

Garage Management system

The Garage Management System is a valuable tool for automotive repair facilities, helping them deliver top-notch service, increase operational efficiency, and build lasting customer relationships. With its user-friendly interface and powerful features, GMS empowers garages to thrive in a competitive market while ensuring a seamless and satisfying experience for both customers and staff. The Garage Management System (GMS) is a comprehensive software solution designed to streamline and optimize the operations of automotive repair facilities, service centers, and garages. It provides an array of features tailored to meet the needs of mechanics, service advisors, and business owners, ensuring smoother workflows and higher customer satisfaction.

Features of a Comprehensive Garage Management System

1. Appointment Scheduling

- **Streamlined Booking:** Simplifies the booking process for customers, enabling hassle-free appointment scheduling.
- **Efficient Resource Allocation:** Helps staff manage daily schedules effectively, reducing downtime and maximizing resource utilization.

2. Vehicle Management

- **Service History Tracking:** Maintains detailed records of vehicles, including service history, repairs, and maintenance schedules.
- **Status Updates:** Tracks vehicle status during servicing to enhance communication with customers and improve transparency.

3. Customer Relationship Management (CRM)

- **Customer Profiles:** Stores customer details and preferences, enabling personalized service delivery.
- **Automated Notifications:** Sends service reminders, follow-ups, and promotional offers to foster customer loyalty.

4. Inventory and Spare Parts Management

- **Stock Monitoring:** Tracks spare parts stock levels to ensure availability.
- **Automated Reordering:** Prevents stock outs by automating the reorder process.
- **Mechanic Support:** Ensures tools and parts are readily available for mechanics to complete jobs efficiently.

5. Billing and Invoicing

- **Professional Invoicing:** Quickly generates accurate and professional invoices.
- **Payment Flexibility:** Supports multiple payment methods, applies discounts, and calculates taxes seamlessly.

6. Work Order Management

- **Detailed Work Orders:** Creates comprehensive work orders, including tasks, costs, and timelines.
- **Prioritization:** Assists staff in prioritizing tasks, ensuring timely job completion.

7. Reporting and Analytics

- **Performance Insights:** Tracks key performance indicators like revenue, job completion rates, and customer feedback.
- **Trend Analysis:** Identifies trends and highlights areas for improvement to optimize business performance.

What Is Salesforce?

Salesforce is a comprehensive customer success platform designed to help businesses streamline their processes for sales, customer service, marketing, analytics, and customer engagement. It empowers you to manage your business operations seamlessly from anywhere by leveraging cloud-based tools.

With Salesforce's standard features and products, you can:

- Manage customer and prospect relationships
- Collaborate efficiently with employees and partners
- Securely store and organize data in the cloud

Why Salesforce?

Before tools like Salesforce, businesses often relied on scattered systems such as spreadsheets, emails, and manual task tracking to manage contacts, follow-ups, and deals. Salesforce consolidates all these operations into one unified platform.

For a better understanding, you can check this quick video explanation: [Salesforce Overview](#)

Creating a Salesforce Developer Account

Follow these steps to create a **Salesforce Developer Org** for hands-on practice:

1. Go to [Salesforce Developer Signup](#).
2. Fill in the sign-up form with the following details:
 - **First Name & Last Name**
 - **Email** (use a valid email for activation)
 - **Role**: Developer
 - **Company**: Use your college name
 - **Country**: India
 - **Zip code**: Your area pin code
 - **Username**: Create a unique username in the format username@organization.com
 - Note: This doesn't have to be a real email ID. Example:
john.doe@mycollege.com
3. After filling in the details, click "**Sign Me Up**".
4. Check your email for the activation link and follow the instructions to complete the setup.

Once your account is ready, you'll have full access to Salesforce's developer environment to practice and build applications.

Build enterprise-quality apps fast to bring your ideas to life

- Build apps fast with drag and drop tools
- Customize your data model with clicks
- Go further with Apex code
- Integrate with anything using powerful APIs
- Stay protected with enterprise-grade security
- Customize UI with clicks or any leading-edge web framework

Sign up for your Salesforce Developer Edition
A Salesforce Platform environment for free.

Complete the form to get access to the Salesforce Developer Edition.

First Name*
Your first name

Last Name*
Your last name

Email*
Your email address

Role*
Your job role

Company*
Company Name

Country/Region*
Country/Region

Postal Code*
Your postal code

Username*
jane@company.sandbox

Your username must be in the form of an email address (it does not have to be real). It must be unique and cannot be associated with another Salesforce login credential. [Read more about username recommendations.](#)

☐ I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement.

Sign me Up

Already have a Salesforce Developer Environment?
[Log In](#)

Object

What Is an Object in Salesforce?

In Salesforce, **objects** are database tables that allow you to store data specific to an organization. They serve as the building blocks for data management in Salesforce.

Types of Salesforce Objects

Salesforce objects are categorized into two types:

1. Standard Objects

- These are pre-defined objects provided by Salesforce.
- Examples include: Users, Contracts, Reports, Dashboards, Accounts, etc.

2. Custom Objects

- These are objects created by users to store information unique to their organization.
- Custom objects form the heart of any application and provide a structured way to share and manage data.

How to Create Custom Objects in Salesforce

1. Create Customer Details Object

Follow these steps to create the "Customer Details" object:

1. Go to **Setup** → **Object Manager** → **Create** → **Custom Object**.
2. Enter the following details:
 - **Label Name:** Customer Details
 - **Plural Label Name:** Customer Details
 - **Record Name Label:** Customer Name
 - **Data Type:** Text
3. Enable the following options:
 - **Allow Reports**
 - **Track Field History**
 - **Allow Search**
4. Click **Save**.

Customer Details

Custom Object Definition Edit

Save Save & New Cancel

Custom Object Information

Required Information

The singular and plural labels are used in tabs, page layouts, and reports.

Be careful when changing the name or label as it may affect existing integrations and merge templates.

Label **Customer Details** Example: Account
Plural Label **Customer Details** Example: Accounts
Starts with vowel sound ☐

The Object Name is used when referencing the object via the API.

Object Name **Customer_Details** Example: Account

Description

Context-Sensitive Help Setting ☒ Open the standard Salesforce.com Help & Training window
☐ Open a window using a Visualforce page

Content Name **--None--**

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name **Customer Name** Example: Account Name

Data Type **Text** Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

Optional Features

- ☒ Allow Reports
- ☐ Allow Activities
- ☒ Track Field History
- ☐ Allow in Chatter Groups
- ☐ Enable Licensing [i](#)

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more.](#)

- ☒ Allow Sharing
- ☒ Allow Bulk API Access
- ☒ Allow Streaming API Access

2. Create Appointment Object

Steps to create the "Appointment" object:

1. Go to **Setup** → **Object Manager** → **Create** → **Custom Object**.
2. Enter the following details:
 - **Label Name:** Appointment
 - **Plural Label Name:** Appointments
 - **Record Name Label:** Appointment Name
 - **Data Type:** Auto Number
 - **Display Format:** app-{000}
 - **Starting Number:** 1
3. Enable the following options:
 - **Allow Reports**
 - **Track Field History**
 - **Allow Search**
4. Click **Save**.

Edit Custom Object

Appointment

[Help for this Page](#)

Custom Object Definition Edit

SaveSave & NewCancel

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and reports. Be careful when changing the name or label as it may affect existing integrations and merge templates.

Label Example: Account

Plural Label Example: Accounts

Starts with vowel sound ☐

The Object Name is used when referencing the object via the API.

Object Name Example: Account

Description

Context-Sensitive Help Setting ☒ Open the standard Salesforce.com Help & Training window
☐ Open a window using a Visualforce page

Content Name

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name Example: Account Name

Data Type Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

Display Format Example: A-{0000} [What Is This?](#)

Optional Features

☒ Allow Reports

☐ Allow Activities

☒ Track Field History

☐ Allow in Chatter Groups

☐ Enable Licensing [i](#)

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more.](#)

☒ Allow Sharing

3. Create Service Records Object

Steps to create the "Service Records" object:

1. Go to **Setup** → **Object Manager** → **Create** → **Custom Object**.
2. Enter the following details:
 - **Label Name:** Service Records
 - **Plural Label Name:** Service Records
 - **Record Name Label:** Service Records Name
 - **Data Type:** Auto Number
 - **Display Format:** ser-{000}
 - **Starting Number:** 1
3. Enable the following options:
 - **Allow Reports**
 - **Track Field History**
 - **Allow Search**
4. Click **Save**.

Edit Custom Object

Service records

[Help for this Page](#)

Custom Object Definition Edit

SaveSave & NewCancel

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and reports. Be careful when changing the name or label as it may affect existing integrations and merge templates.

Label

Service records

Example: Account

Plural Label

Service records

Example: Accounts

Starts with vowel sound

☐

The Object Name is used when referencing the object via the API.

Object Name

Service_records

Example: Account

Description

Context-Sensitive Help Setting

☒ Open the standard Salesforce.com Help & Training window

☐ Open a window using a Visualforce page

Content Name

--None--

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name

Service records Name

Example: Account Name

Data Type

Auto Number

Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

Display Format

ser-{000}

Example: A-{0000} [What Is This?](#)

Optional Features

☒ Allow Reports

☐ Allow Activities

☒ Track Field History

☐ Allow in Chatter Groups

☐ Enable Licensing [i](#)

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more.](#)

☒ Allow Sharing

4. Create Billing Details and Feedback Object

Steps to create the "Billing Details and Feedback" object:

1. Go to **Setup** → **Object Manager** → **Create** → **Custom Object**.
2. Enter the following details:
 - **Label Name:** Billing Details and Feedback
 - **Plural Label Name:** Billing Details and Feedback
 - **Record Name Label:** Billing Details and Feedback Name
 - **Data Type:** Auto Number
 - **Display Format:** bill-{000}
 - **Starting Number:** 1
3. Enable the following options:
 - **Allow Reports**
 - **Track Field History**
 - **Allow Search**
4. Click **Save**.

Edit Custom Object
Billing details and feedback [Help for this Page](#)

Custom Object Definition Edit Save Save & New Cancel

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and reports. Be careful when changing the name or label as it may affect existing integrations and merge templates.

Label Example: Account

Plural Label Example: Accounts

Starts with vowel sound ☐

The Object Name is used when referencing the object via the API.

Object Name Example: Account

Description

Context-Sensitive Help Setting ☒ Open the standard Salesforce.com Help & Training window
☐ Open a window using a Visualforce page

Content Name

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name Example: Account Name

Data Type **Warning:** If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

Display Format Example: A-{0000} [What Is This?](#)

Optional Features

☒ Allow Reports
☐ Allow Activities
☒ Track Field History
☐ Allow in Chatter Groups
☐ Enable Licensing [i](#)

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more.](#)

☒ Allow Sharing

Tabs

What Is a Tab in Salesforce?

A **tab** in Salesforce is a user interface element that allows you to create, view, and manage records for objects. Tabs provide quick access to data and functionality within Salesforce.

Types of Tabs in Salesforce

1. Custom Tabs

- Custom tabs display data from custom objects.
- They behave like standard Salesforce tabs such as Accounts, Contacts, and Opportunities.
- Use Case: Ideal for custom applications you build on Salesforce.

2. Web Tabs

- Web tabs display web content or applications within the Salesforce window.
- Use Case: Provide users quick access to frequently used websites or external apps without leaving Salesforce.

3. sales force Tabs

- sales force tabs display salesforce pages.
- These tabs look and function like standard Salesforce tabs.
- Use Case: Ideal for custom sales force page implementations.

4. Lightning Component Tabs

- Lightning Component tabs allow you to include Lightning components in the navigation menu for Lightning Experience and the mobile app.
- Use Case: Display custom Lightning components for specific business needs.

5. Lightning Page Tabs

- Lightning Page tabs let you add Lightning Pages to the navigation menu.
- **Important Note:** Lightning Page tabs don't appear on the All Tabs page or in the Available Tabs list when customizing tabs for apps.
- Use Case: Integrate Lightning Pages into mobile or desktop navigation.

How to Create Custom Tabs in Salesforce

1. Create a Custom Tab for "Customer Details"

Follow these steps:

1. Go to **Setup** → Type "Tabs" in the Quick Find bar → Click on **Tabs** → Click **New** under "Custom Object Tabs".
2. Select the object: **Customer Details**.
3. Choose the **Tab Style** → Click **Next**.
4. **Add to Profiles Page:** Keep the default settings → Click **Next**.
5. **Add to Custom App:** Uncheck "Include Tab".
6. Ensure "Append tab to users' existing personal customizations" is checked.
7. Click **Save**.

2. Create Tabs for Remaining Objects

Follow the same steps as above to create tabs for the following objects:

- **Appointments**
- **Service Records**
- **Billing Details and Feedback**

The screenshot shows the Salesforce Setup interface for managing custom tabs. At the top, there's a 'SETUP Tabs' header. Below it, a 'Custom Tabs' section provides an overview: 'You can create new custom tabs to extend Salesforce functionality or to build new application functionality.' It explains that Custom Object tabs look like standard tabs, Web tabs embed external content, Visualforce tabs embed Visualforce pages, Lightning Component tabs add components to the navigation menu, and Lightning Page tabs add Lightning Pages to Lightning Experience and the mobile app. The main area is divided into five sections: Custom Object Tabs, Web Tabs, Visualforce Tabs, Lightning Component Tabs, and Lightning Page Tabs. The 'Custom Object Tabs' section is active, showing a table with columns for Action, Label, Tab Style, and Description. It lists six tabs: Appointments (Hexagon), Billing details and feedback (Credit card), Customer Details (Helicopter), Equipment Maintenance Items (Factory), Service records (Computer), and Vehicles (Car). The other sections (Web Tabs, Visualforce Tabs, Lightning Component Tabs) all show 'No [type] Tabs have been defined'. The 'Lightning Page Tabs' section shows one tab: 'Create Default Data' (Star), with the description 'Created by Lightning App Builder'.

Action	Label	Tab Style	Description
Edit Del	Appointments	Hexagon	
Edit Del	Billing details and feedback	Credit card	
Edit Del	Customer Details	Helicopter	
Edit Del	Equipment Maintenance Items	Factory	
Edit Del	Service records	Computer	
Edit Del	Vehicles	Car	

Action	Label	Tab Style	Description
Edit Del	Create Default Data	Star	Created by Lightning App Builder

Lightning App

What Is a Lightning App?

A **Lightning App** in Salesforce is a collection of objects, tabs, and other items designed to fulfill a specific purpose. It allows users to access everything they need in one place via the **navigation bar**.

Key Features of Lightning Apps:

- Custom branding with logos and colors.
- Ability to include a **Utility Bar** and **Lightning Page Tabs**.
- Enhanced productivity with easy app switching.

Steps to Create a Lightning App

1. Go to **Setup** → Type "App Manager" in the Quick Find bar → Select **App Manager** → Click **New Lightning App**.
2. **Fill out App Details:**
 - **App Name:** Garage Management Application
 - Click **Next**.
3. **App Options Page:** Keep the default settings → Click **Next**.
4. **Utility Items:** Keep the default settings → Click **Next**.
5. **Add Navigation Items:**
 - Search and add the following items:
 - **Customer Details**
 - **Appointments**
 - **Service Records**
 - **Billing Details and Feedback**
 - **Reports**
 - **Dashboards**
 - Use the arrow button to move the items → Click **Next**.
6. **Add User Profiles:**
 - Search for the **System Administrator** profile in the search bar.
 - Use the arrow button to add it.
7. Click **Save & Finish**.



SETUP

Lightning Experience App Manager

New Lightning App

New Connected App

25 Items • Sorted by App Name • Filtered by All appmenuItems - TabSet Type, App Type



	App Name ↑	Developer Name	Description	Last Modified ...	Ap...	V...
1	All Tabs	AllTabSet		07/12/2024, 7:19 pm	Classic	
2	Analytics Studio	Insights	Build CRM Analytics dashboards and apps	07/12/2024, 7:19 pm	Classic	✓
3	App Launcher	AppLauncher	App Launcher tabs	07/12/2024, 7:19 pm	Classic	✓
4	Automation	FlowsApp	Automate business processes and repetitive tasks.	07/12/2024, 7:24 pm	Lightning	✓
5	Bolt Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	07/12/2024, 7:22 pm	Lightning	✓
6	Business Rules Engine	ExpressionSetConsole	Create and maintain business rules that perform complex lookups and calculati...	07/12/2024, 7:19 pm	Lightning	✓
7	Community	Community	Salesforce CRM Communities	07/12/2024, 7:19 pm	Classic	✓
8	Content	Content	Salesforce CRM Content	07/12/2024, 7:19 pm	Classic	✓
9	Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	07/12/2024, 7:19 pm	Lightning	✓
10	Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	07/12/2024, 7:19 pm	Lightning	✓
11	Garage Management Applica...	Garage_Management_Applicati...		11/12/2024, 11:35 pm	Lightning	✓
12	How We Roll Maintenance	How_We_Roll_Maintenance		12/12/2024, 1:32 pm	Lightning	✓
13	Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	07/12/2024, 7:19 pm	Lightning	✓
14	Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	07/12/2024, 7:19 pm	Classic	✓
15	Platform	Platform	The fundamental Lightning Platform	07/12/2024, 7:19 pm	Classic	
16	Queue Management	QueueManagement	Create and manage queues for your business.	07/12/2024, 7:19 pm	Lightning	✓
17	Sales	Sales	The world's most popular sales force automation (SFA) solution	07/12/2024, 7:19 pm	Classic	
18	Sales	LightningSales	Manage your sales process with accounts, leads, opportunities, and more	07/12/2024, 7:19 pm	Lightning	✓
19	Sales Console	LightningSalesConsole	(Lightning Experience) Lets sales reps work with multiple records on one screen	07/12/2024, 7:19 pm	Lightning	✓
20	Salesforce Chatter	Chatter	The Salesforce Chatter social network, including profiles and feeds	07/12/2024, 7:19 pm	Classic	✓
21	Salesforce Scheduler Setup	LightningScheduler	Set up personalized appointment scheduling.	07/12/2024, 7:22 pm	Lightning	✓
22	Service	Service	Manage customer service with accounts, contacts, cases, and more	07/12/2024, 7:19 pm	Classic	✓
23	Service Console	LightningService	(Lightning Experience) Lets support agents work with multiple records across c...	07/12/2024, 7:19 pm	Lightning	✓
24	Site.com	Sites	Build pixel-perfect, data-rich websites using the drag-and-drop Site.com applic...	07/12/2024, 7:19 pm	Classic	

Fields

Fields in Salesforce

In Salesforce, **Fields** represent the data stored in the columns of a relational database. Fields hold valuable information for specific objects, making the searching, editing, and deletion of records simpler and quicker.

Types of Fields

1. **Standard Fields**
 - Predefined fields provided by Salesforce for every object (e.g., Name, Created Date).
2. **Custom Fields**
 - User-defined fields created for specific business requirements.

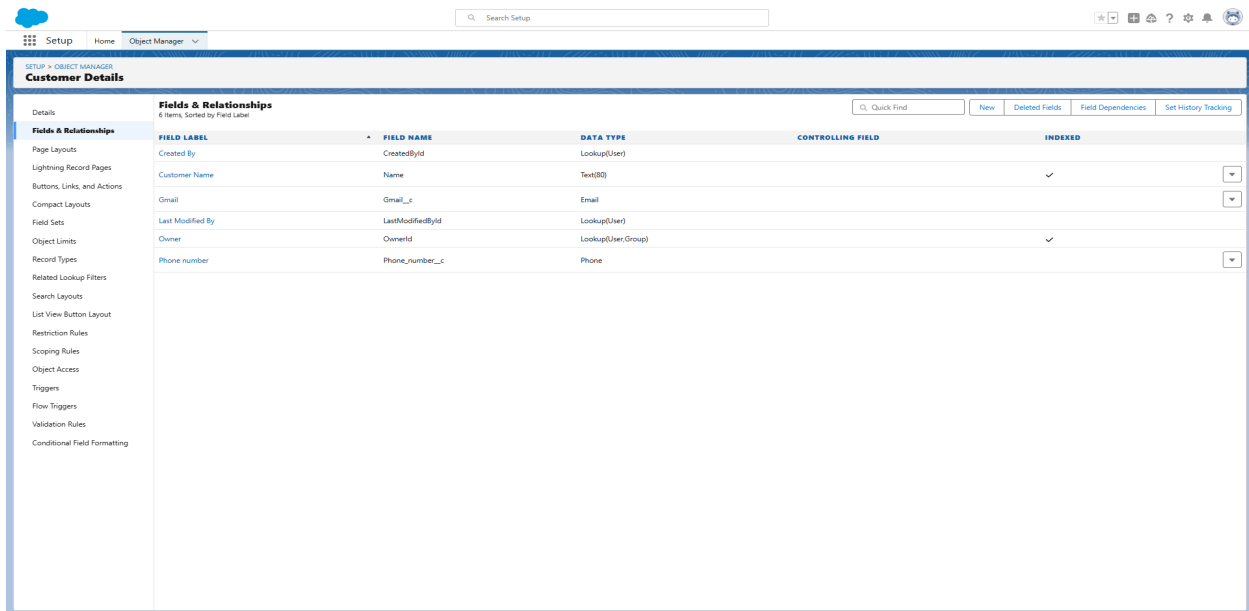
Creation of Fields for Customer Details Object

1. Phone Number Field

1. Go to **Setup** → Click on **Object Manager**.
2. Search for **Customer Details** → Click on the object.
3. Go to **Fields & Relationships** → Click **New**.
4. Select **Phone** as the Data Type → Click **Next**.
5. Fill in the following details:
 - **Field Label**: Phone Number
 - **Field Name**: Auto-generated
6. Click **Next** → **Next** → **Save & New**.

2. Gmail Field

1. Follow Steps 1 & 2 above.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Email** as the Data Type → Click **Next**.
4. Fill in the following details:
 - **Field Label**: Gmail
 - **Field Name**: Auto-generated
5. Click **Next** → **Next** → **Save & New**.



Creation of Lookup Fields

1. Lookup Field on Appointment Object

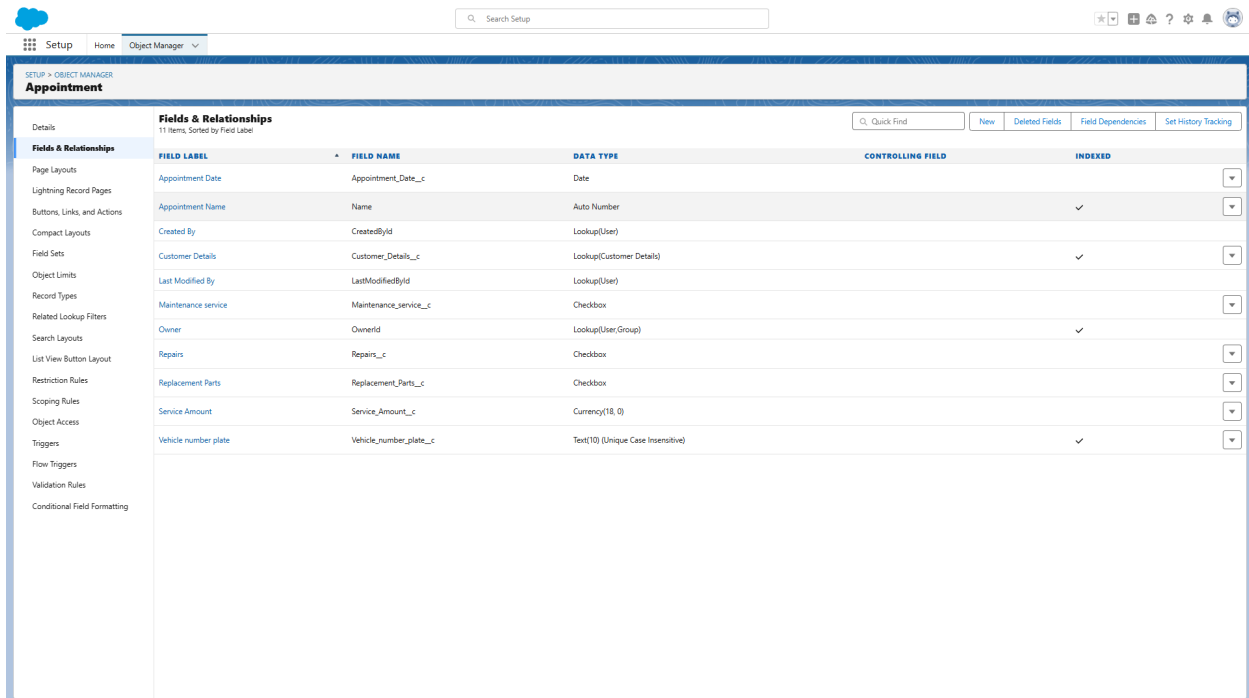
1. Go to **Setup** → **Object Manager** → Search for **Appointment**.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Lookup Relationship** as the Data Type → Click **Next**.
4. Select **Customer Details** as the related object → Click **Next**.
5. Complete the steps → Click **Save**.

2. Lookup Field on Service Records Object

1. Go to **Setup** → **Object Manager** → Search for **Service Records**.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Lookup Relationship** as the Data Type → Click **Next**.
4. Select **Appointment** as the related object → Click **Next**.
5. Make it a required field by selecting **Required**.
6. Add a **Lookup Filter**:
 - **Field**: Appointment: Appointment Date
 - **Operator**: Less Than
 - **Select Field**: Appointment: Created Date
 - **Filter Type**: Required
 - **Error Message**: Value does not match the criteria.
7. Enable the filter by checking **Active** → Click **Next** → **Save**.

3. Lookup Field on Billing Details and Feedback Object

1. Go to **Setup → Object Manager → Search for Billing Details and Feedback.**
2. Go to **Fields & Relationships → Click New.**
3. Select **Lookup Relationship** as the Data Type → Click **Next.**
4. Select **Service Records** as the related object → Click **Next.**
5. Complete the steps → Click **Save & New.**



FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Date	Appointment_Date__c	Date		
Appointment Name	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Customer Details	Customer_Details__c	Lookup(Customer Details)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Maintenance service	Maintenance_service__c	Checkbox		
Owner	OwnerId	Lookup(User Group)		✓
Repairs	Repairs__c	Checkbox		
Replacement Parts	Replacement_Parts__c	Checkbox		
Service Amount	Service_Amount__c	Currency(18, 0)		
Vehicle number plate	Vehicle_number_plate__c	Text(10) (Unique Case Insensitive)		✓

Creation of Checkbox Fields

1. Checkbox Fields on Appointment Object

1. Go to **Setup → Object Manager → Search for Appointment.**
2. Go to **Fields & Relationships → Click New.**
3. Select **Checkbox** as the Data Type → Click **Next.**

Create the following fields one at a time:

- **Maintenance Service:** Default Value - Unchecked
- **Repairs:** Default Value - Unchecked
- **Replacement Parts:** Default Value - Unchecked

2. Checkbox Field on Service Records Object

1. Follow Steps 1-2 above for **Service Records**.
2. Select **Checkbox** as the Data Type → Click **Next**.
3. Field Label: **Quality Check Status**
4. Default Value: **Unchecked**
5. Click **Next** → **Next** → **Save**.

The screenshot shows the Salesforce Setup interface. At the top, there's a navigation bar with 'Setup', 'Home', and 'Object Manager'. Below this, the breadcrumb trail reads 'SETUP > OBJECT MANAGER > Service records'. The main section is titled 'Fields & Relationships' with a subtitle '8 Items, Sorted by Field Label'. On the left, a sidebar lists various configuration options: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main table displays the following fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment	Appointment__c	Lookup(Appointment)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		✓
Quality Check Status	Quality_Check_Status__c	Checkbox		
service date	service_date__c	Formula (Date)		
Service records Name	Name	Auto Number		✓
Service Status	Service_Status__c	Picklist		

Creation of Date Fields

1. Date Field on Appointment Object

1. Go to **Setup** → **Object Manager** → Search for **Appointment**.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Date** as the Data Type → Click **Next**.
4. Field Label: **Appointment Date**
5. Make it a **Required** field.
6. Click **Next** → **Next** → **Save**.

Creation of Currency Fields

1. Currency Field on Appointment Object

1. Go to **Setup** → **Object Manager** → Search for **Appointment**.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Currency** as the Data Type → Click **Next**.
4. Field Label: **Service Amount**
5. Set **Read-Only** for all profiles at the field level.
6. Click **Next** → **Save**.

2. Currency Field on Billing Details and Feedback Object

1. Follow the steps above for **Billing Details and Feedback**.
2. Field Label: **Payment Paid**
3. Click **Save**.

The screenshot shows the Salesforce Setup interface. At the top, there's a navigation bar with 'Setup', 'Home', and 'Object Manager'. Below this, the 'Billing details and feedback' section is selected. The 'Fields & Relationships' tab is active, showing a list of 8 fields. The fields are sorted by Field Label. The table has columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The fields listed are: Billing details and feedbackName (Auto Number, indexed), Created By (Lookup(User)), Last Modified By (Lookup(User)), Owner (Lookup(User/Group), indexed), Payment Paid (Currency(18, 0), indexed), Payment Status (Picklist, indexed), Rating for service (Text(1), indexed), and Service records (Lookup(Service records), indexed).

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Billing details and feedbackName	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		✓
Payment Paid	Payment_Paid__c	Currency(18, 0)		✓
Payment Status	Payment_Status__c	Picklist		✓
Rating for service	Rating_for_service__c	Text(1)		✓
Service records	Service_records__c	Lookup(Service records)		✓

Creation of Text Fields

1. Text Field on Appointment Object

1. Go to **Setup → Object Manager → Search for Appointment.**
2. Go to **Fields & Relationships → Click New.**
3. Select **Text** as the Data Type → Click **Next.**
4. Field Label: **Vehicle License plate**
5. Length: **10**
6. Set field as **Required** and **Unique.**
7. Click **Next → Save.**

2. Text Field on Billing Details and Feedback Object

1. Follow Steps 1-2 for **Billing Details and Feedback.**
2. Field Label: **Rating for Service**
3. Length: **1**
4. Set field as **Required** and **Unique.**
5. Click **Save.**

The screenshot shows the Salesforce Setup interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar is present. The left sidebar shows a list of setup options, with 'Fields & Relationships' selected. The main content area is titled 'Billing details and feedback' and 'Rating for service'. It displays the 'Custom Field Definition Edit' page for a text field. The 'Field Information' section shows the field label 'Rating for service', field name 'Rating_for_service', and data type 'Text'. The 'General Options' section includes checkboxes for 'Required' (checked), 'Unique' (checked), and 'External ID' (unchecked). The 'Text Options' section shows the length set to 1. The 'Compliance Categorization' section shows a list of categories: PI, HIPAA, GDPR, and PCI, with 'Available' and 'Chosen' columns.

Creation of Picklist Fields

1. Picklist Field on Service Records Object

1. Go to **Setup** → **Object Manager** → Search for **Service Records**.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Picklist** as the Data Type → Click **Next**.
4. Field Label: **Service Status**
5. Enter Values:
 - Started
 - Completed
6. Click **Next** → **Save**.

2. Picklist Field on Billing Details and Feedback Object

1. Follow Steps 1-3 above for **Billing Details and Feedback**.
2. Field Label: **Payment Status**
3. Enter Values:
 - Pending
 - Completed
4. Click **Next** → **Save**.

The screenshot shows the Salesforce Setup interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar is present. The left sidebar shows the 'Setup' menu with 'Object Manager' selected. The main content area is titled 'Billing details and feedback' and 'Custom Field Definition Edit'. The 'Field Information' section shows the field label 'Payment Status', field name 'Payment_Status', and data type 'Picklist'. The 'General Options' section has a checkbox for 'Required' which is checked. The 'Picklist Options' section has a checkbox for 'Restrict picklist to the values defined in the value set' which is checked. The 'Available' list includes 'PII', 'HIPAA', 'GDPR', and 'PCI'. The 'Chosen' list is empty. The 'Save' button is visible at the bottom right.

Creation of Formula Fields

1. Formula Field on Service Records Object

1. Go to **Setup** → **Object Manager** → Search for **Service Records**.
2. Go to **Fields & Relationships** → Click **New**.
3. Select **Formula** as the Data Type → Click **Next**.
4. Field Label: **Service Date**
5. Formula Return Type: **Date**
6. Insert Formula: **CreatedDate**
7. Click **Check Syntax** → **Next** → **Save**.

The screenshot shows the Salesforce Setup interface for creating a new formula field. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The left sidebar shows the 'Fields & Relationships' section. The main content area is titled 'Custom Field Definition Edit' for the 'Service date' field on the 'Service records' object. The field is configured as a Date type formula field. The formula is 'Service date (Date) * CreatedDate'. The 'Available' list includes PII, HIPAA, GDPR, and PCI. The 'Chosen' list is empty. The 'Functions' list includes ABS, ACOS, ADDMONTHS, AND, ASCII, and ASIN.

Setup > OBJECT MANAGER
Service records

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

Custom Field Definition Edit

Field Information

Field Label: service date

Field Name: service_date

Description:

Help Text:

Data Owner: User

Field Usage: --None--

Data Sensitivity Level: --None--

Compliance Categorization

Available: PII, HIPAA, GDPR, PCI

Chosen:

Formula Options

Formula Return Type: Date

Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula subtab to use additional fields, operators, and functions.
Example: [Record.Date] + [CloseDate] - 7 [Blockquote]

Simple Formula | Advanced Formula

Insert Field

Service date (Date) * CreatedDate

Insert Operator

Functions

All Function Categories

ABS

ACOS

ADDMONTHS

AND

ASCII

ASIN

Insert Selected Function

Quick Tips: Switch Standard Operations & Functions

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. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

To Create a Validation Rule for the Appointment Object

1. Go to the **Setup** page → Click on **Object Manager**.
2. From the drop-down, click **Edit** for the **Appointment** object.
3. Click on **Validation Rules** → Click **New**.
4. Enter the Rule Name as "**Vehicle**".
5. Insert the **Error Condition Formula**:
6. NOT(REGEX(Vehicle_number_plate__c, "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))
7. Enter the **Error Message** as "**Please enter a valid number**".
8. Set the **Error Location** to **Field** and select the field **Vehicle License plate**.
9. Click **Save**.

The screenshot displays the Salesforce Setup interface for configuring a validation rule on the Appointment object. The left sidebar shows the navigation menu with 'Setup' and 'Object Manager' selected. The main content area is titled 'Appointment Validation Rule' and includes a 'Validation Rule Edit' section. The 'Rule Name' is set to 'Vehicle', and the 'Active' checkbox is checked. The 'Error Condition Formula' is entered as 'NOT(REGEX(Vehicle_number_plate__c, "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))'. The 'Error Message' is 'Please enter valid number', and the 'Error Location' is set to 'Field' with 'Vehicle number plate' selected. The 'Save' button is visible at the bottom right.

Setup > OBJECT MANAGER
Appointment

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Conditional Field Formatting

Appointment Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit

Rule Name: **Vehicle**

Active: ☒

Description:

Error Condition Formula

Example: `Discount_Percent__c > 30`
Display an error if Discount is more than 30%
If this formula expression is true, display the text defined in the Error Message area

Insert Field Insert Operator

NOT(REGEX(Vehicle_number_plate__c, "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))

Check Syntax

Functions

-- All Function Categories --
ABS
ACOS
ADDMONTHS
AND
ASCII
ASIN

Insert Selected Function
ABS(number)
Returns the absolute value of a number, a number without its sign
Click on this function

Error Message

Example: `Discount percent cannot exceed 30%`
This message will appear when Error Condition formula is true

Error Message: **Please enter valid number**

This error message can either appear at the top of the page or below a specific field on the page

Error Location: ☐ Top of Page ☒ Field **Vehicle number plate**

Save Save & New Cancel

To Create a Validation Rule for the Service Records Object

1. Go to the **Setup** page → Click on **Object Manager**.
2. From the drop-down, click **Edit** for the **Service Records** object.
3. Click on **Validation Rules** → Click **New**.
4. Enter the Rule Name as "**service_status_note**".
5. Insert the **Error Condition Formula**:
6. NOT(ISPICKVAL(Service_Status__c, "Completed"))
7. Enter the **Error Message** as "**Still it is pending**".
8. Set the **Error Location** to **Field** and select the field **Service Status**.
9. Click **Save**.

The screenshot displays the Salesforce Setup interface for creating a validation rule on the 'Service records' object. The left sidebar shows the navigation menu with 'Validation Rules' selected. The main content area is titled 'Service records Validation Rule' and includes a description of validation rules. The 'Validation Rule Edit' section contains the following fields:

- Rule Name:** service_status_note
- Active:** ☒
- Description:** (empty field)

The **Error Condition Formula** section shows the formula: `NOT (ISPICKVAL (Service_Status__c, "Completed"))`. A 'Functions' list on the right includes ABS, ACOS, ADDMONTHS, AND, ASCII, and ASIN. The **Error Message** section shows the message: `Still it is pending`. The **Error Location** is set to **Field** and the field **Service Status** is selected. The bottom of the page has 'Save', 'Save & New', and 'Cancel' buttons.

To Create a Validation Rule for the Billing Details and Feedback Object

1. Go to the **Setup** page → Click on **Object Manager**.
2. From the drop-down, click **Edit** for the **Billing Details and Feedback** object.
3. Click on **Validation Rules** → Click **New**.
4. Enter the Rule Name as "**rating_should_be_less_than_5**".
5. Insert the **Error Condition Formula**:
6. NOT(REGEX(Rating_for_service__c, "[1-5]{1}"))
7. Enter the **Error Message** as "**Rating should be from 1 to 5**".
8. Set the **Error Location** to **Field** and select the field **Rating for Service**.
9. Click **Save**.

The screenshot shows the Salesforce Setup interface for creating a validation rule. The left sidebar contains a navigation menu with options like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main content area is titled 'Billing details and feedback Validation Rule' and includes a 'Validation Rule Edit' section with fields for Rule Name (rating_should_be_less_than_5), Active (checked), and Description. Below this is the 'Error Condition Formula' section, which contains the formula NOT(REGEX(Rating_for_service__c, "[1-5]{1}")) and a 'Functions' dropdown menu. The 'Error Message' section shows the message 'Rating should be from 1 to 5' and the 'Error Location' set to 'Field' with 'Rating for Service' selected. The bottom of the page has 'Save', 'Save & New', and 'Cancel' buttons.

Duplicate Rules and Matching Rules

To Create a Matching Rule for the Customer Details Object

1. Go to the **Quick Find** box in **Setup** and search for **Matching Rule**.
2. Click on **Matching Rule** → Click **New Rule**.
3. Select the object as **Customer Details** → Click **Next**.
4. Enter the **Rule Name: Matching Customer Details**.
5. The **Unique Name** is auto-populated.

Field	Matching Method
Gmail	Exact
Phone Number	Exact

6. Define the matching criteria as above.
7. Click **Save**.
8. After saving, click on **Activate**.

The screenshot displays the Salesforce Setup interface. On the left, the navigation menu includes 'Setup', 'Home', and 'Object Manager'. The 'Setup' page is active, showing a search bar and a list of items. The 'Matching Rules' section is highlighted. The main content area shows the 'Matching Rules' configuration for the 'Customer Details' object. The rule name is 'Matching customer details', and the unique name is 'Matching_customer_details'. The matching criteria are defined as: (Customer Details: Gmail EXACT MatchBANK = FALSE) AND (Customer Details: Phone_number EXACT MatchBANK = FALSE). The rule is currently 'Active' and was created by 'AMAN SINGH' on 08/12/2024 at 1:37 am. The 'Matching Rule Detail' section shows the rule's configuration, including the object, rule name, unique name, description, matching criteria, status, and creation/modification details.

Field	Matching Method
Gmail	Exact
Phone Number	Exact

Matching Rule Detail

Field	Value
Object	Customer Details
Rule Name	Matching customer details
Unique Name	Matching_customer_details
Description	
Matching Criteria	(Customer Details: Gmail EXACT MatchBANK = FALSE) AND (Customer Details: Phone_number EXACT MatchBANK = FALSE)
Status	Active
Created By	AMAN SINGH 08/12/2024 1:37 am
Modified By	AMAN SINGH 08/12/2024 1:38 am

To Create a Duplicate Rule for the Customer Details Object

1. Go to the **Quick Find** box in **Setup** and search for **Duplicate Rules**.
2. Click on **Duplicate Rule** → Click **New Rule**.
3. Select the **Customer Details** object.
4. Enter the **Rule Name** as **Customer Detail Duplicate**.
5. Scroll to the **Matching Rule** section.
6. Select the matching rule: **Matching Customer Details**.
7. Click **Save**.
8. After saving, click on **Activate**.

The screenshot shows the Salesforce Duplicate Rules configuration interface. The left sidebar contains the navigation menu with 'Setup', 'Home', and 'Object Manager'. The main content area is titled 'Duplicate Rules' and shows the configuration for a rule named 'Customer Detail duplicate'. The rule is currently in 'Edit' mode, with 'Save', 'Save & New', and 'Cancel' buttons at the top.

Rule Details

- Rule Name:** Customer Detail duplicate
- Description:**
- Object:** Customer Details
- Record-Level Security:** Enforce sharing rules (selected), Bypass sharing rules

Actions

Specify what happens when a user tries to save a duplicate record.

- Action On Create:** Allow (selected), Alert (selected), Report
- Action On Edit:** Allow (selected), 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): Small EXACT Match(Bank = FALSE) AND (Customer: Details): Phone_Housew EXACT Match(Bank = FALSE)
- Field Mapping:** Matched Selected

Conditions

Optionally, specify the conditions a record must meet for the rule to run.

Field	Operator	Value	
None	None		AND
None	None		AND
None	None		AND
None	None		AND
None	None		AND

Add Filter Logic...

At the bottom, there are 'Save', 'Save & New', and 'Cancel' buttons.

Profiles

A profile is a group/collection of settings and permissions that define what a user can do in Salesforce. Profiles control **object permissions, field permissions, user permissions, tab settings, app settings, Apex class access, sales force page access, page layouts, record types, login hours, and login IP ranges.**

To Create a Manager Profile

1. Go to **Setup** → Type **Profiles** in the **Quick Find** box.
2. Click on **Profiles** → Clone the desired profile (**Standard User**).
3. Enter the **Profile Name: Manager** → Click **Save**.
4. Click **Edit** on the profile page.
5. Set the **Custom App Settings** default to **Garage Management**.
6. Scroll to **Custom Object Permissions** and grant access for:
 - **Appointments**
 - **Billing Details and Feedback**
 - **Service Records**
 - **Customer Details**
7. Change the session timeout to **8 hours of inactivity**.
8. Update **Password Policies**:
 - User passwords expire: **Never expires**.
 - Minimum password length: **8**.
9. Click **Save**.

The screenshot displays the Salesforce Setup interface for the 'Manager' profile. The page is titled 'Profiles' and 'Manager'. It includes a search bar at the top with the text 'Search Setup'. The left sidebar shows the 'Setup' menu with 'Profiles' selected. The main content area is divided into several sections:

- Profile Edit:** Includes fields for 'Name' (Manager), 'User License' (Developer), and 'Description'. There are buttons for 'Save', 'Save & New', and 'Cancel'. A 'Custom Profile' checkbox is checked.
- Custom App Settings:** A table with columns 'Visible' and 'Default' for various Salesforce applications. The 'Default' column is set to 'Garage Management' for all applications.
- Service Provider Access:** A section for configuring service provider access.
- Tab Settings:** A section for configuring tab settings, including 'Home', 'Accounts', 'All Sites', 'Alternative Payment Methods', 'Analytics', 'App Launcher', and 'Appointment Categories'. Each tab has a 'Default On' or 'Default Off' checkbox.

Setup Home Object Manager

Search Setup

profiles

Users

Profiles

Didn't find what you're looking for? Try using Global Search.

Profiles

Object	Read	Create	Edit	Delete	View All	Modify All
Expression Set Step Relationships	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Expression Set Versions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Finance Balance Snapshots	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Finance Transactions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Fulfillment Orders	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gateway Provider Payment Method Types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Guest User Anomaly Event Stores	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Identities	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Images	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Incidents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Individuals	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Inventory Reservations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Investors	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Object	Read	Create	Edit	Delete	View All	Modify All
Store Buyer Groups	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Store Catalogs	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Streaming Channels	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
User External Credentials	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Waitlists	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Web Card Documents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Web Store Inventory Sources	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Work Orders	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Work Plans	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Work Plan Templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Work Step Templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Work Types	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Work Type Groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Custom Object Permissions

Object	Object Access				Object Administration	
	Read	Create	Edit	Delete	View All	Modify All
Appointments	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Billing details and feedback	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Customer Details	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Session Settings

Session Times Out After:

Session Security Level Required at Login:

Password Policies

User passwords expire in:

Enforce password history:

Minimum password length:

Password complexity requirement:

Password question requirement:

Maximum invalid login attempts:

Lockout effective period:

Obscure secret answer for password resets: ☐

Require a minimum 1 day password lifetime: ☐

Don't immediately expire links in forgot password emails: ☐

To Create a Sales Person Profile

1. Go to **Setup** → Type **Profiles** in the **Quick Find** box.
2. Click on **Profiles** → Clone the desired profile (**Salesforce Platform User**).
3. Enter the **Profile Name: Sales Person** → Click **Save**.
4. Click **Edit** on the profile page.
5. Set the **Custom App Settings** default to **Garage Management**.
6. Scroll to **Custom Object Permissions** and grant access for:
 - **Appointments**
 - **Billing Details and Feedback**
 - **Service Records**
 - **Customer Details**
7. Click **Save**.

Setup

Home

Object Manager

Search Setup

Q profiles

Users

Profiles

Didn't find what you're looking for? Try using Global Search.

SETUP

Profiles

Profile Edit

sales person

Set the permissions and page layouts for this profile.

Save Save & New Cancel

Name

sales person

User License

Salesforce Platform

Description

Custom Profile

✓

Custom App Settings

Visible

Default

Analytics Studio (standard__insights)

☐

☐

App Launcher (standard__AppLauncher)

☐

☐

Garage Management Application (Garage_Management_Application)

☐

☒

How We Roll Maintenance (How_We_Roll_Maintenance)

☐

☐

Platform (standard__Platform)

☒

☐

WEC (standard__Work)

☐

☐

Service Provider Access

Tab Settings

☐ Overwrite user's personal tab customizations

Standard Tab Settings

Home

Default On

Accounts

Default On

All Sites

Tab Hidden

Alternative Payment Methods

Tab Hidden

Analytics

Default On

Analytics

Default On

App Launcher

Default On

Appointment Categories

Default Off

Appointment Invitations

Default Off

Approval Requests

Default On

Assets

Tab Hidden

Async Operation Logs

Default On

Authorization Form

Default On

Authorization Form Consent

Default On

Authorization Form Data Use

Default On

Authorization Form Text

Default On

Awards

Tab Hidden

Background Operations

Tab Hidden

Business Brands

Tab Hidden

Calculation Procedures

Tab Hidden

Inventory Item Reservations

Tab Hidden

Inventory Reservations

Tab Hidden

Invoices

Tab Hidden

Labels

Default On

Leads

Tab Hidden

Libraries

Default On

Lightning Bulk Solutions

Default On

Lightning Usage

Default On

List Emails

Tab Hidden

Location Groups

Default On

Locations

Default On

Location Shipping Carrier Methods

Tab Hidden

Lookup Tables

Default On

Mobile Home

Default On

Omnichannel Inventory

Tab Hidden

Operating Hours

Tab Hidden

Orchestration Runs

Default On

Orchestrations

Default On

Orchestration Work Items

Default On

Order Product Receipts

Tab Hidden

Setup

Home

Object Manager

Search Setup

Q profiles

Users

Profiles

Didn't find what you're looking for? Try using Global Search.

SETUP

Profiles

Custom Object Permissions

	Read	Create	Edit	Delete	View All	Modify All
Appointments	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Billing details and feedback	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Customer Details	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Session Settings

Session Times Out After

2 hours of inactivity

Session Security Level Required at Login

None

Password Policies

User passwords expire in

90 days

Enforce password history

3 passwords remembered

Minimum password length

8

Password complexity requirement

Must include alpha and numeric characters

Password question requirement

Cannot contain password

Maximum invalid login attempts

10

Lockout effective period

15 minutes

Require a secret answer for password resets

☐

Require a minimum 1 day password lifetime

☐

Don't immediately expire links to forgot password emails

☐

Save Save & New Cancel

Roles and Role Hierarchy

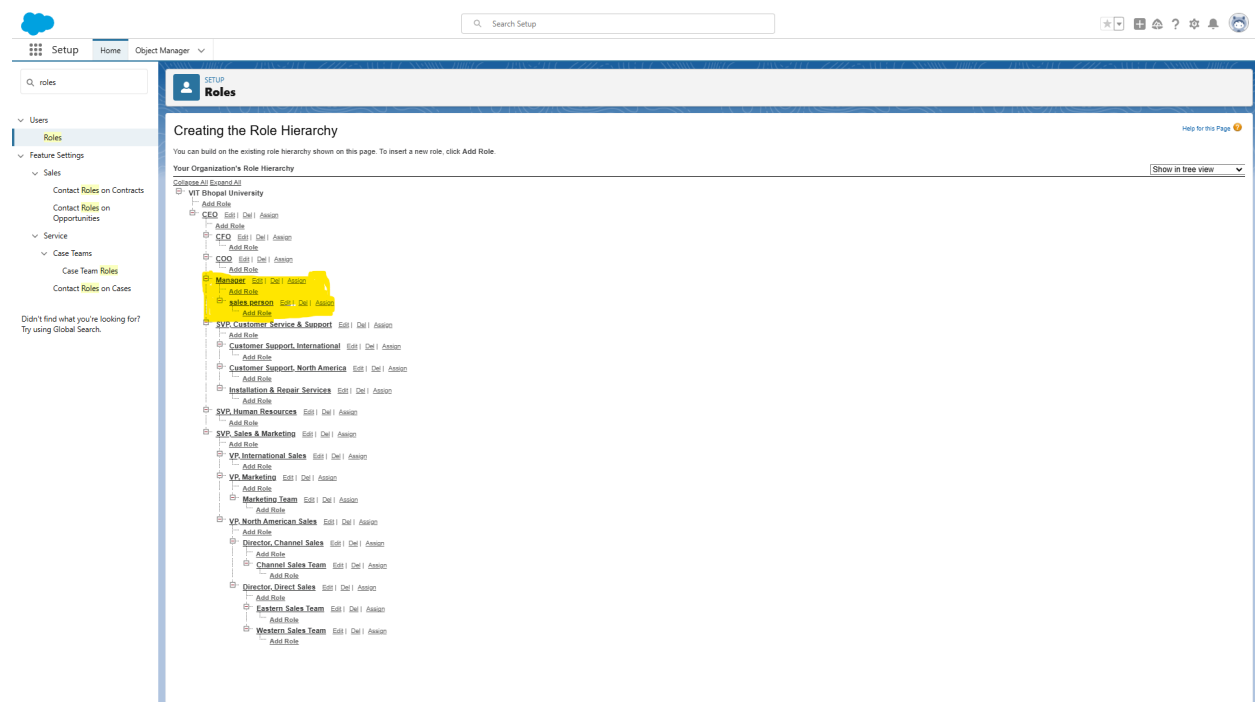
A **Role** in Salesforce defines a user's record-level visibility access. Roles specify the types of access that users have to records.

To Create the Manager Role

1. Go to **Quick Find** → Search for **Roles** → Click on **Set Up Roles**.
2. Click on **Expand All** → Add a role under the desired role.
3. Enter the **Label: Manager**. The **Role Name** auto-populates.
4. Click **Save**.

To Create the Sales Person Role

1. Go to **Quick Find** → Search for **Roles** → Click on **Set Up Roles**.
2. Click on the **plus** next to the CEO role.
3. Add a role under **Manager**.
4. Enter the **Label: Sales Person**. The **Role Name** auto-populates.
5. Click **Save**.



Users

A **User** is anyone who logs into Salesforce. Users represent employees who need access to the company's records.

To Create a User

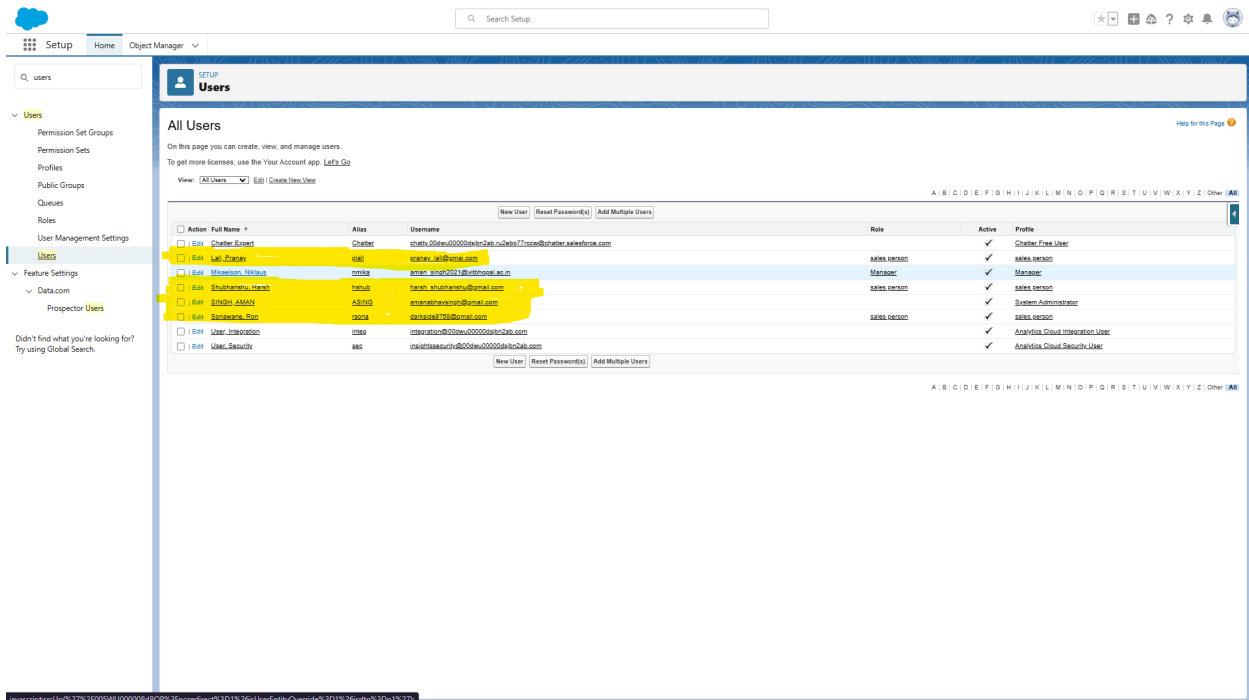
1. Go to **Setup** → Type **Users** in the **Quick Find** box.
2. Select **Users** → Click **New User**.
3. Fill in the fields:
 - **First Name:** Nicklaus
 - **Last Name:** Mikaelson
 - **Alias:** (Enter Alias Name)
 - **Email ID:** (Personal Email ID)
 - **Username:** (Format: text@text.text)
 - **Nickname:** (Enter Nickname)
 - **Role:** Manager
 - **User License:** Salesforce
 - **Profile:** Manager
4. Click **Save**.

The screenshot shows the Salesforce 'User Edit' interface for a user named 'Niklaus Mikaelson'. The page is divided into several sections for user configuration:

- General Information:** Fields for First Name (Nicklaus), Last Name (Mikaelson), Alias (nmika), Email (aman_singh2021@gmail.com), Username (aman_singh2021@gmail.com), Nickname (User773360321999385996), Title, Company, Department, and Division.
- Role and License:** Role is set to 'Manager', User License is 'Salesforce', and Profile is 'Manager'.
- Active Status:** The 'Active' checkbox is checked.
- User Types:** A list of checkboxes for various user types: Marketing User, Offline User, Knowledge User, Flow User, Service Cloud User, Site.com Contributor User, Site.com Publisher User, and WDC User. 'Marketing User' is checked.
- Advanced Settings:** Includes 'Data.com User Type' (set to '-None-'), 'Data.com Monthly Addition Limit' (300), 'Accessibility Mode (Classic Only)' (checked), 'High-Contrast Palette on Charts' (checked), 'Load Lightning Pages While Scrolling' (checked), 'Debug Mode' (checked), and 'Make Setup My Default Landing Page' (unchecked).
- Notifications:** Checkboxes for 'Salesforce CRM Content User', 'Receive Salesforce CRM Content Email Alerts', 'Receive Salesforce CRM Content Alerts as Daily Digest', and 'Allow Forecasting'. The first three are checked.
- Phone Information:** Fields for Call Center, Phone (00007665797), Extension, Fax, and Mobile (with an example number +1 5035550123).
- Email Encoding:** Set to 'Unicode (UTF-8)'.
- Employee Number:** A field for the employee's unique identifier.
- Mailing Address:** A section for the user's physical address, with 'Street' and 'Postward Road' fields visible.

To Create Another User

1. Repeat the steps above for:
 - **Role:** Sales Person
 - **User License:** Salesforce Platform
 - **Profile:** Sales Person
2. Create at least **three users** with these permissions.



Public Groups

Public groups are an essential tool for Salesforce administrators and developers, helping streamline user management, data access, and security settings. By creating and utilizing public groups effectively, you can ensure a secure and organized Salesforce environment, providing users with appropriate access to necessary resources.

Creating a New Public Group

1. Navigate to **Setup** → Search for **Users** in the Quick Find box → Select **Public Groups** → Click **New**.
2. Assign a label such as **Sales Team**.
3. The group name will be auto-filled based on the label.
4. Search for the relevant **Roles**.
5. From the **Available Members** list, select **Salesperson**, then click **Add** to move them to **Selected Members**.
6. Save the changes.

The screenshot shows the Salesforce Setup interface for creating a new Public Group. The left sidebar contains the navigation menu with 'Setup' selected. The main content area is titled 'Public Groups' and shows the 'Edit Public Group' form. The form includes fields for 'Label' (Sales Team), 'Group Name' (Sales Team), and 'Description'. Below these fields is a search bar for 'Public Groups'. The 'Available Members' list on the left shows 'Group: Lead Routing' and 'Role: sales person'. The 'Selected Members' list on the right is empty. The 'Add' button is visible between the two lists. The 'Save' and 'Cancel' buttons are at the bottom of the form.

Sharing Settings

Sharing settings in Salesforce govern how records are accessed and shared across the organization, ensuring data privacy and security. These configurations allow administrators to define specific access rules tailored to organizational needs:

- **Organization-Wide Default (OWD) Settings:**

These settings determine the baseline access level for all objects in Salesforce. Options include:

- **Private:** Only record owners and those explicitly granted access can view/edit the records.
- **Public Read-Only:** All users can view, but only owners can edit.
- **Public Read/Write:** All users can view and edit.
- **Controlled by Parent:** Access is inherited from the parent object.

- **Role Hierarchy:**

Role hierarchies enable record access based on user roles. Users higher in the hierarchy can view or manage records owned by users below them. This approach complements OWD settings by refining access permissions.

- **Profiles and Permission Sets:**

- **Profiles:** Assign object and field-level permissions to user groups.
- **Permission Sets:** Extend additional permissions to specific users as needed.

Configuring Sharing Settings

1. Go to **Setup** → Search for **Users** in the Quick Find box → Select **Sharing Settings** → Click **Edit**.
2. Set the **OWD setting** for the **Service Records** object to **Private**.
3. Save the changes and refresh the page.
4. Scroll down and click **New** under **Service Records Sharing Rules**.
5. Provide a label, such as **Sharing Setting**. The rule name will auto-fill based on the label.
6. **Step 3:** Specify which records to share: Select **Roles** → **Salesperson**.
7. **Step 4:** Indicate with whom to share: Choose **Roles** → **Manager**.
8. **Step 5:** Set the access level to **Read/Write**.
9. Save the sharing rule.

Setup

Home

Object Manager

Search Setup

Q, sharing

Security

Guest User

Sharing

Rule Access

Report

Sharing Settings

Didn't find what you're looking for?

Try using Global Search.

Role Edit

sales person

Help for this Page

Role Edit

Label

sales person

Role Name

sales_person

This role reports to

Manager

Role Name as displayed on reports

Save

Save & New

Cancel

Flows

Flows in Salesforce

Flows are powerful automation tools in Salesforce that help administrators create seamless processes for managing data and performing actions within the system. Below is a guide to creating a flow:

Steps to Create a Flow

1. Navigate to Setup:

- Go to **Setup** → Search for **Flow** in the Quick Find box → Click on **Flow** → Select **New Flow**.

2. Choose Flow Type:

- Select **Record-Triggered Flow** → Click **Create**.

3. Configure Object and Trigger:

- Choose the object as **Billing Details and Feedback** from the dropdown.
- Set the trigger condition to **A Record is Created or Updated**.
- Optimize the flow for **Actions and Related Records** → Click **Done**.

4. Add an Update Records Element:

- Under **Record-Triggered Flow**, click the “+” symbol → Select **Update Records** from the dropdown.
- **Label Name:** Enter **Amount Update**.
- **API Name:** Auto-filled.
- **Filter Condition:** Set as **All Conditions are Met (AND)**.
 - **Field:** Payment_Status__c
 - **Operator:** Equals
 - **Value:** Completed
- Set the field values for the **Billing Details and Feedback Record**:
 - **Field:** Payment_Paid__c
 - **Value:** {!\$Record.Service_records__r.Appointment__r.Service_Amount__c}
- Click **Done**.

5. Create a New Resource:

- From the toolbox, click **New Resource** → Select **Variable**.
- Choose **Text Template** as the resource type.
- **API Name:** Enter **alert**.
- Change the view to **Plain Text**.
- In the body field, paste the following text:

Dear

`{!$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},`

I hope this message finds you well. I wanted to take a moment to express my sincere gratitude for your recent payment for the services provided by our garage management team. Your prompt payment is greatly appreciated, and it helps us continue to provide top-notch services to you and all our valued customers.

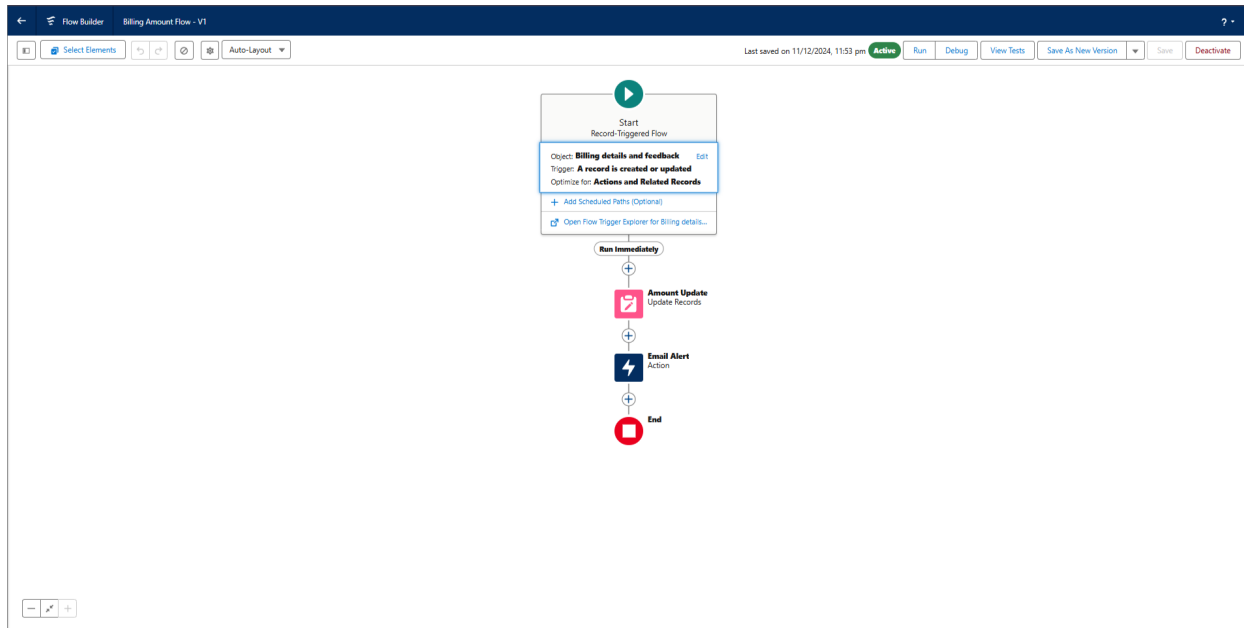
Amount Paid: `{!$Record.Payment_Paid__c}`
Thank you for coming.
- Click **Done**.

6. Add an Action Element:

- Click **Add Element** → Select **Action**.
- In the action search bar, type **Send Email** → Click on it.
- **Label Name:** Enter **Email Alert**.
- **API Name:** Auto-filled.
- Set input values:
 - **Body:** Select the previously created text template (**alert**).
 - **Recipient Address List:**
`{!$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c}`
 - **Subject:** Enter **Thank You for Your Payment - Garage Management**.
- Click **Done**.

7. Save and Activate the Flow:

- Save the flow with a suitable **Flow Label**.
- The **Flow API Name** will auto-populate.
- Click **Save** → Click **Activate**.



Apex Triggers

Apex triggers in Salesforce allow developers to execute custom actions in response to changes in Salesforce records. These triggers can be defined to execute before or after specific database operations, enabling greater control and flexibility in record handling.

Supported Trigger Operations

Triggers can run **before** or **after** the following events:

- **Insert**
- **Update**
- **Delete**
- **Merge**
- **Upsert**
- **Undelete**

For instance, triggers can be used to:

- Validate data before insertion or update.
- Perform actions after a record is deleted or restored.
- Automate changes in related records when an operation occurs.

Triggers can be defined for:

- **Top-Level Standard Objects** (e.g., Account, Contact).
- **Standard Child Objects** (e.g., CaseComment).
- **Custom Objects.**

To define a trigger, navigate to the **Object Management Settings** for the relevant object → Select **Triggers**.

Types of Apex Triggers

1. **Before Triggers:**
 - Used to validate or update records before they are saved to the database.
 - Common use: Ensuring data meets specific criteria before committing changes.
2. **After Triggers:**
 - Used to make changes to other records or access system-set field values after the record is saved.
 - Common use: Modifying related records based on changes to the current record.

Apex Handler Use Case: Amount Distribution

This example demonstrates how to distribute the service amount for each service the customer selects for their vehicle.

Steps to Implement

1. **Access Developer Console:**
 - Log in to your Salesforce Trailhead account.
 - Click on the **Gear Icon** (top-right corner) → Select **Developer Console**.
2. **Create a New Apex Class:**
 - In the Developer Console, click **File** → **New** → **Apex Class**.
 - Name the class **AmountDistributionHandler**.

```
Developer Console - Google Chrome
vitbhopaluniversity47-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

AmountDistribution.apex | AmountDistributionHandler.apex *
Code Coverage: None | API Version: 62

1 public class AmountDistributionHandler {
2
3
4
5 public static void amountDist(list<Appointment__c> listApp){
6
7     list<Service_records__c> serList = new list<Service_records__c>();
8
9
10
11     for(Appointment__c app : listApp){
12
13         if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
14
15             app.Service_Amount__c = 10000;
16
17         }
18
19         else if(app.Maintenance_service__c == true && app.Repairs__c == true){
20
21             app.Service_Amount__c = 5000;
22
23         }
24
25         else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
26
27             app.Service_Amount__c = 8000;
28
29         }
30
31         else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
32
33             app.Service_Amount__c = 7000;
34
35         }
36     }
37 }
```

```
Developer Console - Google Chrome
vitbhopaluniversity47-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

AmountDistribution.apex | AmountDistributionHandler.apex *
Code Coverage: None | API Version: 62

27         app.Service_Amount__c = 8000;
28     }
29 }
30
31 else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
32
33     app.Service_Amount__c = 7000;
34
35 }
36
37 else if(app.Maintenance_service__c == true){
38
39     app.Service_Amount__c = 2000;
40
41 }
42
43 else if(app.Repairs__c == true){
44
45     app.Service_Amount__c = 3000;
46
47 }
48
49 else if(app.Replacement_Parts__c == true){
50
51     app.Service_Amount__c = 5000;
52
53 }
54
55 }
56
57 }
58
59 }
60
61 }
```

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;
            }

        }

    }
}
```


Creating a New Trigger in Salesforce

Triggers are a vital part of Salesforce development, allowing custom logic to execute automatically when certain actions occur on an object. Below is a step-by-step guide for creating a trigger in Salesforce and a use case example.

Steps to Create a Trigger

1. Open Developer Console:

- Log in to your Salesforce Trailhead account.
- Click the **Gear Icon** in the top-right corner → Select **Developer Console**.

2. Create a New Trigger:

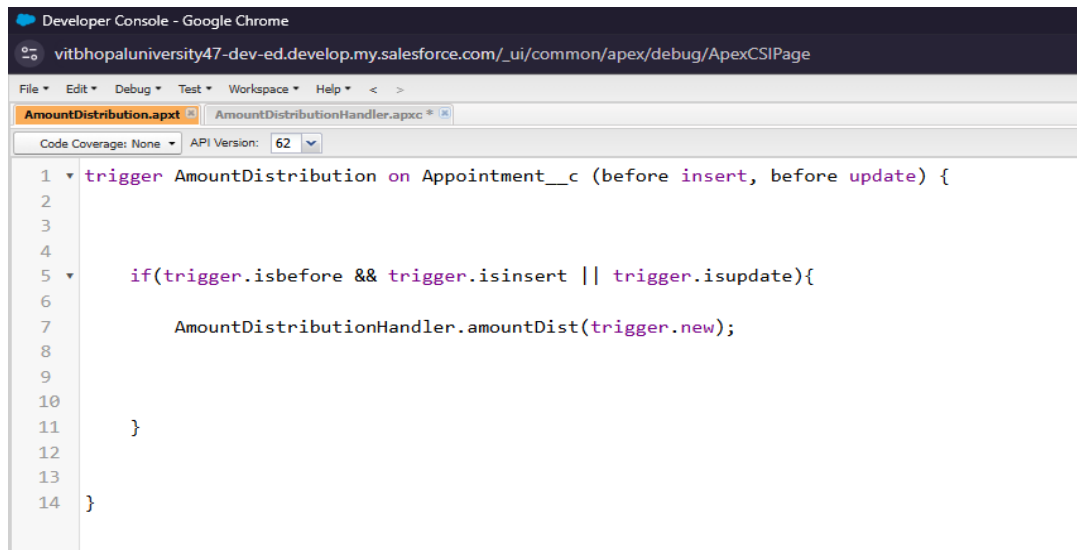
- In the Developer Console, go to **File** → **New** → **Trigger**.
- Provide a **Trigger Name** and select the **Object** it will act upon.
- Example:
 - **Trigger Name:**AmountDistribution
 - **Object:**Appointment__c

3. Basic Trigger Syntax:

- The syntax for creating a trigger:
- ```
trigger [TriggerName] on [ObjectName] (Before/After Events)
{
 // Trigger Logic
}
```

#### 4. Use Case Overview:

- The trigger will execute when a record's total exceeds a defined threshold (minimum business requirement).
- When this condition is met, the trigger logic runs to perform necessary actions.



```
1 trigger AmountDistribution on Appointment__c (before insert, before update) {
2
3
4
5 if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
6
7 AmountDistributionHandler.amountDist(trigger.new);
8
9
10 }
11
12
13
14 }
```

### **CODE:**

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

 if(trigger.isbefore && trigger.isinsert || trigger.isupdate){

 AmountDistributionHandler.amountDist(trigger.new);

 }

}
```

## **Report**

Salesforce Reports enable users to analyze, visualize, and share data in meaningful ways. They offer various formats and features to suit diverse needs. This guide walks you through report creation, sharing, and customization basics.

### **Types of Reports in Salesforce**

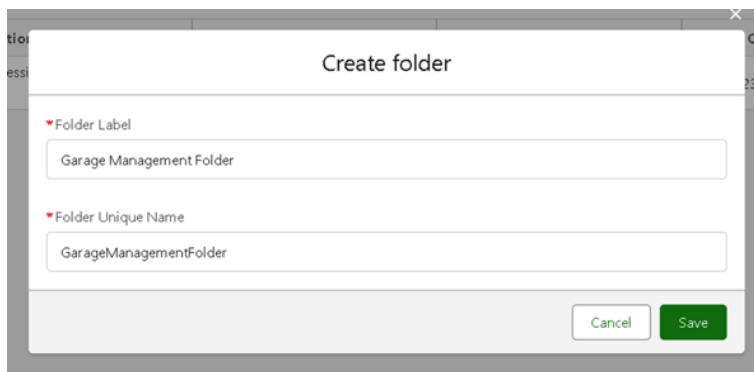
#### **1. Tabular Reports:**

- Present data in a simple table format.

- Best suited for creating lists.
- 2. **Summary Reports:**
  - Provide grouped data with subtotals.
  - Useful for analysis with categorization.
- 3. **Matrix Reports:**
  - Compare data in rows and columns.
  - Ideal for detailed analysis with multiple dimensions.
- 4. **Joined Reports:**
  - Combine multiple report types into one view.
  - Great for summarizing related information.

## Steps to Create a Report Folder

1. **Open the Reports Tab:**
  - Click on the **App Launcher** and search for **Reports**.
  - Select the **Reports Tab**.
2. **Create a New Folder:**
  - Click on **New Folder**.
  - Enter the folder name as **Garage Management Folder**.
  - The unique folder name will auto-populate.
3. **Save the Folder:**
  - Click **Save** to finalize.



The screenshot shows a 'Create folder' dialog box. It has two input fields: 'Folder Label' containing 'Garage Management Folder' and 'Folder Unique Name' containing 'GarageManagementFolder'. At the bottom right, there are 'Cancel' and 'Save' buttons.

## Sharing a Report Folder

1. **Locate the Folder to Share:**
  - Go to the **Reports Tab** → Click on **All Folders**.
  - Find **Garage Management Folder** and click the dropdown arrow.
2. **Set Sharing Options:**
  - Select **Share** from the dropdown menu.
  - In the **Share With** field, choose **Roles**.
  - Enter the role name, such as **Manager**.
  - Set the access level to **View**.
3. **Complete Sharing:**
  - Click **Share** and then **Done**.

## Creating a Custom Report Type

1. **Access Report Type Settings:**
  - Go to **Setup** → Type **Report Type** in the Quick Find box → Select **Report Types** → Click **Continue**.
2. **Define a New Report Type:**
  - Click on **New Custom Report Type**.
  - Select the **Primary Object** (e.g., *Customer Details*).
  - Provide the following:
    - **Report Type Label:** *Service Information*
    - **Report Type Name:** Auto-populated
    - **Description:** Keep default or provide relevant details.
    - **Store in Category:** Select **Other Reports**.
    - **Deployment Status:** Set to **Deployed**.
3. **Relate Additional Objects:**
  - Click the **Related Object** box → Select **Appointment Object**.
  - Repeat the process to add related objects:
    - **Service Records**
    - **Billing Details and Feedback**
4. **Save the Report Type:**
  - Click **Save** to finalize.

## Edit Custom Report Type Service information



Changes you make to this report type will affect existing reports created from it.  
We recommend that you clone any report type you wish to edit, rather than change a working report type.

### Object Relationships

Save Cancel

This report type will generate reports about Customer Details. You may define which related records from other objects are returned in report results by choosing a relationship to another object.

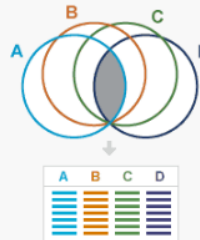
**A Customer Details**  
Primary Object

**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.

**C Service records**  
B to C Relationship:  
☒ Each "B" record must have at least one related "C" record.  
☐ "B" records may or may not have related "C" records.

**D Billing details and feedback**  
C to D Relationship:  
☒ Each "C" record must have at least one related "D" record.  
☐ "C" records may or may not have related "D" records.

Object Limit Reached  
You can associate up to four objects to a custom report type.



Save Cancel

## Steps to Create a Report

### Preparation:

Before creating the report, ensure that at least **10 records** are created for each object, with every field populated for optimal results.

#### 1. Navigate to Reports Tab:

- Open the **App Launcher**, search for **Reports**, and click on the tab.

#### 2. Start a New Report:

- Click on **New Report**.
- Choose the **Category** as **Other Reports**, search for **Service Information**, and select it.
- Click **Start Report**.

#### 3. Select Report Fields:

- In the **Outline Pane**, add the following fields to the **Column Section**:
  - **Customer Name**
  - **Appointment Date**
  - **Service Status**
  - **Payment Paid**
- Remove any unnecessary fields.

#### 4. Group Rows:

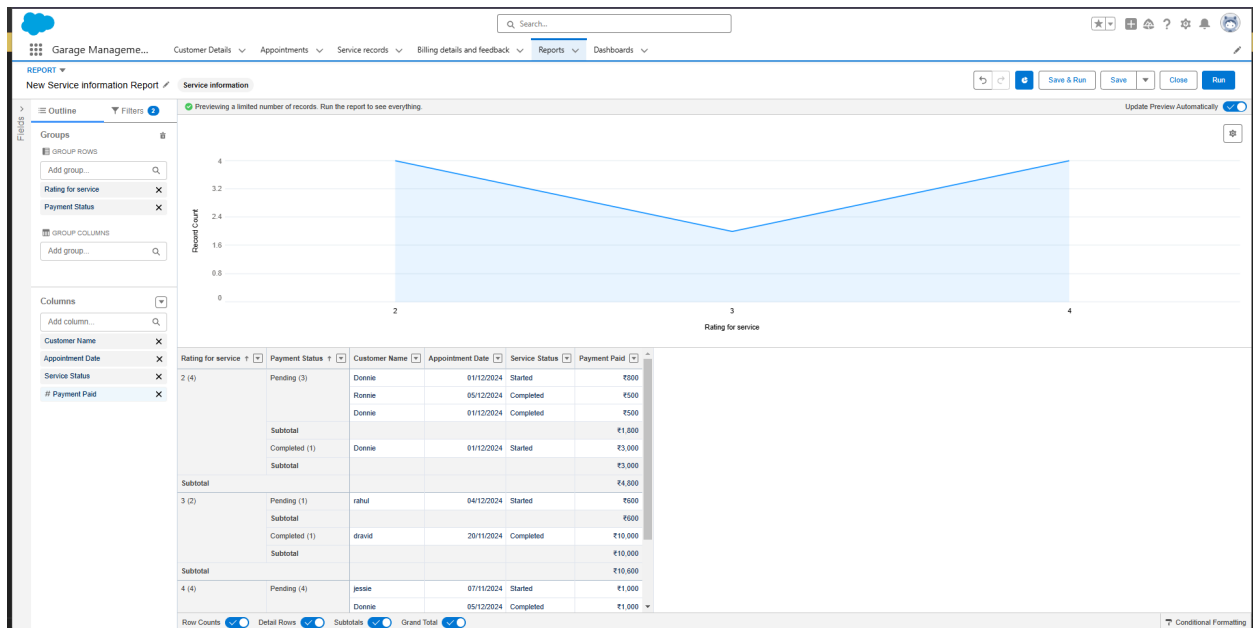
- In the **Group Rows Section**, add:
  - **Rating for Service**
  - **Payment Status**

#### 5. Add a Chart:

- Click **Add Chart** and select the **Line Chart**.

#### 6. Save the Report:

- Provide the report name as **New Service Information Report**.
- The unique name will auto-populate.
- Select the previously created folder and click **Save**.



## Dashboard

Dashboards provide a visual overview of business data and trends, empowering users to make data-driven decisions.

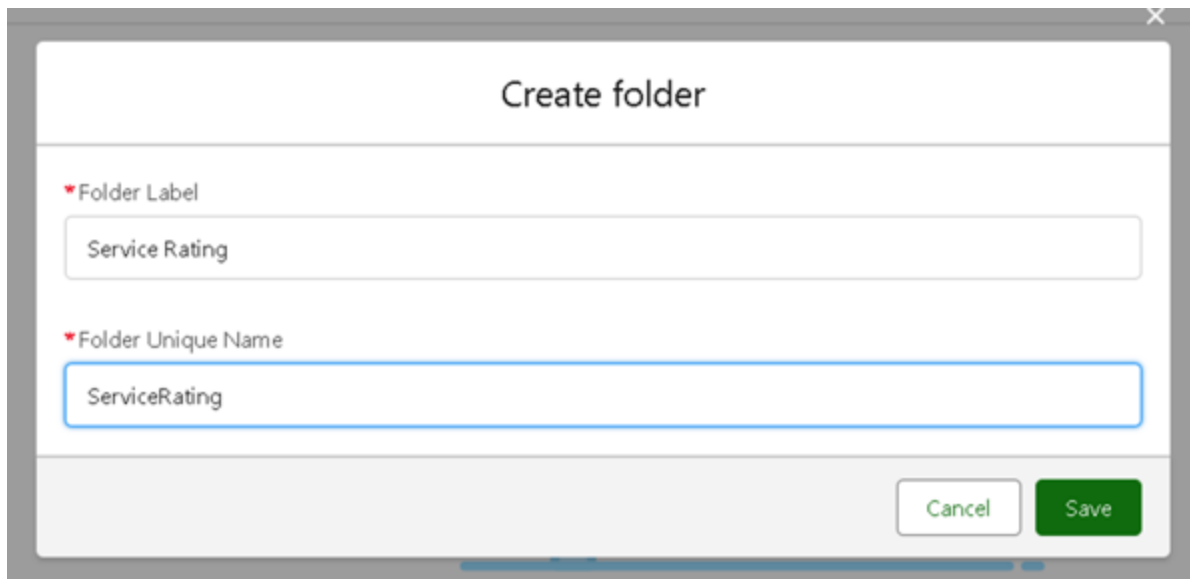
### Steps to Create a Dashboard Folder

#### 1. Access Dashboards:

- Open the **App Launcher**, search for **Dashboards**, and click on the tab.

#### 2. Create a Folder:

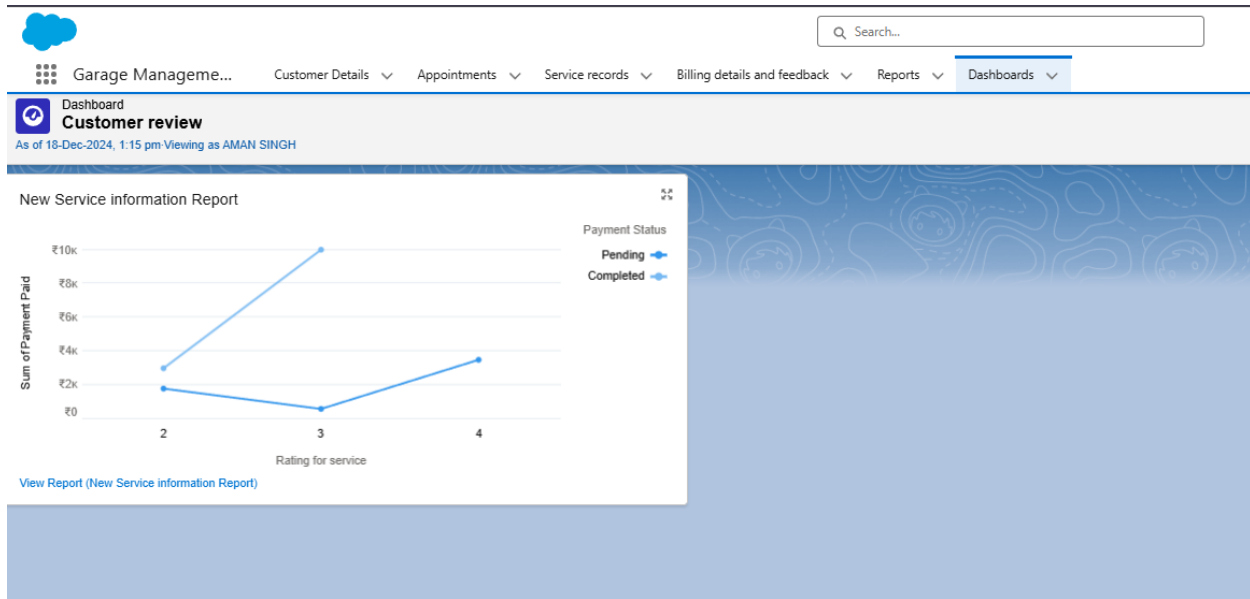
- Click **New Folder** and name it **Service Rating**.
  - The unique folder name will auto-populate.
  - Click **Save**.
3. **Set Sharing Settings:**
- Follow the folder-sharing steps from earlier to grant access to appropriate roles (e.g., Managers with view-only access).



The screenshot shows a 'Create folder' dialog box. It has a title bar with a close button. The main area contains two labeled input fields. The first is 'Folder Label' with a red asterisk, containing the text 'Service Rating'. The second is 'Folder Unique Name' with a red asterisk, containing the text 'ServiceRating'. At the bottom right, there are two buttons: 'Cancel' and 'Save'.

## Steps to Create a Dashboard

1. **Start a New Dashboard:**
  - Open the **Dashboards Tab** → Click **New Dashboard**.
  - Enter a name and select the folder created earlier, then click **Create**.
2. **Add a Component:**
  - Click **Add Component**.
  - Choose the report created earlier (e.g., *New Service Information Report*).
  - Select the **Line Chart** as the visual type.
  - Customize the chart theme as needed.
3. **Save and Finalize:**
  - Click **Add**, then **Save**, and finally **Done**.
  - Preview the dashboard to confirm the visualizations.



## Dashboard Subscription Setup

- Subscribe to Dashboard Updates:**
  - On the dashboard screen, click **Subscribe** in the top-right corner.
- Set Subscription Frequency:**
  - Choose **Weekly** as the frequency.
  - Set the day to **Monday**.
- Save the Subscription:**
  - Click **Save** to complete the subscription process.