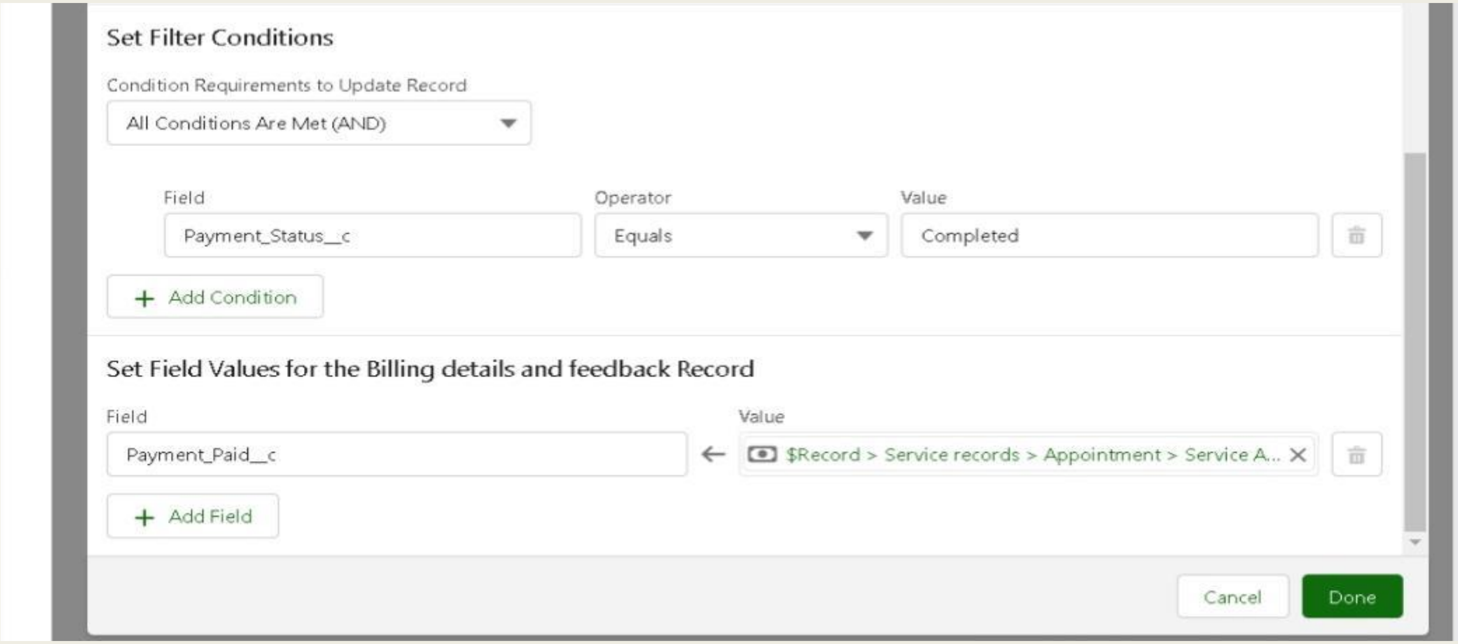


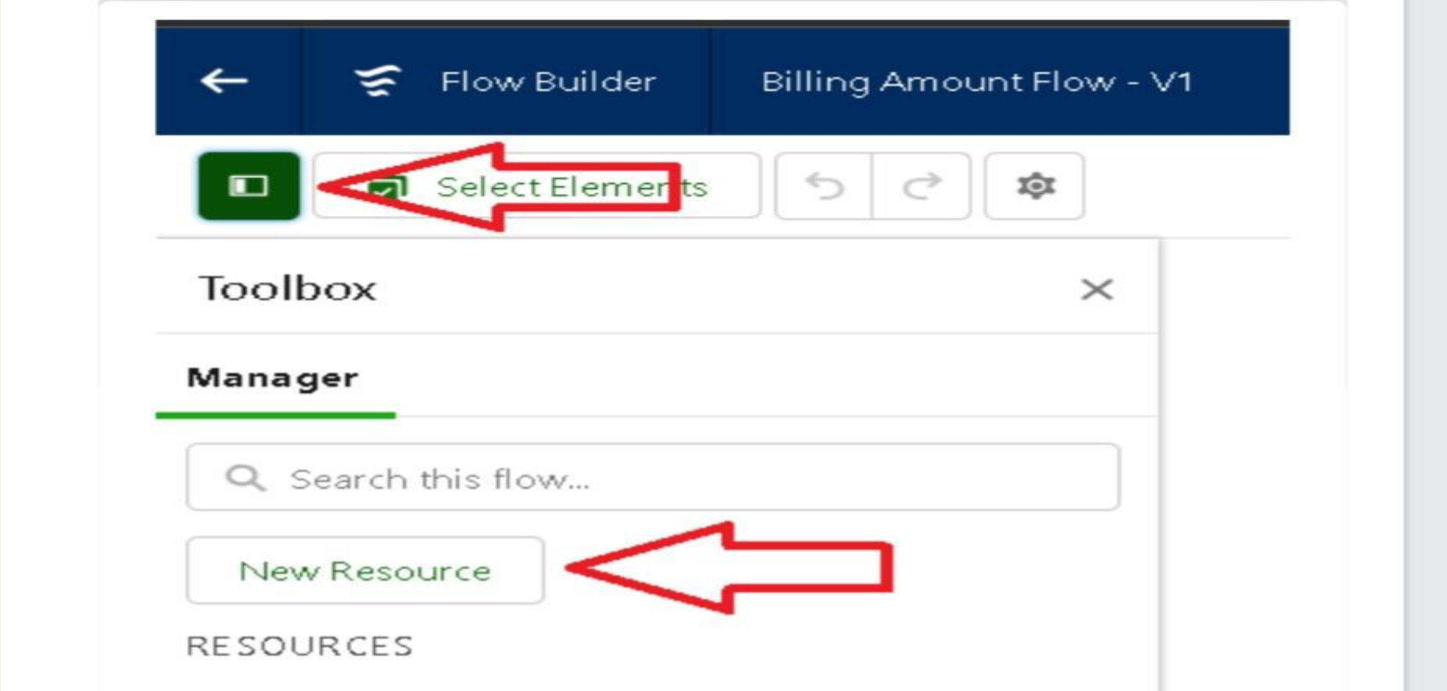

New Flow
Core
All + Templates
Screen Flow
Guides users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, and more.
Record-Triggered Flow
Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.
1
Schedule-Triggered Flow
Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the background.
Platform Event—Triggered Flow
Launches when a platform event message is received. This autolaunched flow runs in the background.
Autolaunched Flow (No Trigger)
Launches when invoked by Apex, processes, REST API, and more. This autolaunched flow runs in the background.
Record-Triggered Orchestration
Launches when a record is created or updated. An orchestration lets you create a multi-step, multi-user process.
2
Create

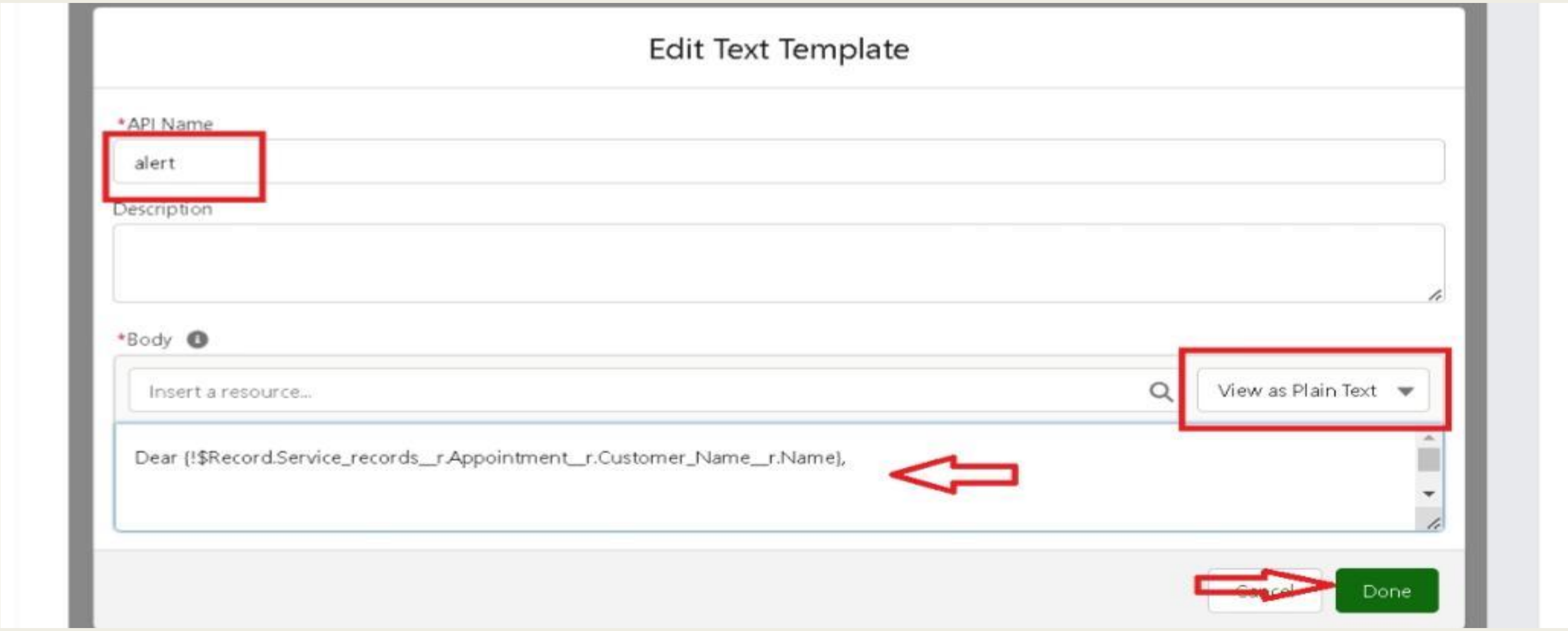

Configure Start
Select Object
Select the object whose records trigger the flow when they're created, updated, or deleted.
*Object
Billing details and feedback
Configure Trigger
*Trigger the Flow When:
A record is created
A record is updated
A record is created or updated
A record is deleted
Set Entry Conditions
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.
If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the Only when a record is updated to meet the condition requirements option for When to Run the Flow for Updated Records.
Condition Requirements
None
*Optimize the Flow for:
Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs before the record is saved to the database.
Actions and Related Records
Update any record and perform actions, like send an email. This more flexible flow runs after the record is saved to the database.
3
Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed
4
Cancel
Done

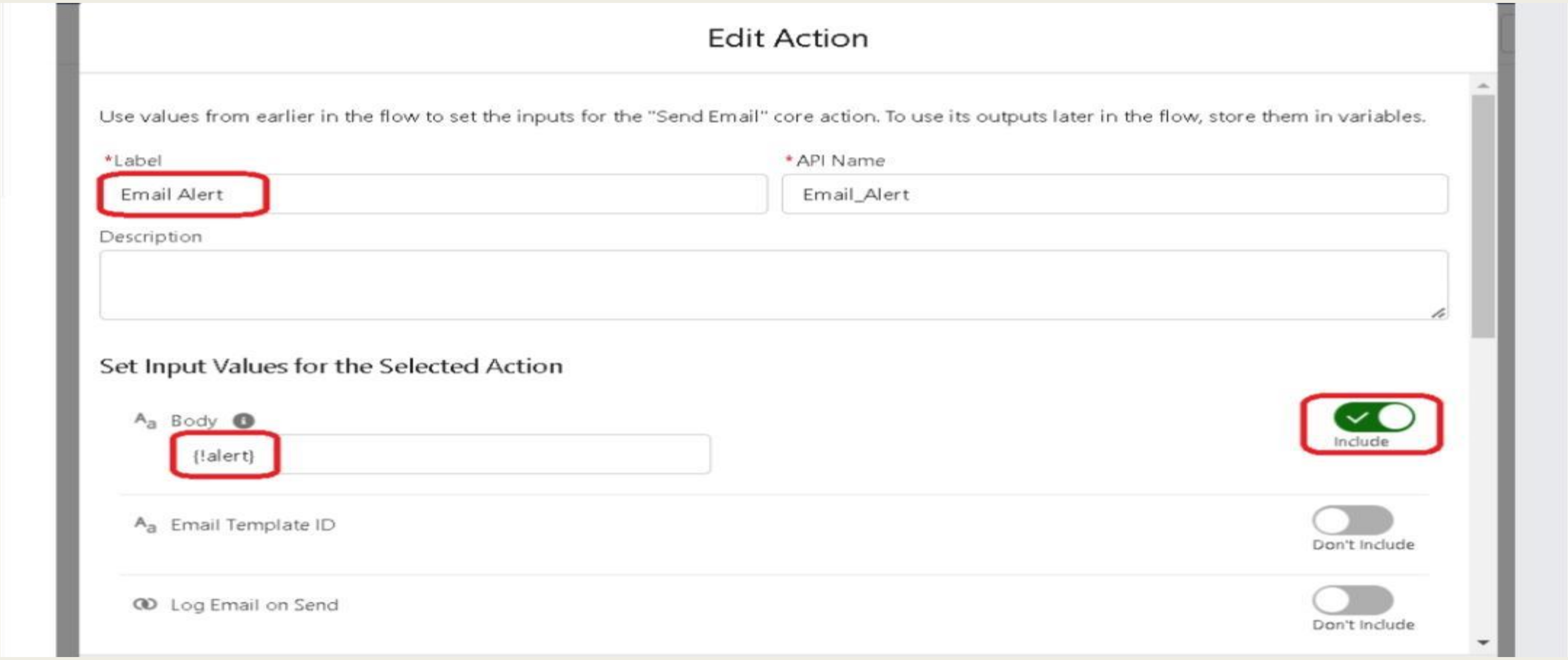

Record-Triggered Flow
Start
Run Immediately
Add Element
Search...
Collection Sort
Collection Filter
Data
Create Records
Update Records
Get Records
Delete Records

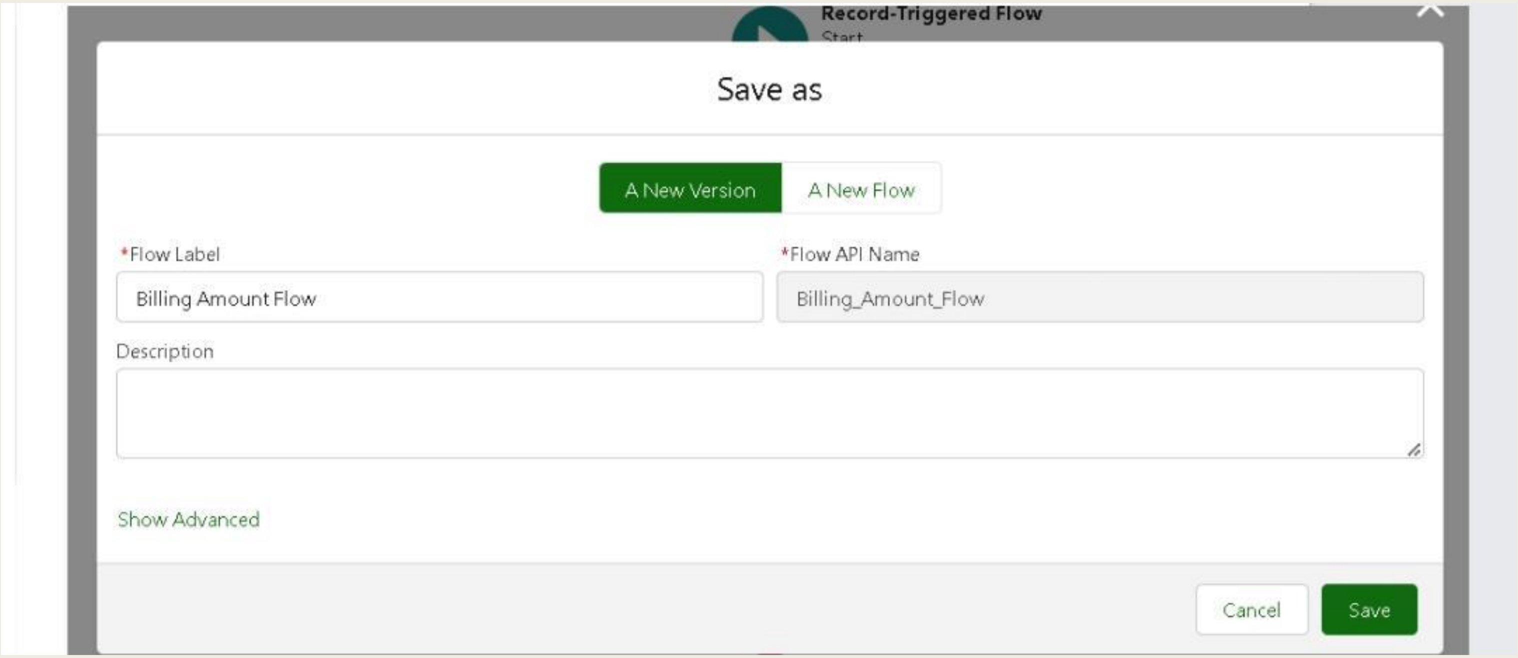

Edit Update Records
Update Salesforce records using values from the flow.
*Label
Amount Update
*API Name
Amount_Update
Description
*How to Find Records to Update and Set Their Values
Use the billing details and feedback record that triggered the flow
Update records related to the billing details and feedback record that triggered the flow
Use the IDs and all field values from a record or record collection
Specify conditions to identify records, and set fields individually
Set Filter Conditions
Condition Requirements to Update Record
All Conditions Are Met (AND)
Cancel
Done

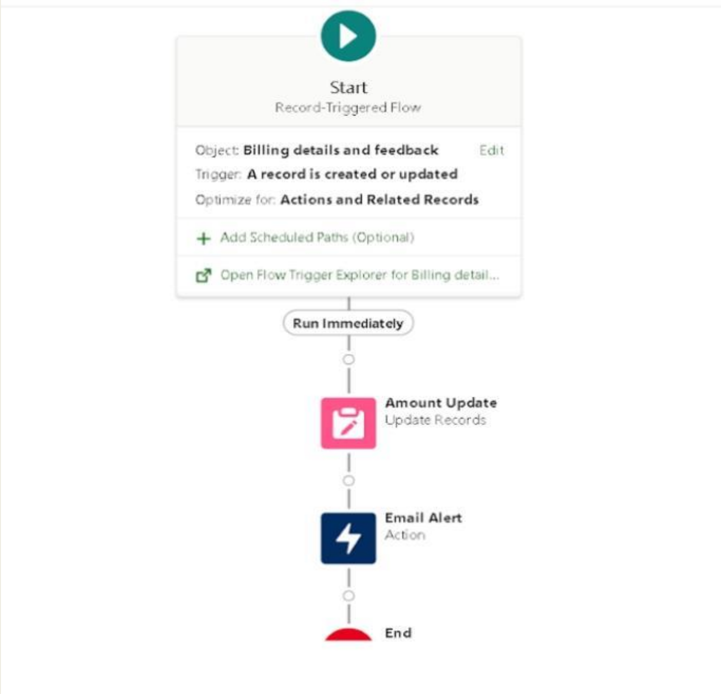

Set Filter Conditions
Condition Requirements to Update Record
All Conditions Are Met (AND)
Field
Operator
Value
Payment_Status__c
Equals
Completed
Add Condition
Set Field Values for the Billing details and feedback Record
Field
Value
Payment_Paid__c
$Record > Service records > Appointment > Service A...
Add Field
Cancel
Done


Flow Builder
Billing Amount Flow - V1
Select Elements
Toolbox
Manager
Search this flow...
New Resource
RESOURCES


Edit Text Template
*API Name
alert
Description
*Body
Insert a resource...
View as Plain Text
Dear {!$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},
Done


Edit Action
Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.
*Label
Email Alert
*API Name
Email_Alert
Description
Set Input Values for the Selected Action
Body
{!alert}
Include
Email Template ID
Don't Include
Log Email on Send
Don't Include


Record-Triggered Flow
Start
Save as
A New Version
A New Flow
*Flow Label
Billing Amount Flow
*Flow API Name
Billing_Amount_Flow
Description
Show Advanced
Cancel
Save


Start
Record-Triggered Flow
Object: Billing details and feedback
Edit
Trigger: A record is created or updated
Optimize for: Actions and Related Records
Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Billing detail...
Run Immediately
Amount Update
Update Records
Email Alert
Action
End

APEX TRIGGER

Apex can be invoked by using triggers. Apex triggers enable you to perform custom actions

APEX HANDLER

```
public class AmountDistributionHandler {

    public static void amountDist(list<Appointment__c> listApp){
        list<Service_records__c> serList = new list <Service_records__c>();

        for(Appointment__c app : listApp){
            if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 10000;
            }
            else if(app.Maintenance_service__c == true && app.Repairs__c == true){
                app.Service_Amount__c = 5000;
            }
            else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 8000;
            }
            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 7000;
            }
            else if(app.Maintenance_service__c == true){
```

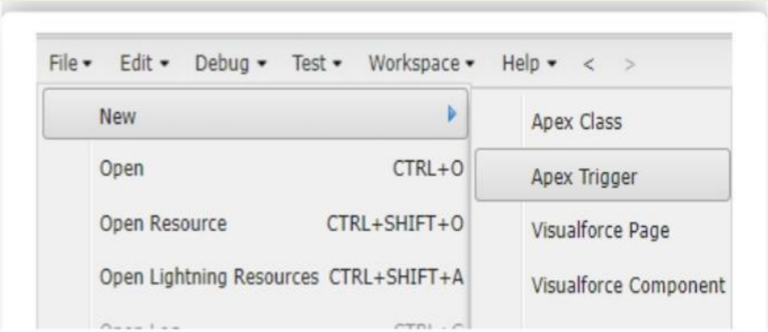
AmountDistribution.apxt | AmountDistributionHandler.apxc

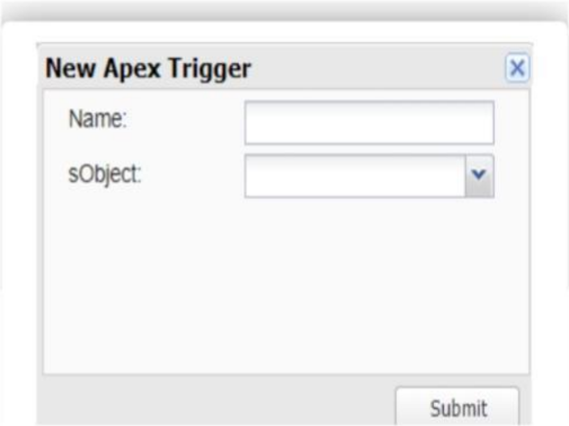
Code Coverage: None | API Version: 58

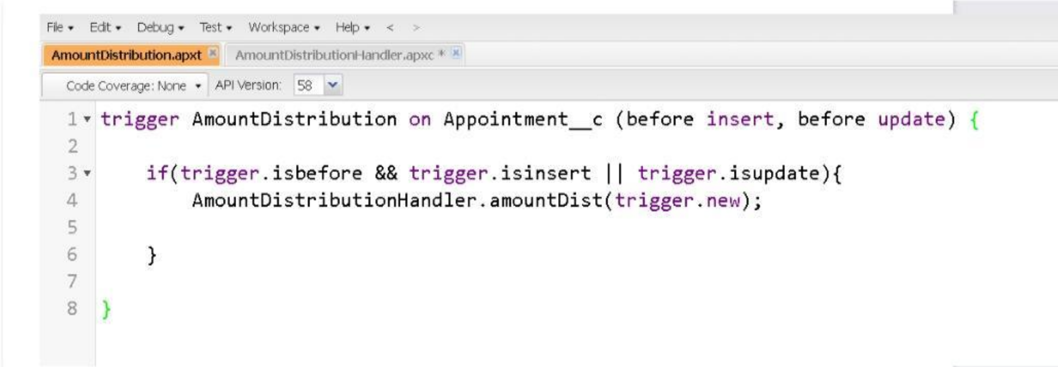
```
            }
            else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 8000;
            }
            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 7000;
            }
            else if(app.Maintenance_service__c == true){
                app.Service_Amount__c = 2000;
            }
            else if(app.Repairs__c == true){
                app.Service_Amount__c = 3000;
            }
            else if(app.Replacement_Parts__c == true){
                app.Service_Amount__c = 5000;
            }

        }
    }
}
```

Code:

```
public class AmountDistributionHandler {

   public static void amountDist(list<Appointment__c> listApp){
      list<Service_records__c> serList = new list <Service_records__c>
();

      for(Appointment__c app : listApp){
         if(app.Maintenance_service__c == true && app.Repairs__c ==
true && app.Replacement_Parts__c == true){
            app.Service_Amount__c = 10000;
         }
         else if(app.Maintenance_service__c == true && app.Repairs__c
== true){
            app.Service_Amount__c = 5000;
         }
         else if(app.Maintenance_service__c == true &&
app.Replacement_Parts__c == true){
            app.Service_Amount__c = 8000;
         }
         else if(app.Repairs__c == true && app.Replacement_Parts__c
== true){
            app.Service_Amount__c = 7000;
       }
       else if(app.Maintenance_service__c == true){
          app.Service_Amount__c = 2000;
       }
       else if(app.Repairs__c == true){
          app.Service_Amount__c = 3000;
       }
       else if(app.Replacement_Parts__c == true){
          app.Service_Amount__c = 5000;
       }


  }
  }
}
```







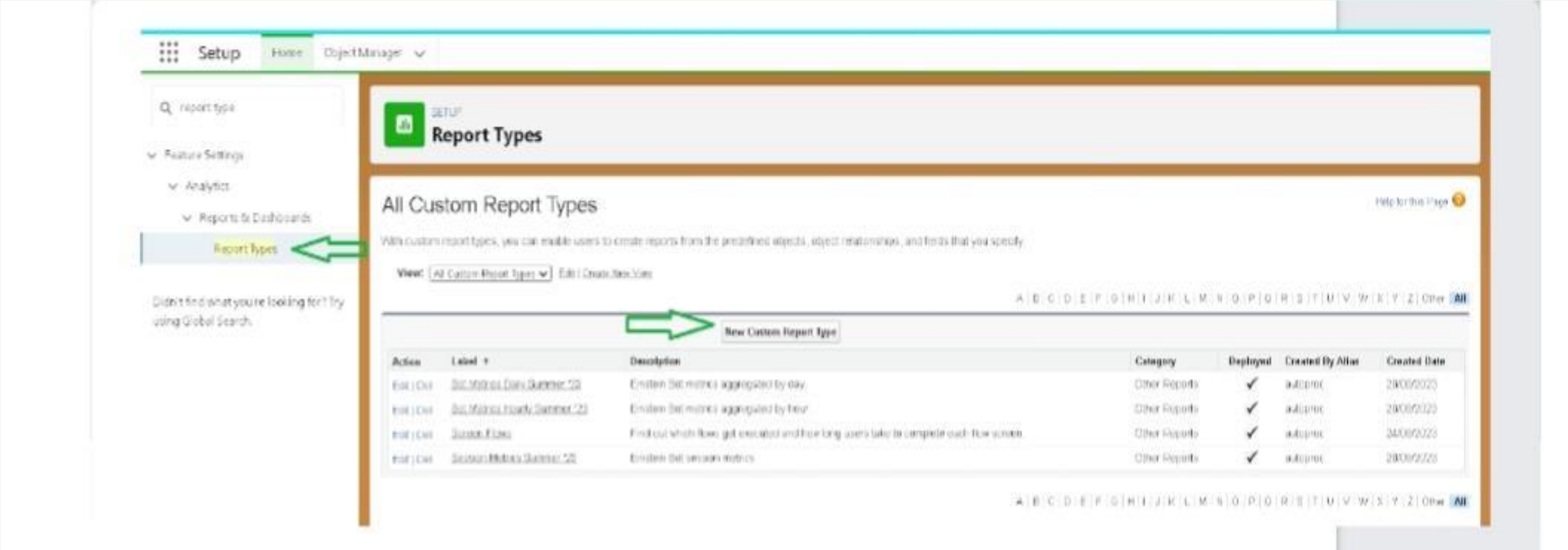
Code:

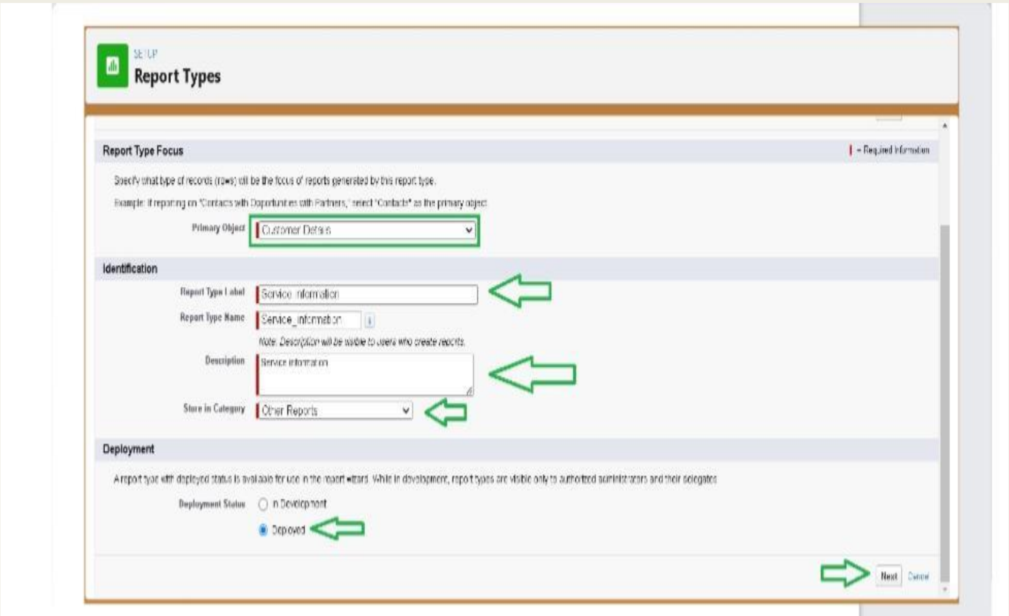
```
trigger AmountDistribution on Appointment__c (before insert,
before update) {

  if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
    AmountDistributionHandler.amountDist(trigger.new);

  }

}
```

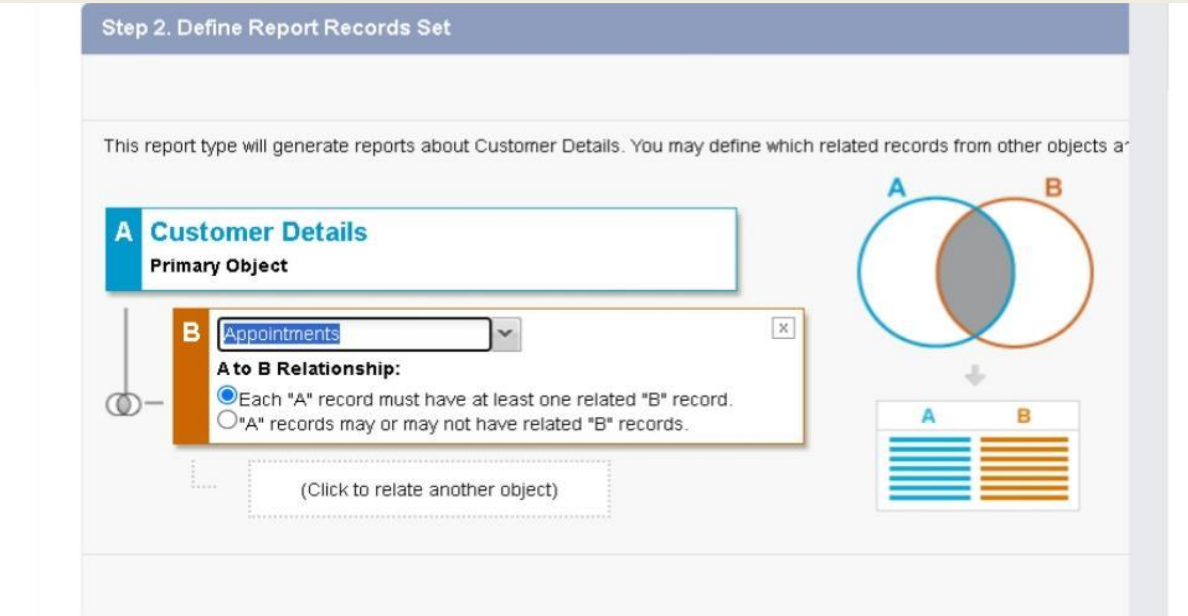
REPORTS

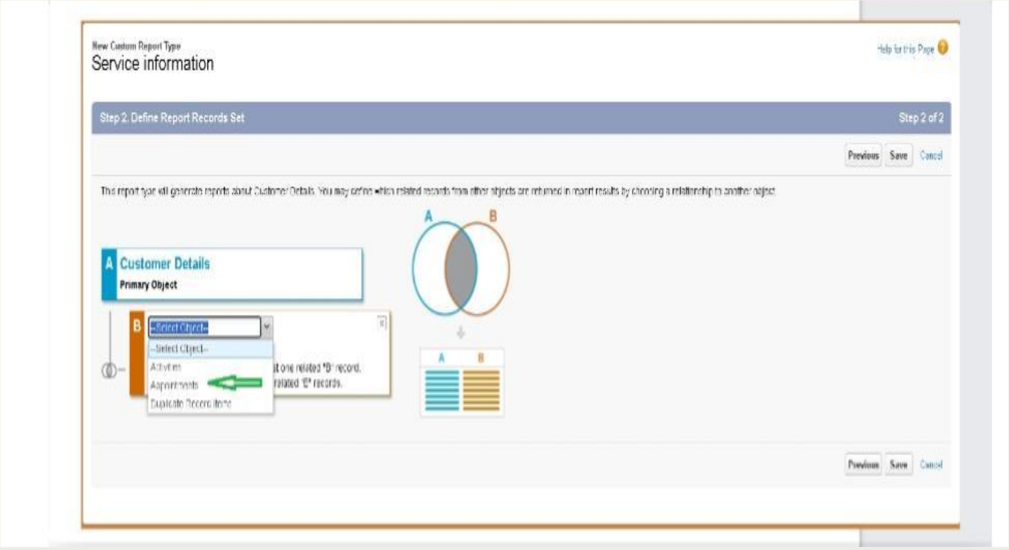
Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations

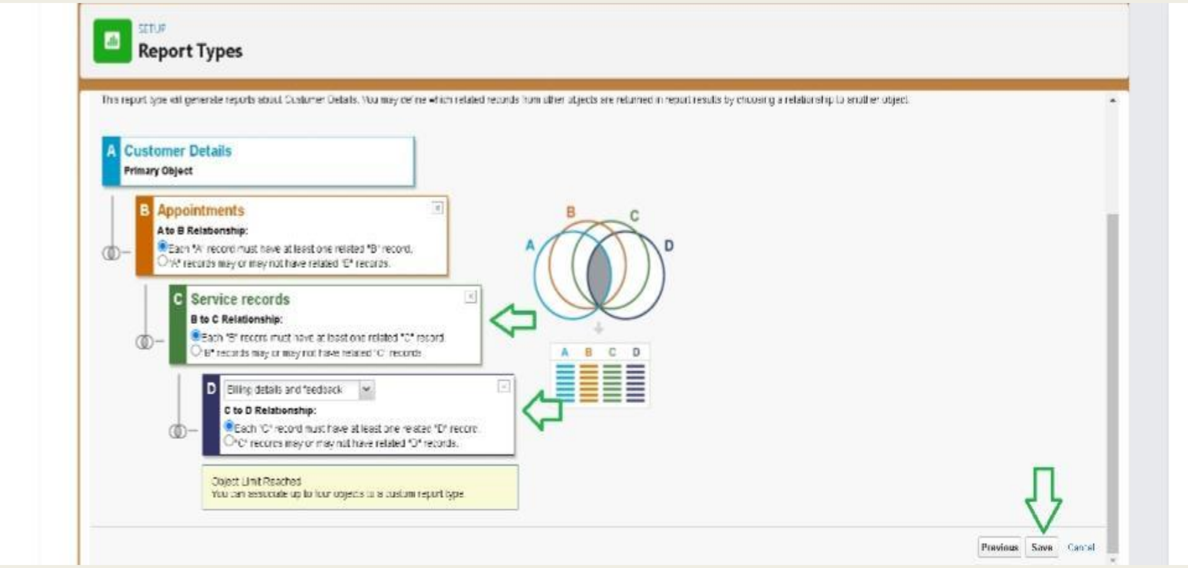
CREATE REPORT TYPE



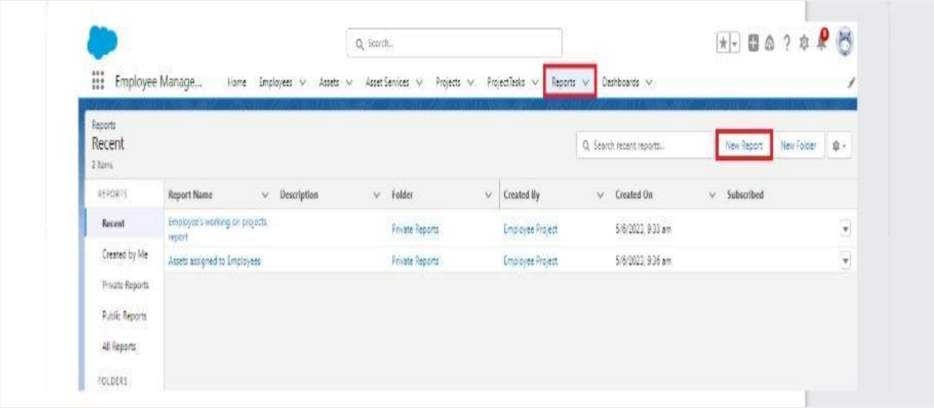

SETUP
Report Types
Report Type Focus
Identification
Deployment
Next

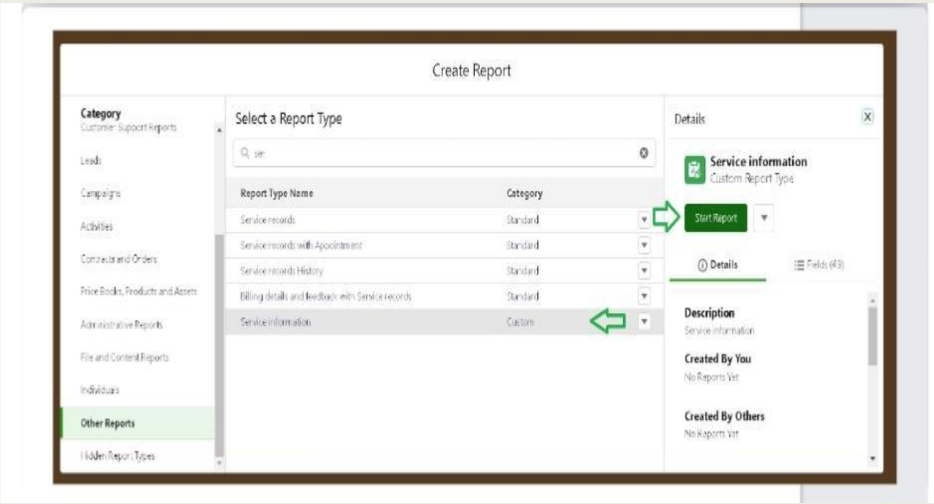

Step 2. Define Report Records Set
This report type will generate reports about Customer Details. You may define which related records from other objects a
A Customer Details
Primary Object
B Appointments
A to B Relationship:
Each "A" record must have at least one related "B" record.
"A" records may or may not have related "B" records.
(Click to relate another object)
A
B

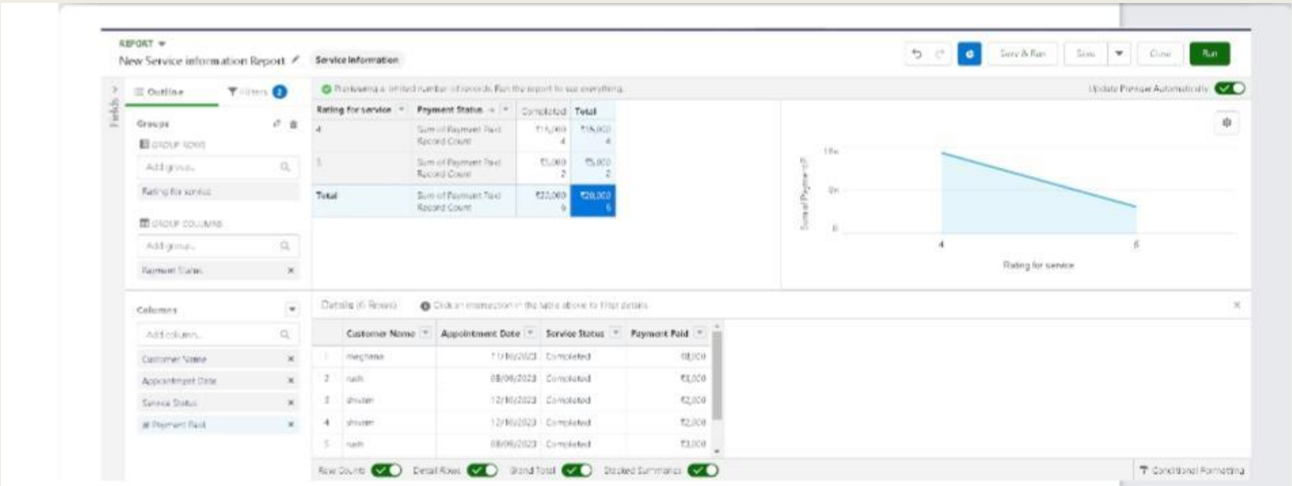

Service information
Step 2. Define Report Records Set
Step 2 of 2
Previous
Save
Customer Details
Primary Object


SETUP
Report Types
Customer Details
Primary Object
Appointments
A to B Relationship:
Service records
B to C Relationship:
C to D Relationship:
Previous
Save
Cancel

CREATE REPORT



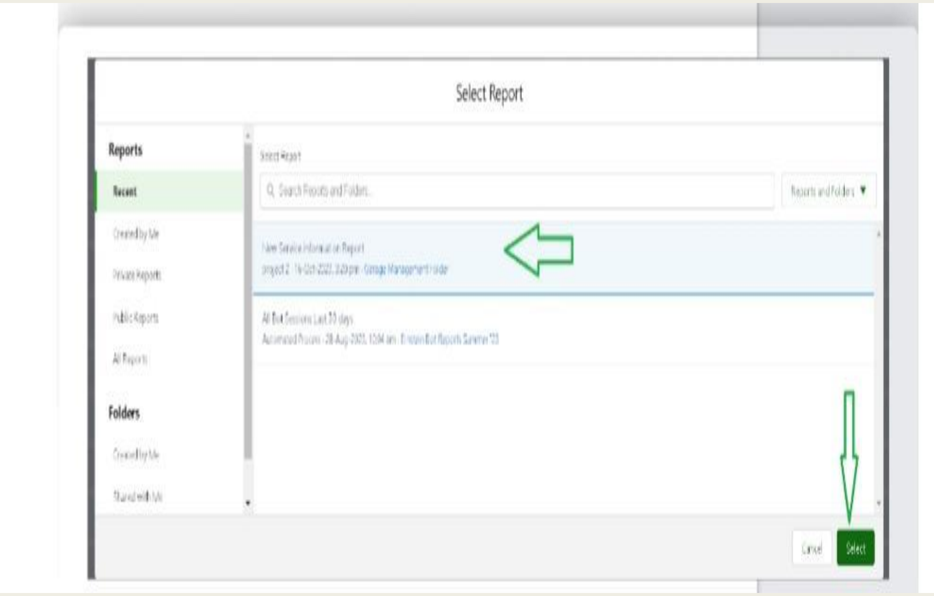




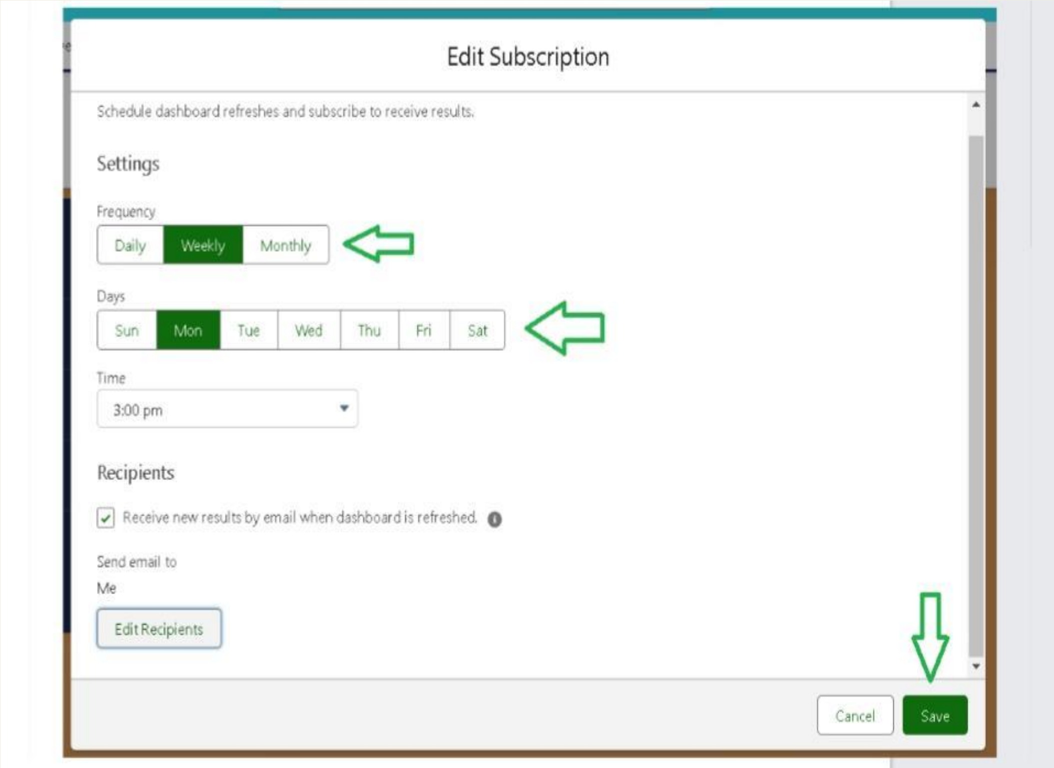
DASHBOARDS

Dashboards help you visually understand changing business conditions so you can make decisions based on the real-time data you've gathered with reports

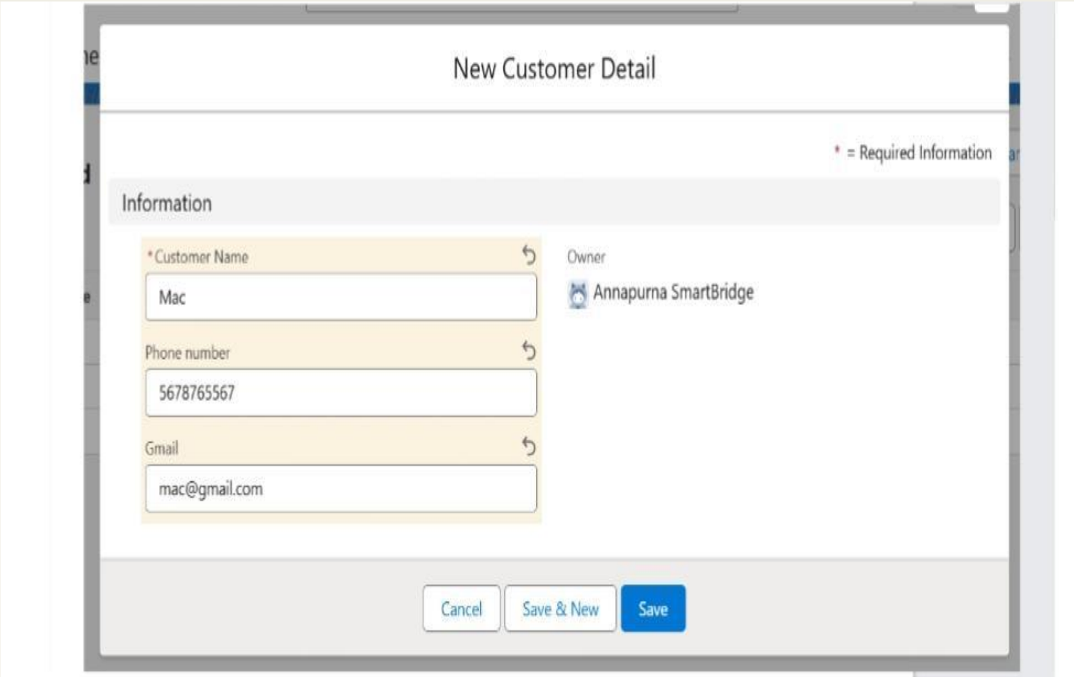
CREATE DASHBOARD

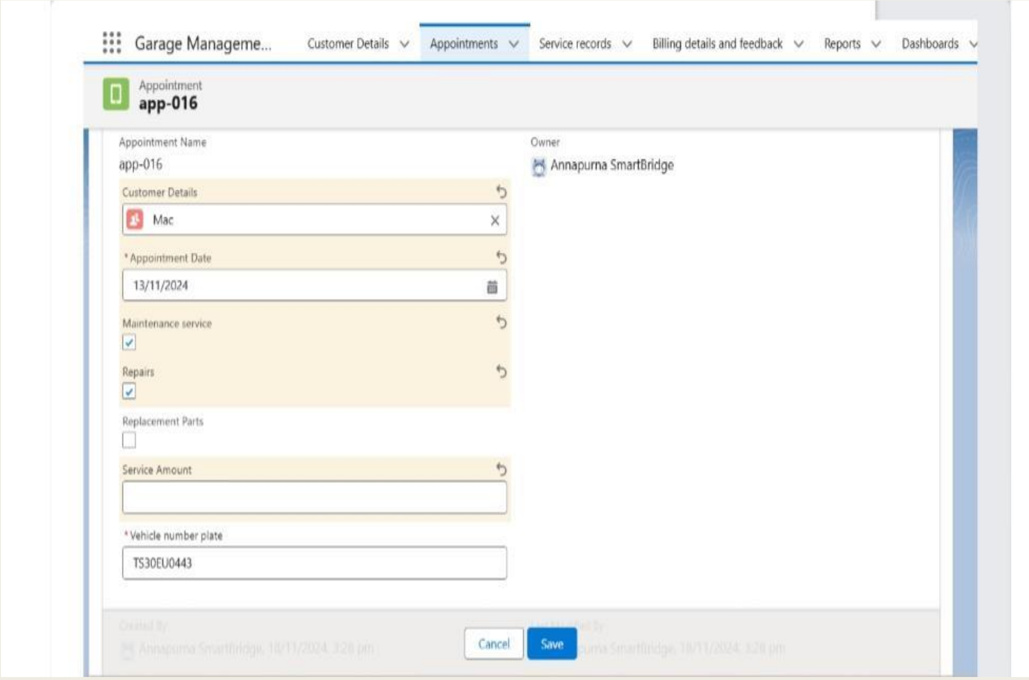

Select Report
Reports
Folders

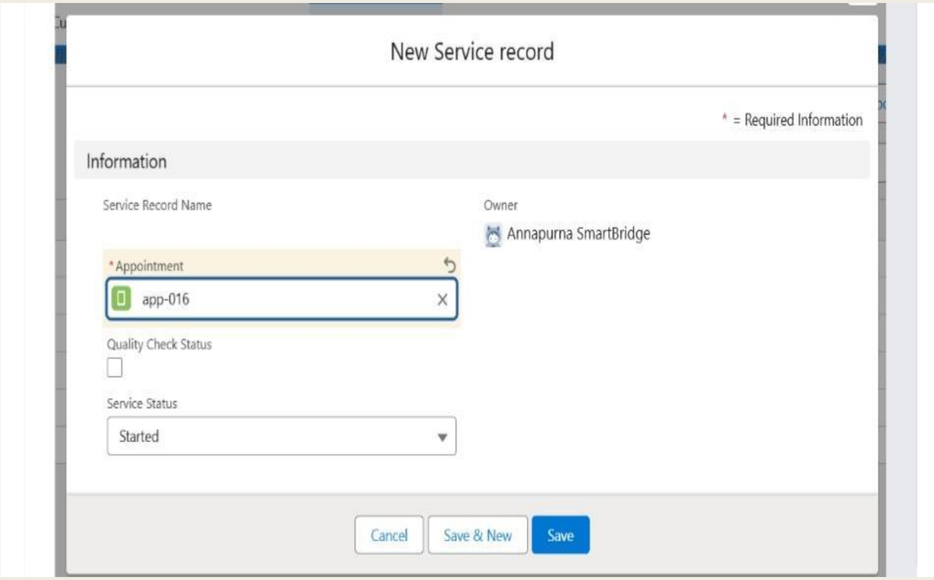

New Service information Report
Payment Status
Completed
Sum of Payment Paid
₹15K
₹10K
₹5K
₹0
4
5
Rating for service
View Report (New Service information Report)

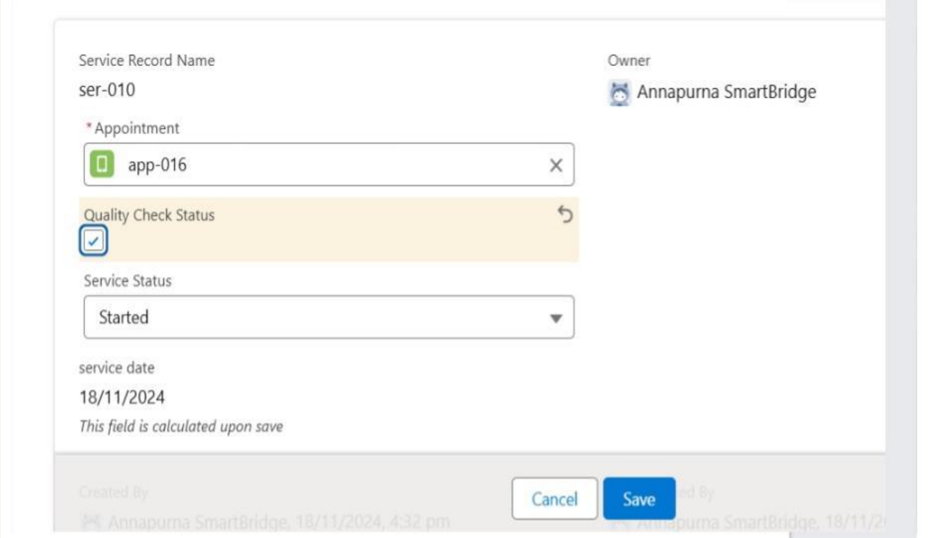

Edit Subscription
Schedule dashboard refreshes and subscribe to receive results.
Settings
Frequency
Daily
Weekly
Monthly
Days
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Time
3:00 pm
Recipients
Receive new results by email when dashboard is refreshed.
Send email to
Me
Edit Recipients
Cancel
Save

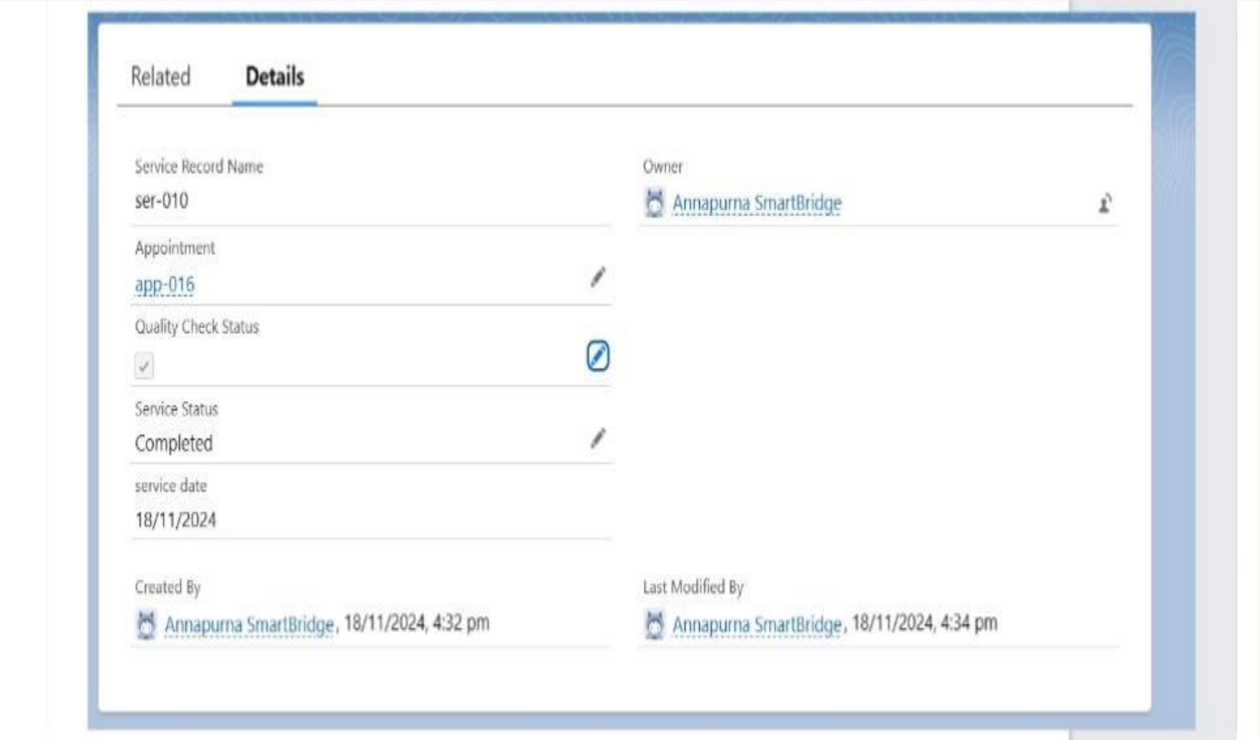
USER ADOPTION
CREATING RECORDS



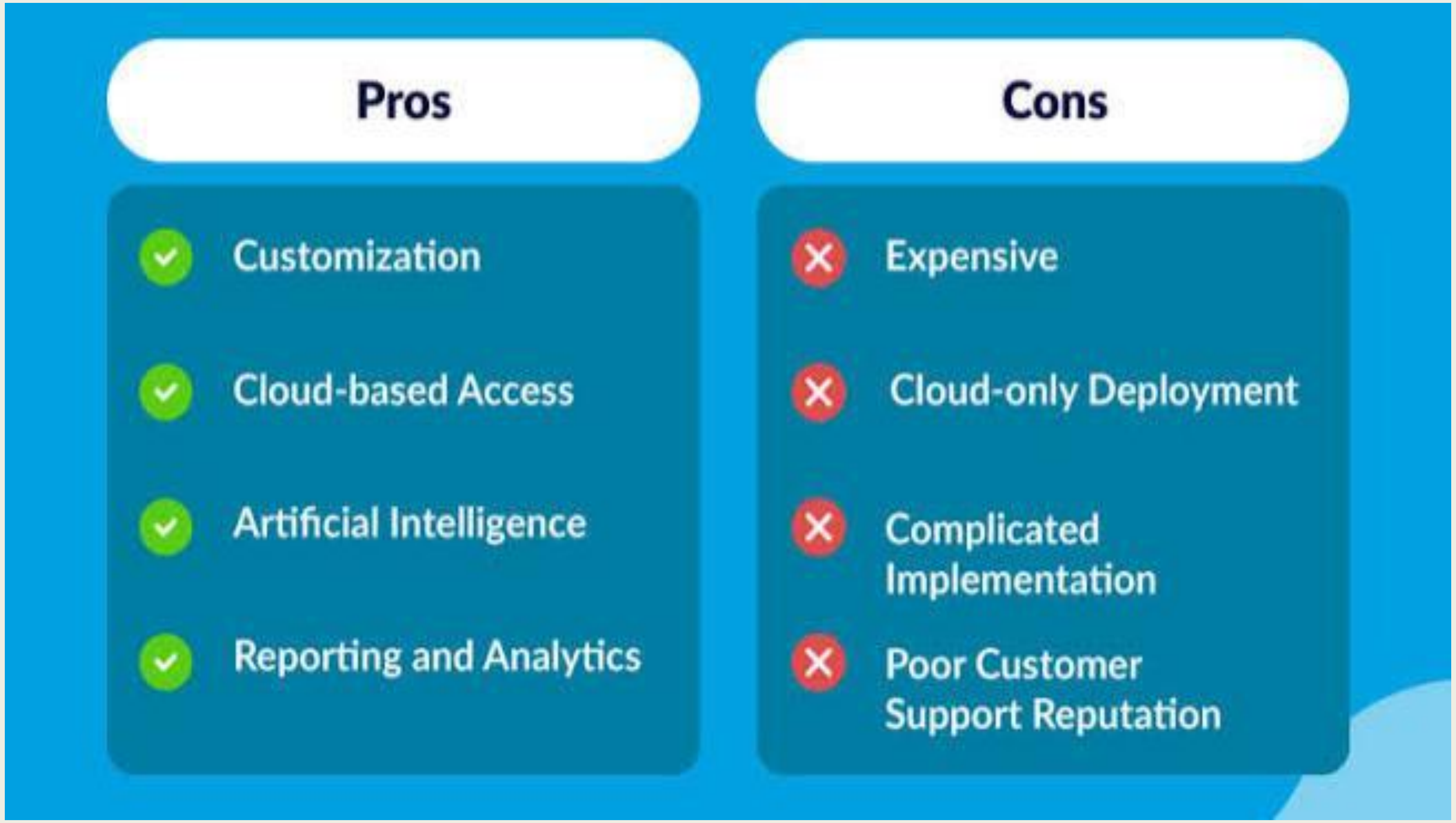



New Service record
* = Required Information
Information
Service Record Name
Owner
Annapurna SmartBridge
*Appointment
app-016
Quality Check Status
Service Status
Started
Cancel
Save & New
Save


Service Record Name
ser-010
Owner
Annapurna SmartBridge
*Appointment
app-016
Quality Check Status
Service Status
Started
service date
18/11/2024
This field is calculated upon save
Cancel
Save


Related
Details
Service Record Name
ser-010
Owner
Annapurna SmartBridge
Appointment
app-016
Quality Check Status
Service Status
Completed
service date
18/11/2024
Created By
Annapurna SmartBridge, 18/11/2024, 4:32 pm
Last Modified By
Annapurna SmartBridge, 18/11/2024, 4:34 pm

ADVANTAGES & DISADVANTAGES



CONCLUSION

Implementing a garage management system in Salesforce can be a powerful solution for automotive service centers and garages.

By leveraging the capabilities of Salesforce, organizations can streamline their operations, improve customer satisfaction, and increase efficiency.