



UNIT-1

By
Mrs. P.SHYAMALA

UNIT-I

Salesforce Platform Basics: Understand the Salesforce Architecture,

Data Modeling : Understand Custom & Standard Objects, Create Object Relationships, Work with Schema Builder,

Data Management : Import Data, Export Data,

Picklist Administration: Get Started with Picklists, Manage Your Picklist Values, Share Values with Global Value Sets.

Formulas and Validations: Use Formula Fields, Implement Roll-Up Summary Fields, Create Validation Rules.

String Functions



What is Salesforce?

Salesforce is the name of the famous cloud-base software company in America and name of the application.

Salesforce is a **cloud-based customer relationship management (CRM) platform** that helps **companies manage** their **sales, marketing, customer service, and other business operations in an integrated manner.**

It provides **tools** for managing **customer interactions, analyzing data, and automating various processes to streamline business operations and improve productivity.**

Salesforce is known for its **flexibility, scalability, and ability to customize** to meet the specific needs of different industries and businesses.

It has become a **leading CRM solution globally**, serving organizations of all sizes across various sectors.

Is Salesforce SaaS or PaaS?

Although Salesforce started as a Software as a Service (SaaS) company, it has grown into a Platform as a Service (PaaS) company.

Salesforce as a SaaS Provider

The membership-based licensed software customers buy on demand is **Software as a Service or SaaS**.

We **don't need to install the software** to leverage its benefits; they can only purchase its subscription plan.

We need a browser to access SaaS.

Whatever service you buy, be it for marketing, user interaction, or sales, arrives with **off-the-shelf software(Ready-made)** in Customer Relation Management.

Salesforce as a PaaS Provider

Salesforce's role as a PaaS provider through its **Force.com platform** enables developers and businesses to leverage Salesforce's infrastructure and tools **to build, deploy, and integrate custom applications** tailored to **their specific business needs**, enhancing the overall value and flexibility of the Salesforce ecosystem.

Developers can use Force.com to create custom business logic, workflows, and integrations using **Apex (programming language)** and **Visualforce (markup language)**.

Different Platform of salesforce to create Application are:

- **Force.com** (This is sales force very own developer Platform)
- **Heroku** (Aquired by SalesForce) (Advantage of Heroku is that developers can write application on different Languages i.e (java, python, Ruby on rails))
- **Exact Target Fuel** (Aquired by SalesForce) (Applications created on Fuel platform can only run on Exact Target Market Cloud)



Top 12 companies are using Salesforce CRM in USA

List of prominent companies
using **Salesforce CRM**

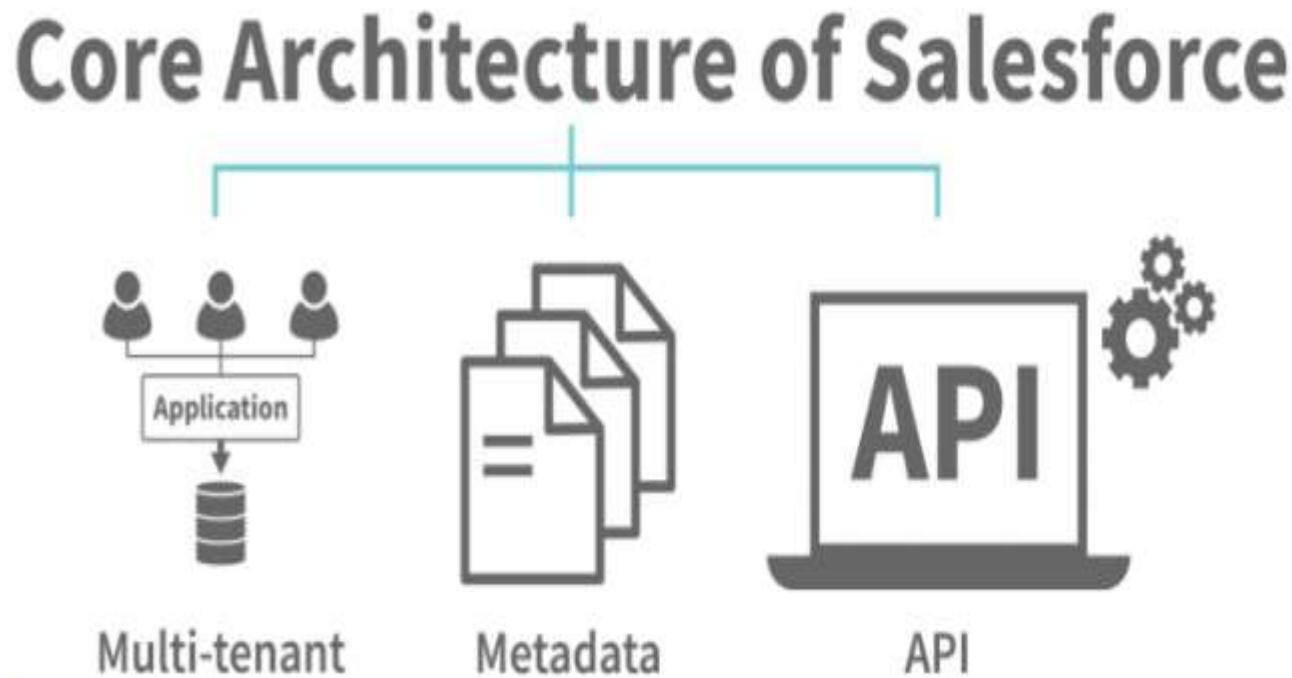


www.CRSINFOSOLUTIONS.COM

Understand the Salesforce Architecture

Core Architecture of Salesforce

The Salesforce architecture is composed of numerous layers that are stacked on top of one another. The diagram below illustrates these layers.

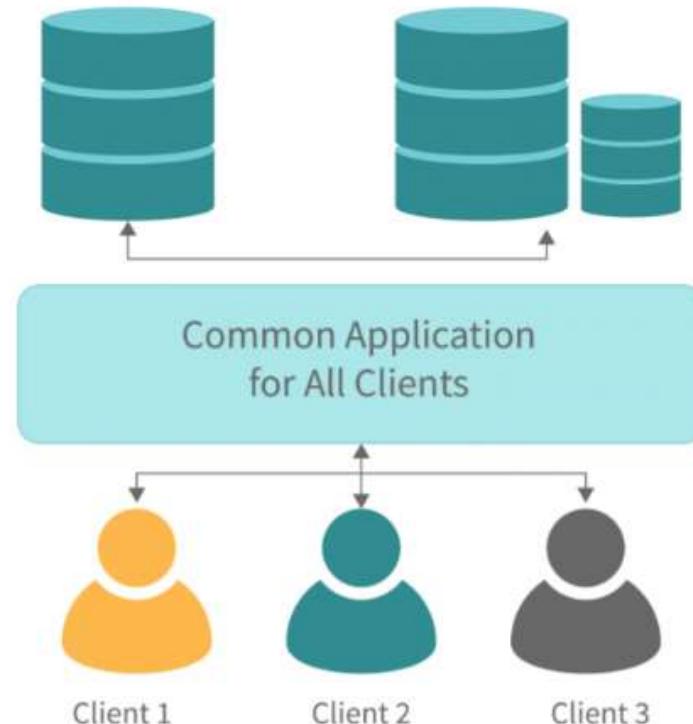


Understand the Salesforce Architecture

Multi-tenant

Multi-tenant architectures, in which **multiple tenants use the same application**, are popular. In this scenario, the developer may create a Salesforce Application and upload it to the cloud, and then share it with multiple clients. The application may be cost-effective because it is **distributed among multiple clients**. The development and maintenance costs of an application can be shared at once, making it very economical per client.

- Salesforce data is placed in a **single database schema** for all its customers.
- Resources and maintenance are shared in the instance of low economic activity.
- A single software server can host multiple tenants, but multiple instances of the same server are not allowed.
- The provider must make one change to all of his clients' applications and those changes will be applied to all of his clients.



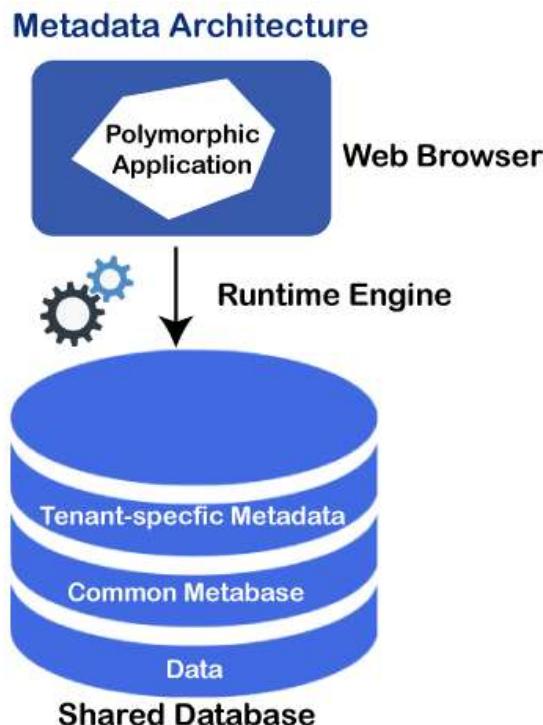
Understand the Salesforce Architecture

Metadata

The Salesforce platform follows the meta-data development model. The metadata means **data about the data**.

Salesforce stores the metadata in the shared database along with the data. It means **it stores the data as well as what data does**.

The tenant-specific data ensures that the common data is only shared with one tenant, not with another tenant or group. This ensures the security of the data even in the shared database.



API Services

The Salesforce metadata-driven model allows the developers to create their applications easily with the help of various tools.

But sometimes developers **need some more functionalities** for their apps to make some modifications. To make such modifications, Salesforce provides a powerful APIs. These APIs helps the developers to **customize the Salesforce application**. These APIs allows the various bits of programming to interface with each other and trade data. Without knowing many details, we can connect our apps with other apps.

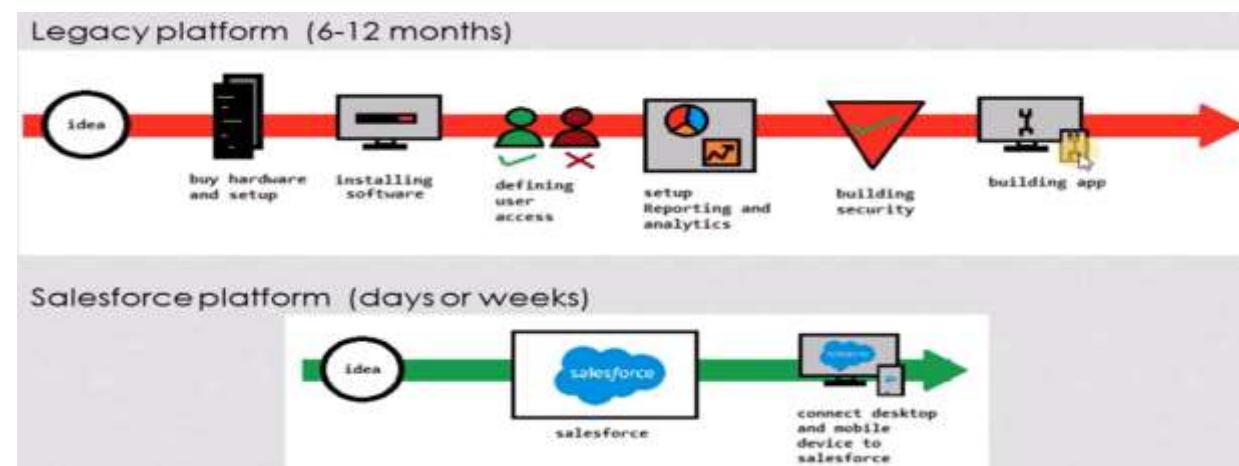
The API provides a simple but powerful, and open way to programmatically access the data and any app running on the Salesforce platform.

These APIs helps the developers to access apps from any location, using any programming language that supports Web services, like Java, PHP, C#, or .NET.

Some important points to get introduced to Salesforce

1. Salesforce is a **multi-tenant cloud company** that is protected by a trusted and multi-cloud ecosystem.
2. All of Salesforce's services are **supported by the Salesforce platform**, which is composed of **numerous metadata parts, including data services, artificial intelligence, and robust APIs for development**.
3. Salesforce Marketing Cloud and Sales Cloud are pre-built offerings within Salesforce that all apps rely on for their functionality. All apps sit atop the Salesforce platform, so you can create your apps using the platform or use **Salesforce's pre-built offerings**.
4. Salesforce Einstein's predictive intelligence and the lightning framework are designed to meet the needs of customers. Everything is integrated with Salesforce. Platform technologies, such as **Salesforce Einstein**, are built to meet the customer's convenience.

SALESFORCE CRM IS DIFFERENT FROM LEGACY CRM?



Create free Developer Account in Salesforce

1. Go to <http://developer.salesforce.com/signup>
2. Fill all details with valid **Email Address**.
3. An Email will b sent to your Email Address.
4. Now click the **link provided in Email**.
5. **Set your password** to your account.
6. Now you are able to login with [salesforce developer](#) account.
7. Go to <https://login.salesforce.com>.
8. Enter your Username and Password click on LogIn.

The screenshot shows the 'Sign up for your Salesforce Developer Edition' page. The page has a blue header with the Salesforce logo and the text 'Build enterprise-quality apps fast to bring your ideas to life'. Below the header is a list of benefits: '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', and 'Customize UI with clicks or any leading-edge web framework'. The main form area is titled 'Complete the form to get access to the Salesforce Developer Edition.' It contains fields for First Name*, Last Name*, Email*, Role*, Company*, Country/Region*, Postal Code*, and Username*. A note below the Username field specifies the requirements: '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.' There is also a link to 'I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement.' At the bottom right of the form is a large blue 'Sign me Up' button. Below the form, there is a link to 'Already have a Salesforce Developer Environment?' and a 'Log In' link.

Terminology in Salesforce

POD in Salesforce

A **POD (Point of Deployment)** is also known as an **Instance** and it is a self contained unit that contains all that is required to run Salesforce, including the **application server, database server, database itself, search and file system**.

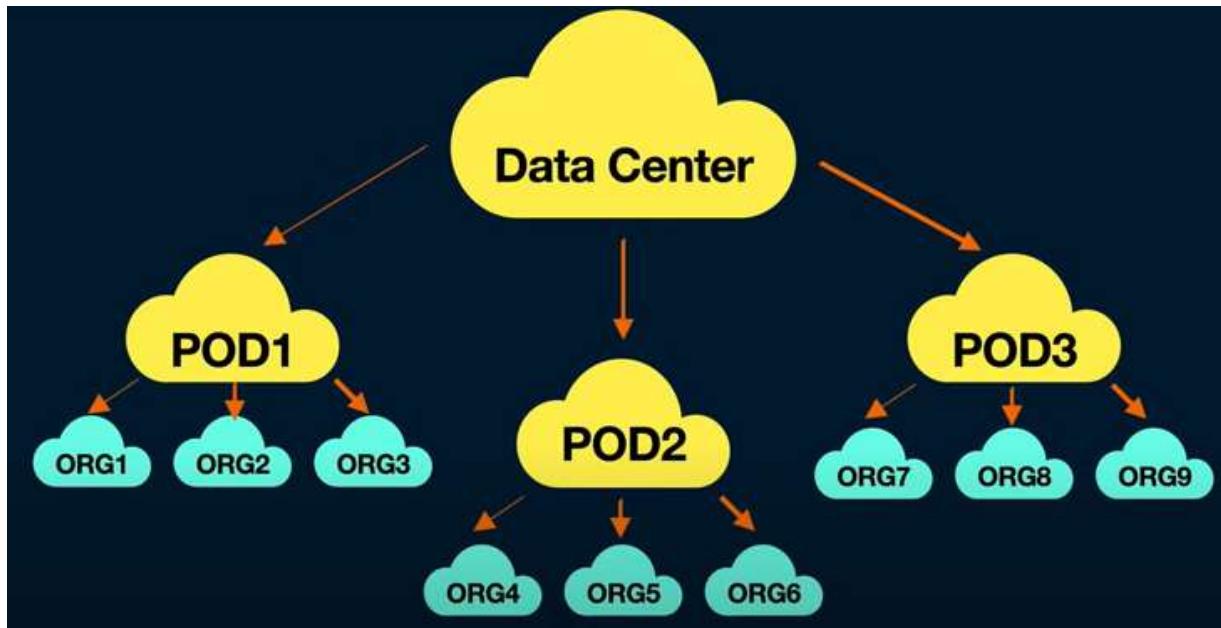
Each customer **is allocated to one and only POD** and that is where their data resides.

Org in Salesforce

An **Salesforce org or organization** is an identifier that represents a **customers version of Salesforce** and its data within an instance.

Each org can be **highly customized** including **custom fields, custom objects, workflows, data sharing rules, visualforce pages and apex coding**.

We can find your own org id in the **Company Information page** within Salesforce.



SETUP Company Information

Company Information
MallaReddy University

The organization's profile is below.

User Licenses (10+) | Permission Set Licenses (10+) | Feature Licenses (11) | Usage-based Entitlements (10+)

Organization Detail

Organization Name	MallaReddy University	Phone	
Primary Contact	Shyamala Padmanabhan	Fax	
Division		Default Locale	English (India)
Address	IN	Default Language	English
Fiscal Year Starts In	January	Default Time Zone	(GMT+05:30) India Standard Time (Asia/Kolkata)
Activate Multiple Currencies	<input type="checkbox"/>	Currency Locale	English (India) - INR
Enable Data Translation	<input type="checkbox"/>	Used Data Space	344 KB (7%) [View]
Newsletter	<input checked="" type="checkbox"/>	Used File Space	13 KB (0%) [View]
Admin Newsletter	<input checked="" type="checkbox"/>	API Requests, Last 24 Hours	0 (15,000 max)
Hide Notices About System Maintenance	<input type="checkbox"/>	Streaming API Events, Last 24 Hours	0 (10,000 max)
Hide Notices About System Downtime	<input type="checkbox"/>	Restricted Logins, Current Month	0 (0 max)

Salesforce.com Organization ID: 00DdM000005ctbp
Organization Edition: Developer Edition
Instance: IND136

Activate Windows

Go to Continue to activate Windows



Data Modeling

Object & Tab, Fields, Records in Salesforce

In Salesforce, an **Object** is a **fundamental data structure** that stores information.

An object is **similar to a database table** and is used to store specific data and information related to your business processes.

Objects in Salesforce are used to organize and manage data within the platform.

They are the **building blocks of Salesforce applications** and are essential for creating robust and customized solutions to meet your business requirements.

Objects are used to represent **real-world or abstract entities** such as customers, products, leads, opportunities, and more.

A **tab** is used to expose an object and its data to the end user through the web interface.

A **field** is one data point within an object (e.g. “Account Name” on the Account object).

A **record** is row of field data within an object.

Fields

A	B	C	D	E
Invoice #	Amount	Service provided	Sent on	Due by
519	\$200.00	Interior painting	10/5/22	10/19/22
520	\$1,000.00	Exterior painting	10/6/22	10/20/22
521	\$300.00	Deck painting	10/7/22	10/21/22
522	\$100.00	Interior painting	10/7/22	10/21/22

Records

Object

Tab & Object

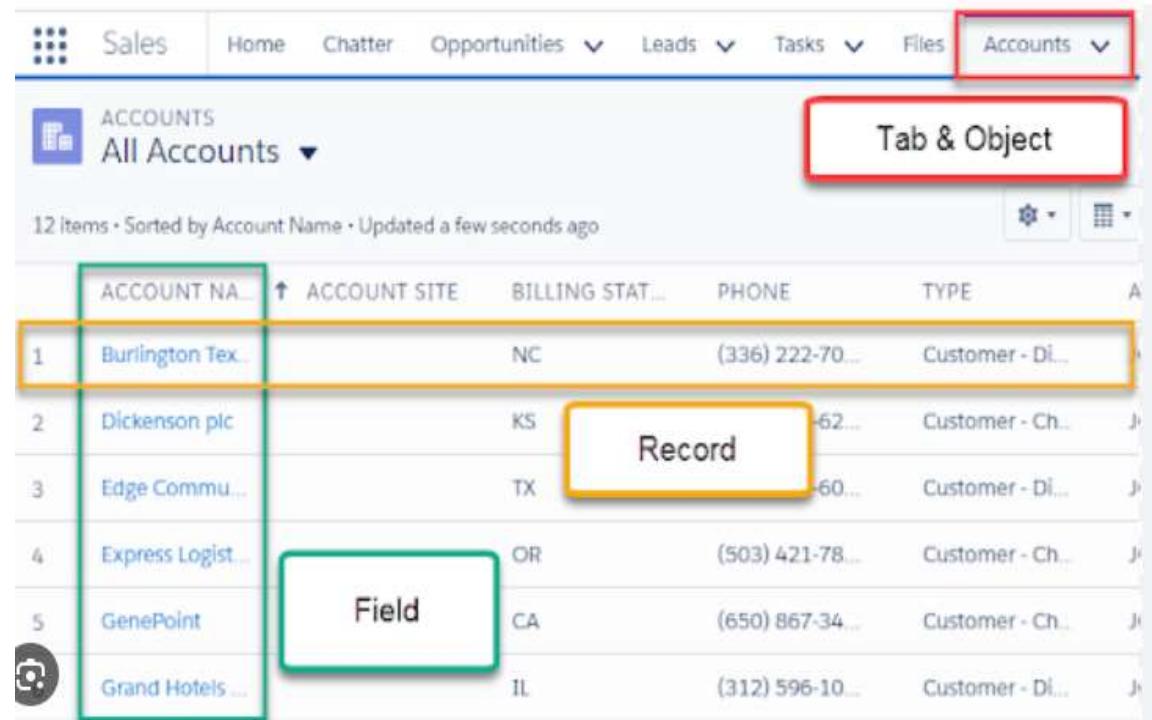
ACCOUNTS All Accounts

12 items • Sorted by Account Name • Updated a few seconds ago

	ACCOUNT NA...	ACCOUNT SITE	BILLING STAT...	PHONE	TYPE
1	Burlington Tex...	NC	(336) 222-70...	Customer - Di...	J
2	Dickenson plc	KS	(312) 555-62...	Customer - Ch...	J
3	Edge Commu...	TX	(503) 421-78...	Customer - Di...	J
4	Express Logist...	OR	(650) 867-34...	Customer - Ch...	J
5	GenePoint	CA	(312) 596-10...	Customer - Di...	J
	Grand Hotels ...	IL			

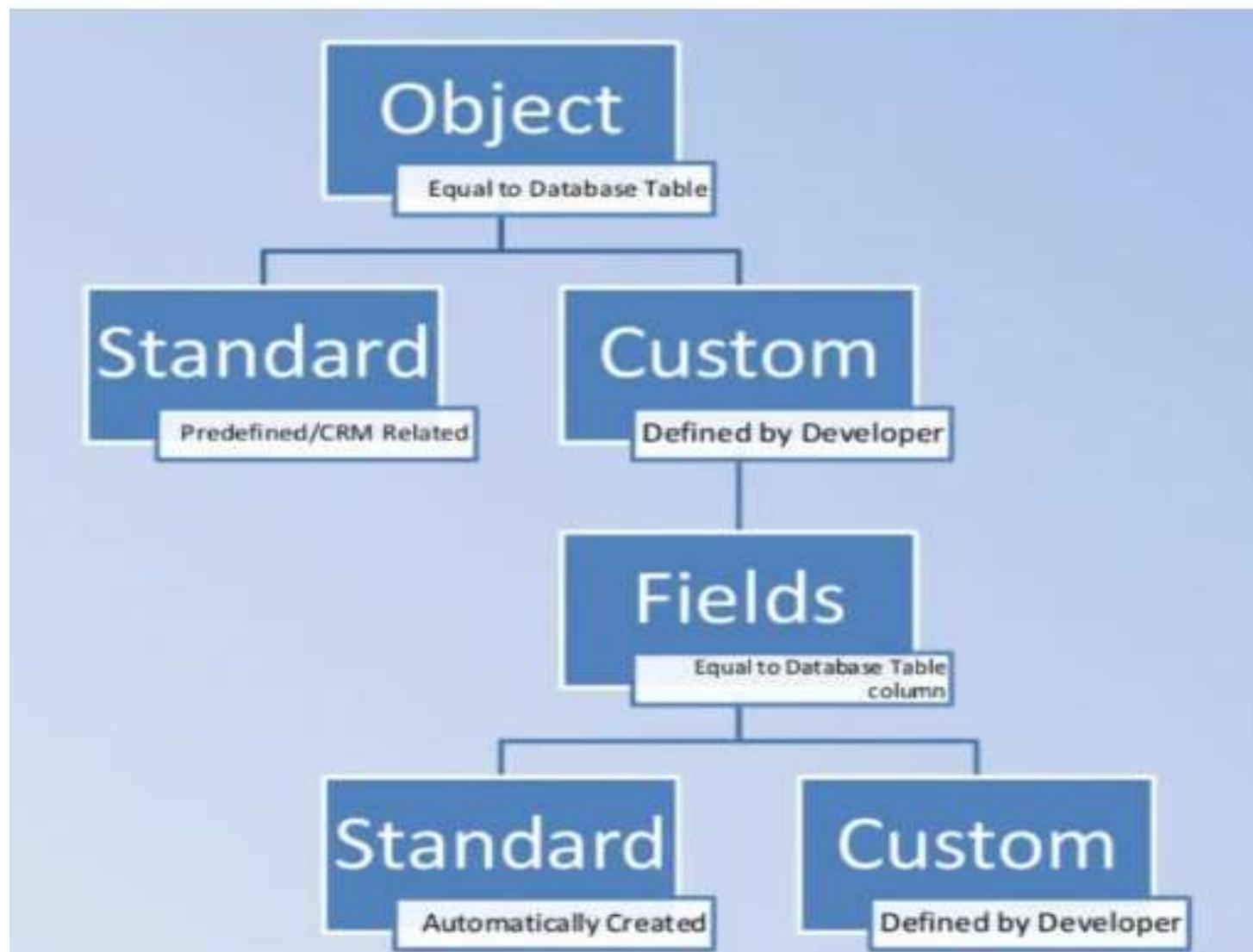
Record

Field



Data Modeling

Types of Object & Fields in Salesforce



Standard Objects in Salesforce

- Standard objects are **already created by Salesforce to use**. Once you log into Salesforce, you can see the available standard objects.
- These include **account, contact, opportunity, lead, campaign**, and so on.
- The most commonly referred standard object is the **Account Object**.
- It is the object that **stores the preliminary information** about a *customer, partner, competitor, or another organization*.

**Users cannot delete standard objects instead,
Salesforce allows users to create custom fields on standard objects.**

List of Standard Objects In Salesforce

Salesforce provides a range of standard objects for users to store and process data. These objects have numerous customization options which users can easily access from the user interface.

1. Account: Refers to a business or organization that you have a business relationship with, such as a customer or partner.

2. Contact: This object represents an individual who is associated with an account, like a customer, partner, or vendor contact.

3. Lead: Refers to a possible sales opportunity. If a lead is deemed qualified, it can be converted into an account, a contact, or an opportunity.

4. Opportunity: Refers to a possible sale that is linked to a particular account. It contains details such as the stage of the sales process, the likelihood of it actually happening, the amount of the sale, and when it is expected to be completed.

5. Case: This object indicates that there is an issue or inquiry from a customer that needs to be resolved.

6. Task: This object describes an item that can be assigned to a user for action, such as a call, email, or meeting.

7. Event: Refers to a type of entry in a calendar, such as a meeting or appointment, which can be linked to other Salesforce items like accounts, contacts, and opportunities.

8. Campaign: Represents a marketing initiative such as an email campaign, trade show, or webinar. It is used to generate leads and drive sales.

9. Quote: Describes a sales proposal that is closely linked to an opportunity. It contains information about the products being sold, their prices, and the terms of the sale.

10. Product: This refers to a product or service that is available for sale to customers and can be linked to opportunities, quotes, and orders.

11. Pricebook: A collection of products and prices for those products. They can be used to manage pricing for different customers or sales channels.

12. Contract: A binding agreement between your organization, a customer, or a partner. It can be associated with accounts and opportunities.

Custom Objects in Salesforce

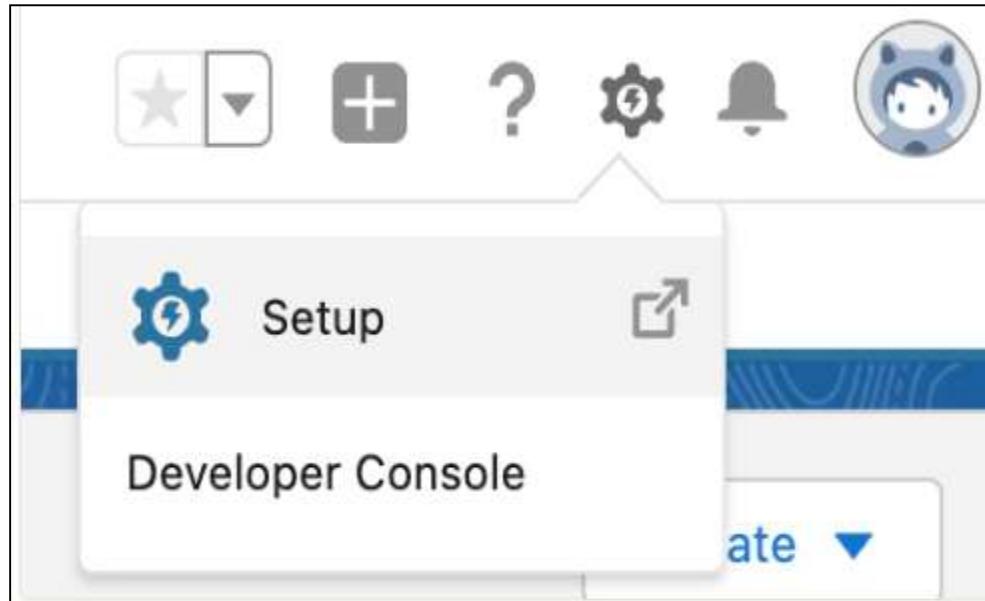
- Custom objects in Salesforce are **customized objects** created to store information **specific to your organization's requirements and processes.**
- It is developed to capture unique data and precise data for your business.
- Contradictory to standard objects, it is **manually created by the user and adds data per the requirement.**
- Custom objects are usually identified by a **c suffix.**

For example, a courier company can create a custom object to store the schedule and dispatch details for every week. So these objects store the data that is unique to the business.

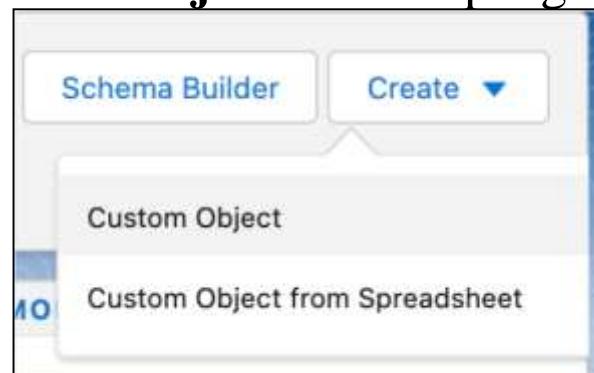
Data Modeling

Steps to Create a Custom object in Salesforce

1. In your Salesforce org, click the cog icon, and select Setup.



2. Click the **Object Manager** tab.
3. Click **Create > Custom Object** in the top-right corner.



Data Modeling

4. In the **Label** section, enter whatever you want to call your custom object. The **Object Name** and **Record Name** fields will **auto-fill** with the same name.
5. For **Plural Label**, enter the plural form of your custom object name.

Custom Object Definition Edit **Save** **Save & New** **Cancel**

Custom Object Information

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

Label	<input type="text" value="Invoice"/>	Example: Account
Plural Label	<input type="text" value="Invoices"/>	Example: Accounts
Starts with vowel sound	<input checked="" type="checkbox"/>	

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

Object Name	<input type="text" value="Invoice"/>	Example: Account
-------------	--------------------------------------	------------------

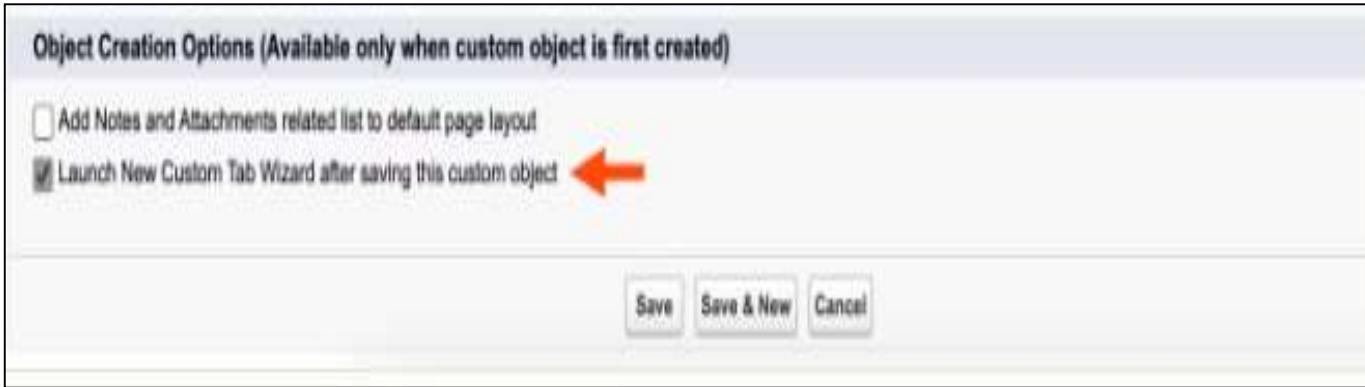
Description

Invoices sent to clients.



Data Modeling

6. Scroll to the bottom of the page, and **select the checkbox *Launch New Custom Tab Wizard after saving this custom object***. Selecting this box will add your custom object as a tab in Salesforce.



7. Click **Save**.

8. On the *New Custom Object Tab* page, click the **Tab Style** field, and choose a style. The style sets the icon to display in the UI for the object.

9. Click **Next**, **Next**, and **Save**.

Field in Salesforce

- In Salesforce, Fields represent exactly the same as what the **columns represent in CSV (Comma Separated Values) files or any relational databases.**
- Basically, it uses to **store data values** of records that are required for a particular object.
- A record is a set of fields value.

In Salesforce there are **two types of fields** that is **standard field** and **custom field**.

1. Standard Fields

Fields that are **already present in Salesforce** for standard and custom objects by default and **cannot be deleted or edited**.

Salesforce provides **four standard fields in every custom object by default** that are:

- **Created By**
- **Last Modified By**
- **Owner and**
- **Created at the time of the creation of an object.**

Data Modeling

2. Custom Fields

The Custom fields are **added by the user according to the business requirements** of any organization.

They may or may not be required.

Users can add different kinds of data depending on the requirements with help of **fields data type which is available on field creation**.

Data Types Of Fields

Data Modeling

Each field in Salesforce has a **defined data type**. When these fields are created in Salesforce, the User has to choose their data type such as Text, Text Area, Number, PickList, etc.

In the below table, we categorized some of the field's data type and their properties.

Data Type	Property	
Lookup Relationship	Create a relationship that links an object to another object.	Allows users to enter any combination of letters and numbers.
Master-Details Relationship	Create a special type of parent-child relationship between two objects.	Allows users to enter up to 255 characters on separate lines.
Checkbox	Allows users to select a True or False value.	Allows users to enter formatted text, and add images and links. Up to 131,072 characters on separate lines.
Date	Allows users to enter or pick a date from a popup calendar.	
Date/Time	Allows users to enter or pick a date with the current time from a popup calendar.	A read-only field that derives its value from a formula expression you define.
Email	Allow user to enter email.	

Data Modeling

Formula

A read-only field that derives its value from a formula expression you define.

Currency

Allows users to enter a dollar or other currency amount and automatically formats.

Geolocation

Allows users to define locations. Includes latitude and longitude components.

Phone

Allows users to enter any phone number.

Picklist

Allows users to select a value from a list you define.

URL

Allows users to enter any valid website address.

Roll-Up Summary

A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.

Picklist (Multi-Select)

Allows users to select multiple values from a list you define.

How To Create Custom Field In Salesforce

- From Setup, go to **Object Manager**.
- Select the **Object** on which you wanted to create a field.
- In the sidebar, click **Fields & Relationships**. Notice that there are already some fields there. There's a name field and some of the system fields you learned about earlier.
- Click **New** in the top right.
- For data type, select **any data type** based on the requirement.
- Click **Next**.
- **Fill out the following:**
 - **Field Label:**
 - **Description:**
- Check the Required box.
- Click Next, Next again, and then Save.

Object Relationships in Salesforce

Relationship in Salesforce refers to the way **two or more objects are connected or related to each other** within the data model.

Relationships define how **records in one object are associated with records in another object**, allowing us to establish **connections and retrieve related data efficiently**.

In Salesforce,

To create a **One-Many relationship** we use the concept of **Parent and Child concept**.

- **Parent** is called as **one side** in one-Many relationship
- **Child** is called as **many side** in one-Many relationship
- The **field is created on the many side object (child)** and it is **related to the one side object.(Parent)**

Types of relationship in Salesforce

1. Lookup Relationship

2. Master-Detail Relationship



Master-Detail Relationship

- A **tightly coupled relationship** where one object (detail) is considered subordinate to another object (master).
- Why we called tightly coupled relationship, **we can't create child record without entering parent value.**
- The **master record(parent) owns the detail record(Child).**
- The two objects are highly dependent on one another.
- When a **parent record is deleted, all of its child records are also deleted.**
- The **Master-Detail Relationship field should be created on the detail object.**
- The **detail record inherits the permission and the sharing rule from master records.**
- Master-Detail Relationship in Salesforce is a one-to-many relationship.
- It is used when we want to control the display of detail records based on the value in the master record.
- **Make sure that no records should be created on detail(Child) object. Because If we have records on Child object we can't create Master Detail Relationship directly.**

“A Standard object cannot be a detail object”

Data Modeling

Key characteristics and benefits of master-detail relationships in Salesforce:

1. Data Ownership and Access Control: In a master-detail relationship, the master object controls the ownership and access settings for the detail records. The ownership of the detail records is transferred to the owner of the master record. This allows for centralized control over data access and sharing rules.

2. Cascade Deletion: With a master-detail relationship, we can define the behaviour for cascade deletion. When a master record is deleted, all associated detail records are automatically deleted. This ensures data integrity and helps maintain data consistency.

3. Roll-Up Summary Fields: Master-detail relationships enable the use of roll-up summary fields. These fields allow us to perform calculations and aggregate data from related detail records onto the master record. This simplifies data analysis and reporting by providing summarized information at the master record level.

4. Validation and Workflow Rules: Master-detail relationships allow us to define validation rules and workflow rules that operate across both the master and detail records. This helps maintain data accuracy and enforce business processes consistently.

5. Security and Sharing: Master-detail relationships inherit the sharing settings and security settings from the parent object to the child object. This ensures that the same level of security and sharing rules apply to both the master and detail records.

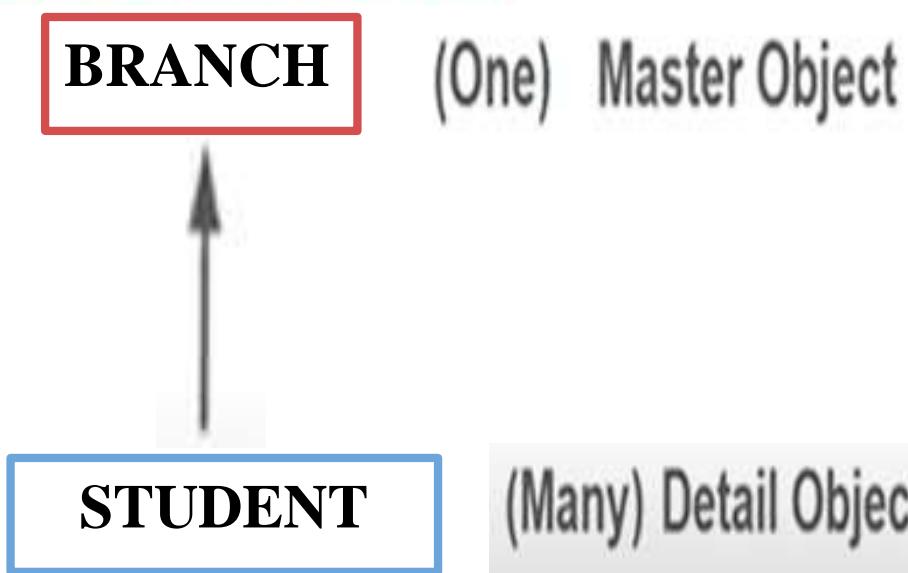
6. Parental Access to Child Records: With a master-detail relationship, the parent object has full access to all related child records. This facilitates data visibility and collaboration between related records.

7. One-to-Many Relationship: Each detail record is associated with a single master record, creating a one-to-many relationship. This means that multiple detail records can be linked to the same master record.

Master-Detail Relationship

EXAMPLE

Master-detail Relationship



- Master –Detail Relationship Field created in Many Object
- No records should be created on detail (Many) Object.

Master-Detail Relationship

(One)
Branch (Master)

B. Id	BName
B1	CSE
B2	AIML
B3	DS

(Many)
Student (Detail)

S. Id	S.Name	FEE	Year
S-01	Student1	50000	II
S-02	Student2	45000	III

After Master-Detail Relationship field created on the Many object

Student (Detail)

S. Id	S.Name	FEE	Year	Rel-B.Id
S-01	Student1	50000	II	B2
S-02	Student2	45000	III	B1

RelSid

Branch (Master)

B. Id	BName	Rel-S. Id
B1	CSE	S-02
B2	AIML	S-01

EXAMPLE

Master-Detail Relationship

Master-detail Relationship

Book (One) Master Object



Book History (Many) Detail Object

- Master –Detail Relationship Field created in Many Object
- No records should be created on detail (Many) Object.

Data Modeling

Master-Detail Relationship

BOOK

B.No.	BName	Price
B1	C++	1200
B2	Python	1700

BOOK History

BH.No	B.Name	Iss Date	Ret Date
H-01	Python	21/4/24	28/4/24
H-02	C++	12/6/24	28/6/24

After Master-Detail Relationship field created on the Many object

BOOK History

BH.No	B.Name	Iss Date	Ret Date	Rel-B.No
H-01	Python	21/4/24	28/4/24	B2
H-02	C++	12/6/24	28/6/24	B1

BOOK

B.No.	BName	Price	Rel-BH.No
B1	C++	1200	H-02
B2	Python	1700	H-01

Steps to create Master-Detail Relationship in salesforce

- **Select the Detail Object:** Identify the object that will serve as the detail object in the relationship.
- **Create a New Field:** Within the detail object's page, click on the Fields & Relationships section, and then click on New.
- **Select Field Type:** In the New Custom Field page, choose Master-Detail Relationship and then click Next.



Steps to create Master-Detail Relationship in salesforce

- **Select Master Object:** You will then be asked to select the master object that you want your detail object to relate to. Choose the appropriate master object from the list and then click Next.



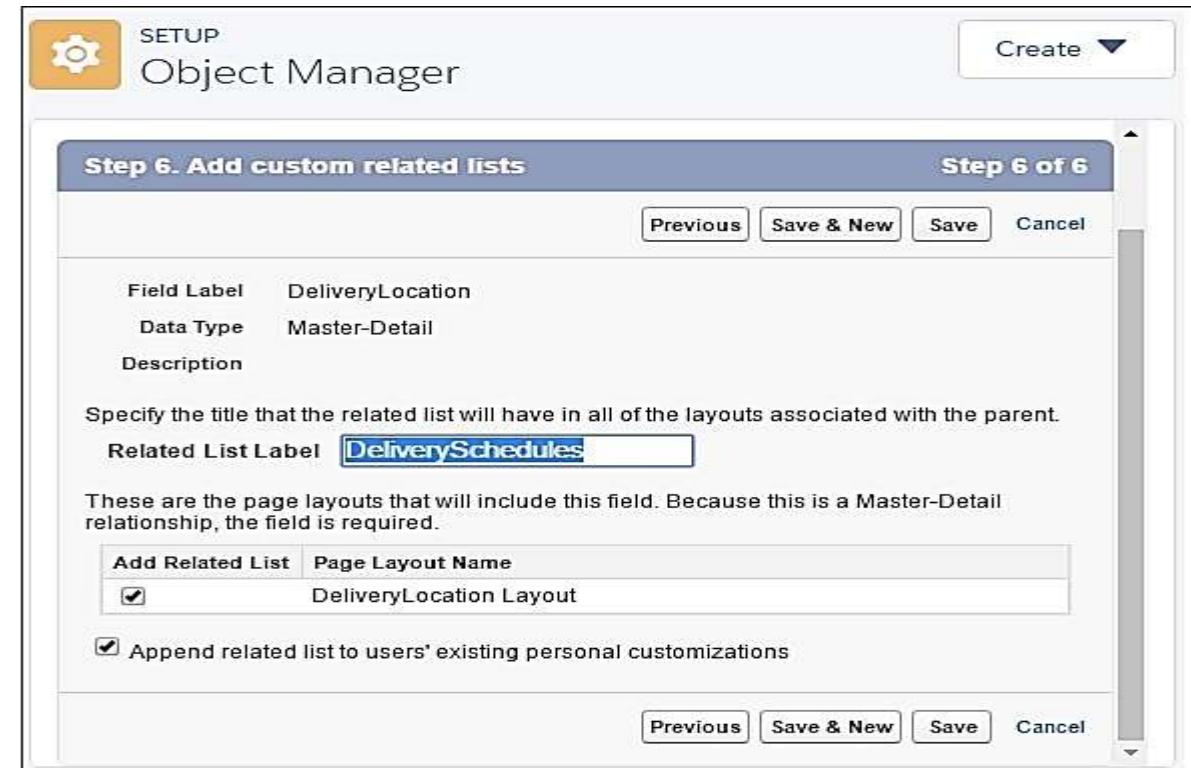
Steps to create Master-Detail Relationship field

- **Define Field Properties:**
- Now, you need to set the field properties.
- Provide a Field Label and a Field Name.
- The field name will be auto-populated based on the label you input, but it can be customized if required.
- Click Next after filling in these details.

The screenshot shows the Salesforce Object Manager interface, specifically Step 3 of 6 for creating a new field. The field is being defined as a lookup field for the 'DeliveryLocation' object. The 'Field Label' is set to 'DeliveryLocation'. The 'Field Name' is also 'DeliveryLocation'. There is no 'Description' or 'Help Text' provided. The 'Child Relationship Name' is set to 'DeliverySchedules'. Under 'Sharing Setting', the 'Read Only' option is selected, which allows users with at least Read access to the Master records. The 'Allow reparenting' checkbox is checked, indicating that child records can be reassigned to other parent records. In the 'Lookup Filter' section, there is a note about optionally creating a filter to limit the records available to users in the lookup field, with a link to 'Tell me more!' and a 'Show Filter Settings' button. Navigation buttons for 'Previous', 'Next', and 'Cancel' are visible at the bottom of the step.

Steps to create Master-Detail Relationship field

- **Set Field-Level Security:** In this step, you can determine which user profiles will be able to see and access the new Master-Detail relationship field. After making your selections, click Next.
- **Add to Page Layout:** The system will then ask you to select the page layouts you want the new Master-Detail relationship field to appear on. Make your selections and click Save.



Lookup Relationship

- A lookup connection is a **loosely connected relationship** in which one object refers to another.
- It enables **one object (child) to link to another object (parent) without requiring the parent record to be present.**
- A single-parent record can have multiple child records linked to it, and the **child record can exist independently of the parent record.**
- Child records in a lookup relationship **are not owned by** the parent record.
- When a **parent record is deleted, the child record is not affected**, and the lookup field on the child record is left empty.
- Lookup relationships do not automatically inherit the parent record's sharing and security settings. Access to the parent record does not allow access to the child records that are related to it.

Data Modeling

Key characteristics of a lookup relationship include:

- 1. Unidirectional Relationship:** Lookup relationships are **unidirectional**, meaning that we can establish a reference from the child object to the parent object, but not vice versa. The parent object does not have any direct knowledge of the records in the child object.
- 2. One-to-Many Relationship:** Each record in the child object can have a reference to only one record in the parent object. However, multiple child records can point to the same parent record, creating a one-to-many relationship.
- 3. Optional Relationship:** Lookup relationships are optional, which means that the lookup field in the child object can be left blank (null) if there is no associated parent record.
- 4. No Cascading Deletion:** Deleting a parent record does not automatically delete the child records. The child records will still exist, but their lookup field will no longer reference the deleted parent record.
- 5. Data Integrity:** Unlike master-detail relationships, lookup relationships do not enforce data integrity constraints. This means that we can create orphaned child records (i.e., child records without a parent) without any error or restriction.
- 6. Use Cases:** Lookup relationships are commonly used when we want to create relationships between standard or custom objects, but we do not require the strong enforcement of data integrity or cascading deletion provided by master-detail relationships.

Data Modeling

Look-Up Relationship

EXAMPLE

Lookup Relationship

TEACHER (One) Master Object



STUDENT

(Many) Detail Object

Lookup Relationship Field created in Many Object

Data Modeling

Lookup Relationship

Teacher (One)

T. Id	TName	Experience
T-001	Ravi	10
T-002	Ashok	12

Student (Many)

S. Id	S.Name	FEE	Year
S-01	Student1	45000	III
S-02	Student2	45000	III
S-03	Student3	50000	II

After Lookup Relationship field created on the Many object

Student (Many)

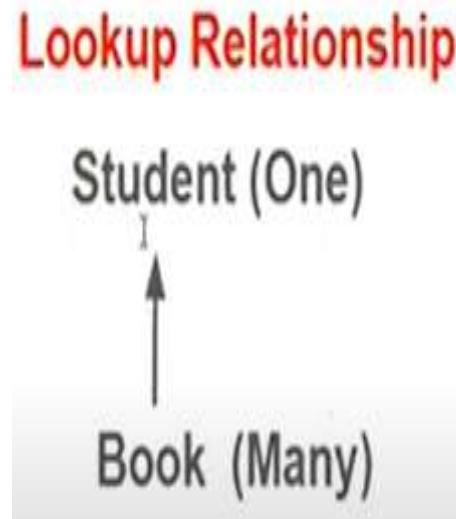
S. Id	S.Name	FEE	Year	Rel. T.Id
S-01	Student1	45000	III	T-001
S-02	Student2	45000	III	T-001
S-03	Student3	50000	II	T-002

RelSid

Teacher (One)

T. Id	TName	Experience	Rel. S.Id
T-001	Ravi	10	S-01 S-02
T-002	Ashok	12	S-03

Lookup Relationship



Lookup Relationship Field created in Many Object

Data Modeling

Lookup Relationship

Student

Std.No	SName
S-001	Ravi
S-002	Ashok

BOOK

B.No.	BName	Price
B1	C++	1200
B2	Python	1700

After Lookuo Relationship field created on the Many object

BOOK

B.No.	BName	Price	Rel Std.No
B1	C++	1200	S-002
B2	Python	1700	S-001

Student

Std.No	SName	Taken Book
S-001	Ravi	B2
S-002	Ashok	B1

Steps to create Lookup Relationship field

1. Go to **SETUP > OBJECT MANAGER**
2. Choose the Child Object that will be part of the Lookup Relationship.
3. Click on Custom Fields and Relationship, then click the New button.

Steps to create Lookup Relationship field

4. Select the Lookup Relationship option and click on Next.

The screenshot shows the 'Step 1. Choose the field type' page for creating a new custom field. The title bar says 'Sample New Custom Field'. The main heading is 'Specify the type of information that the custom field will contain.' Below it, there's a section titled 'Data Type' with five options:

- None Selected - Select one of the data types below.
- Auto Number - A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.
- Formula - A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.
- Roll-Up Summary - A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.
- Lookup Relationship - Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.

The 'Lookup Relationship' option is highlighted with a red box. The top right corner of the page has a 'Step 1' button with a cyan border, which is also highlighted with a cyan box. Below it are 'Next' and 'Cancel' buttons.

5. Choose the Related to Object label, and then select the Child Object. Provide the Field Label, Field Name, and click “Next.”

Steps to create Lookup Relationship field

6. Configure the field-level security for the reference field. Ensure all profiles have access to the field-level security, then click “Save.”



7. Select the page layout for the child object field and select “Next.”
8. Click “Add Custom Related List” in the new window and click “Save.”

Difference between LookUp and Master-Detail Relationship

Look Up Relationship	Master-Detail Relationship
We can create 25 lookup relationships for both standard and custom objects	We can create only 2 master- detail relationships for both standard and custom objects
Look Up Relationship can create even if records already exists in child object	master- detail relationships cant be created if records already exists in child object.
If we delete parent record, then child records will not be deleted/	If we delete parent record, then child records will be deleted automatically.
It is a Optional field	It is a Mandatory field
The ownership and sharing of a child record are not determined by the parent record	The ownership and sharing of a child record are determined by the master record

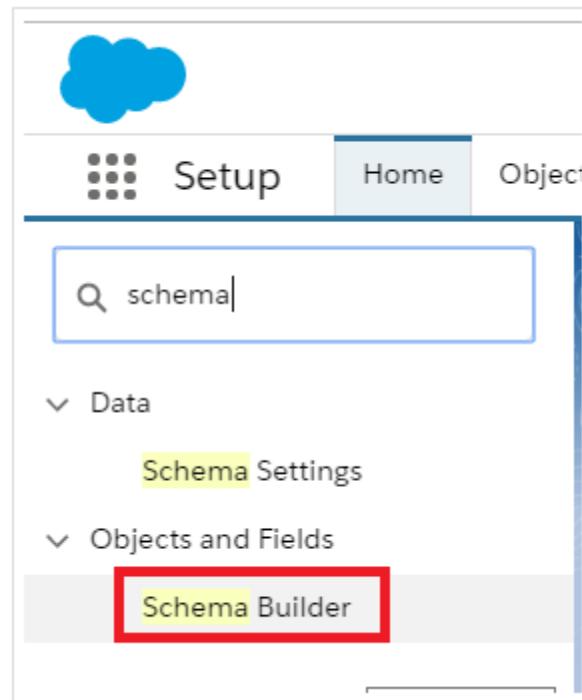
Work with Schema Builder

What is Salesforce Schema Builder?

- The schema builder is a **simple graphical interface for** visualizing and **editing** our **data model**.
- ***Schema builder*** is **a pictorial representation for of all objects, fields and their relationships** in Salesforce.com.
- It provides a dynamic environment to view and modify all the objects and their relationships.
- It's an excellent tool for seeing details like field values and required fields, as well as master-detail relationships and lookup relationships.
- With the use of **schema builder** we can **design data model** with existing objects or **easily add a custom object** or **new field**, and saves the layout of our schema by simply **dragging and dropping**.
- We can **view the details** for both **standard** and **custom** objects in Schema Builder.
- It is helpful to explain **the way data flows** throughout our system.

Work with Schema Builder

1. From Setup, search for and click Schema Builder in the Quick Find box.

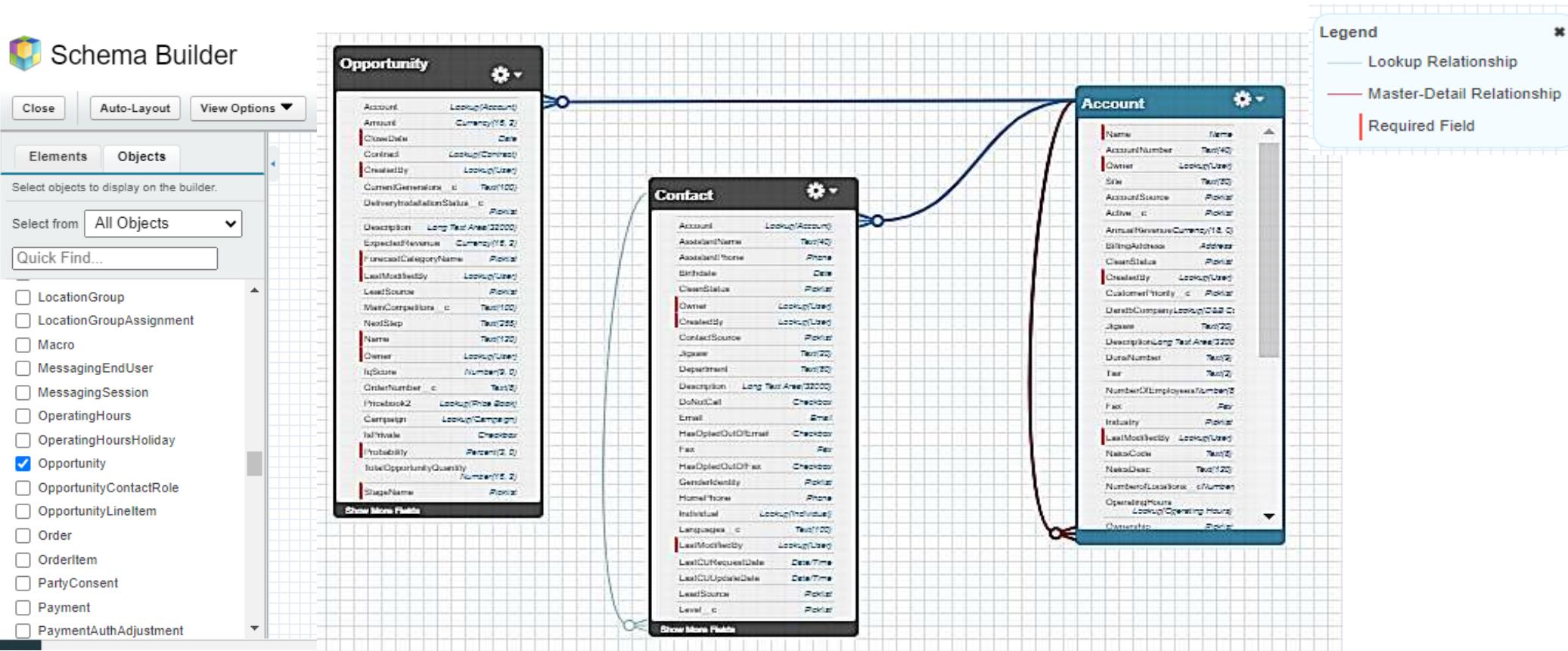


2. In the left panel, click **Clear All**.

Data Modeling

Work with Schema Builder

From the **Objects Palette** select the **Objects**. For example here we are taking **Account**, **Contact** and **Opportunity**.



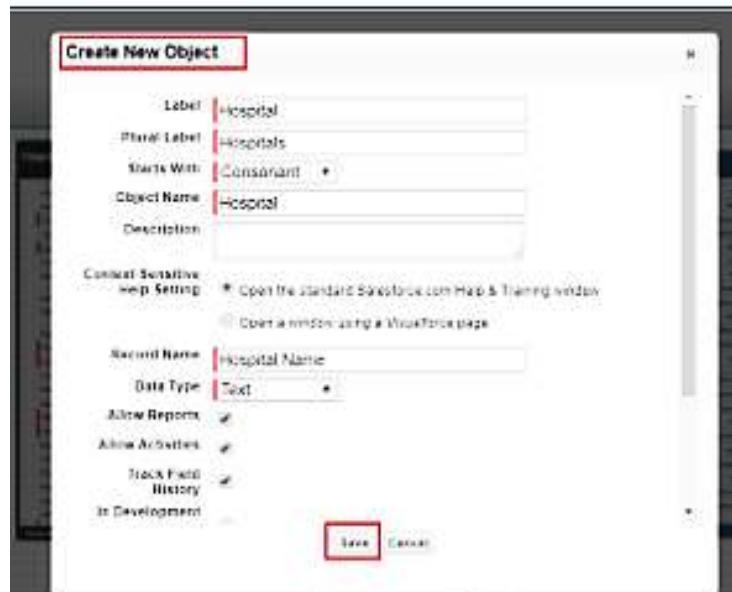
- As shown above we have **three standard objects** and they have some relationships with each other. We can **move** object to any place in Schema Builder. You can drag these objects around the canvas.
- This doesn't change your objects or relationships.
- But it can help you visualize your data model in a useful way.

Blue connections are lookup relationship and red connections are master detail relationship.

Work with Schema Builder

Create an Custom Object with Schema Builder:

- In the left sidebar, click the **Elements** tab.
- Click **Object** and drag it onto the canvas. It prompts you to fill in the details of the object. Fill the detail.



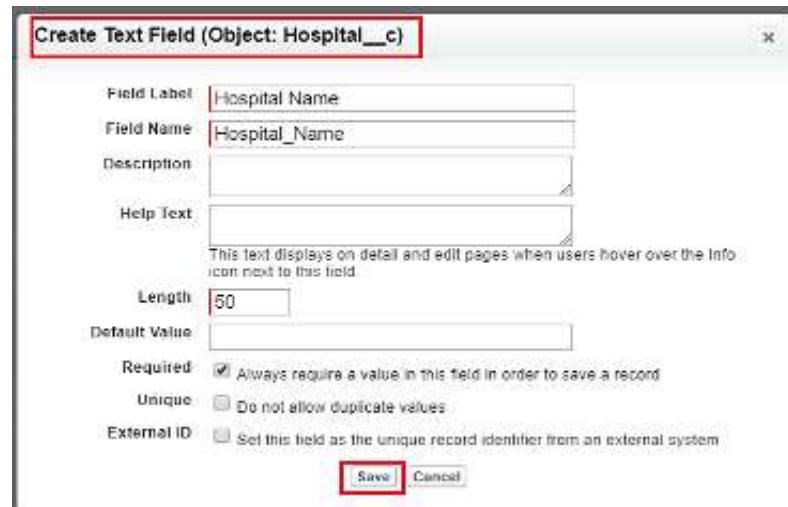
- Enter information about your object. You can make it whatever you want. For example here Hospital object created
- Click **Save**.

Work with Schema Builder

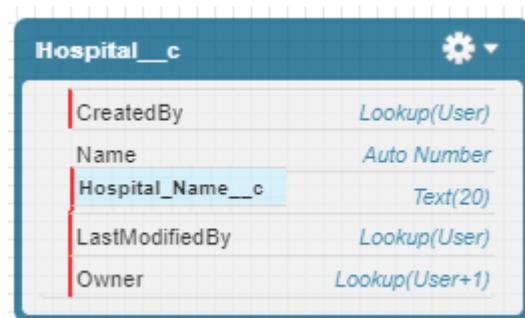
Create Fields with Schema Builder:

Creating fields with Schema Builder is just like creating objects.

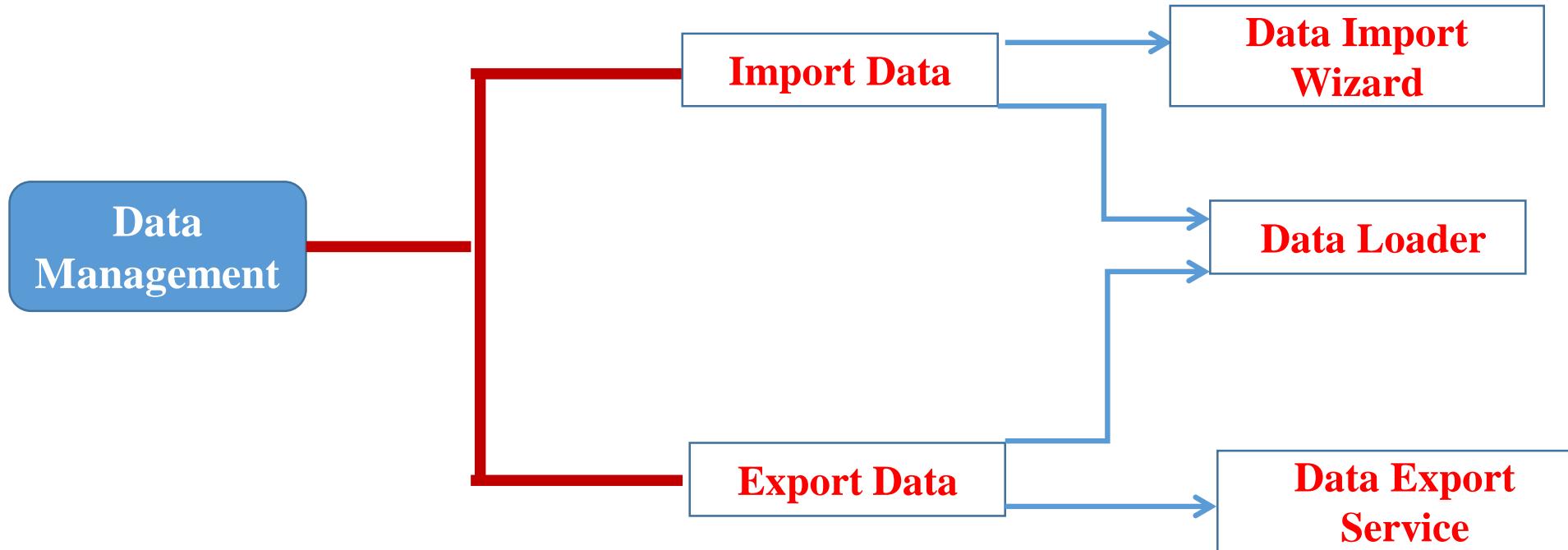
- From the **Elements tab**, choose a **field type** and **drag it onto the object** you just created.



- Notice that you can **create relationship fields, formula fields, and normal fields** in Schema Builder.
- Fill out the details** about your new field.
- Click Save.**



Data Management



Data Management

Data Management means **import/export of data** (or data sets) from/to the external system using out of the box or third-party tools. We can classify into two types:

1. **Data Import**- Importing means **uploading and syncing datasets** from various databases or cloud repositories into your Salesforce account.
2. **Data Export** -Exporting means **creating a copy, extracting, and syncing your data** from Salesforce into **another format and app**, such as exporting a Salesforce report to a spreadsheet file in **CSV (Comma Separated Value) format**.

1. Data Import-

There are many tools available when it comes to importing data into your Salesforce org.

The two most common tools are the

- I. **Data Import Wizard** - a tool available within Salesforce Setup,
- II. **Data Loader** - an external tool provided by Salesforce that easily connects to your orgs.

Data Management

i) Data Import Wizard -

Features of the Data Import Wizard

- The Data Import Wizard is available as an out-of-the-box tool and is accessible through your Salesforce environment – so there is no additional work needed to set it up.
- It can import **less than 50,000 records** at a time.
- It provides a simple interface to specify the configuration parameters, data sources, and field mappings.
- It won't allow you to load **duplicate records**.
- The tool can be **used to import some standard objects**- Accounts, Contacts, Leads, Campaign Members as well as **any custom object**. It **does not support standard objects such as Opportunities**

There are **3 operations possible** while Importing Data in Salesforce using Data Import Wizard:

- **Insert** – Simply creates new records in Salesforce.
- **Update** – Modifies existing records in Salesforce with the help of a record id or external id.
- **Upsert** – Combination of insert and update. It modifies the existing records and creates a new one if the record is unavailable in the organization.

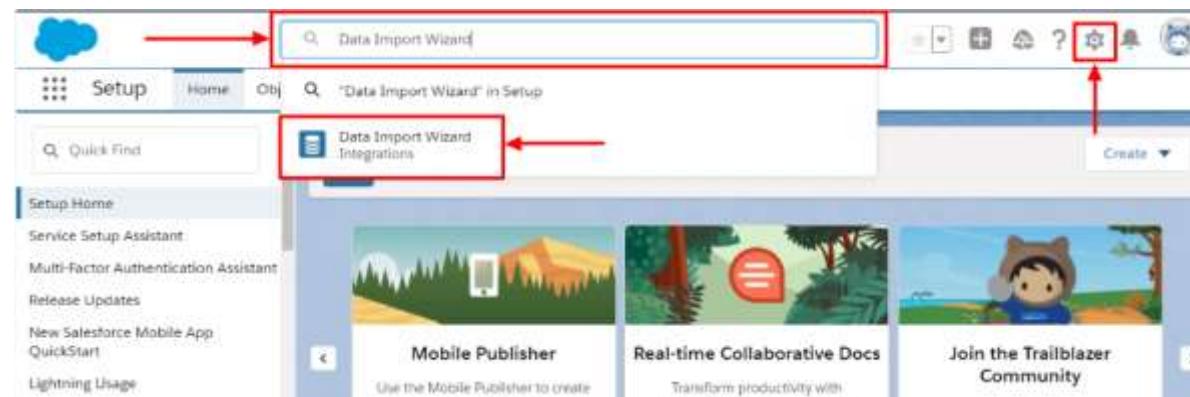
Data Management

Steps to Use the Data Import Wizard:

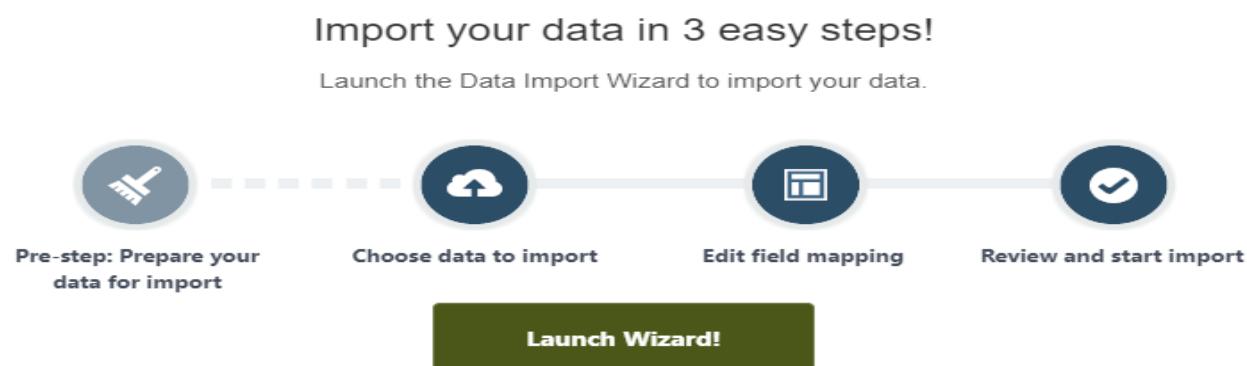
Step 1: Prepare your file for import After preparing your import data, **save it as a CSV file**.

Step 2: Start the wizard.

- i. From Setup, enter Data Import Wizard in the Quick Find box, then select **Data Import Wizard**.



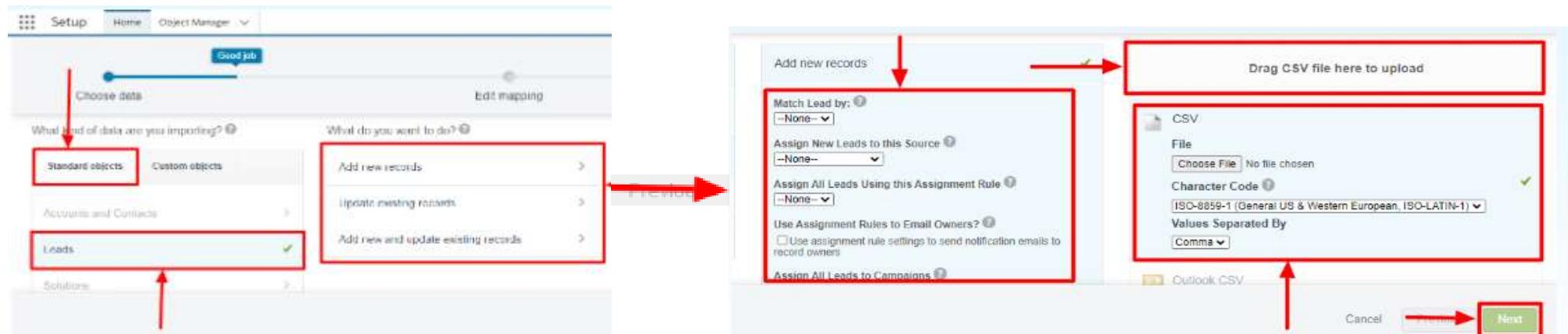
- ii. Review the information provided on the welcome page, then click **Launch Wizard!**



Data Management

Step 3: Choose the data that you want to import.

- i. To import accounts, contacts, leads, solutions, person accounts, or campaign members, click **Standard Objects**.
- ii. To import custom objects, click **Custom Objects**.
- iii. Specify whether you want to add new records to Salesforce, update existing records, or add and update records simultaneously.
- iv. Specify matching and other criteria as necessary. Hover over the question marks for more information about each option.
- v. Specify the file that contains your data. You can **specify your data file by dragging the CSV to the upload area** of the page or by clicking the CSV category you're using and then navigating to and selecting the file.



- vi. Choose a character encoding method for your file. Most users can accept the default character encoding.
- vii. Click **Next**.

Data Management

Step 4. Map your data fields to Salesforce data fields. Salesforce will mark your **unmapped fields**. Click *Map* under the *Edit* column to change and map the fields to an existing Salesforce field.

Setup	Home	Object Manager	Alfred done
Choose data		Edit mapping	
Edit	Mapped Salesforce Object	CSV Header	Example
Change	Last Name	First Name+Family Name	Brian
Change	E-mail	E-mail	Stark
Change	Phone	Phone	(415) 828-1690
Change	Title	Title	Director of IT
Change	Company	Company	123 Stark Industries
Map	Unmapped	Address Line 1	1775 18th

Review and start your import.

Review your import information on the Review page. If you still have unmapped fields that you want to import, click **Previous** to return to the previous page and specify your mappings.

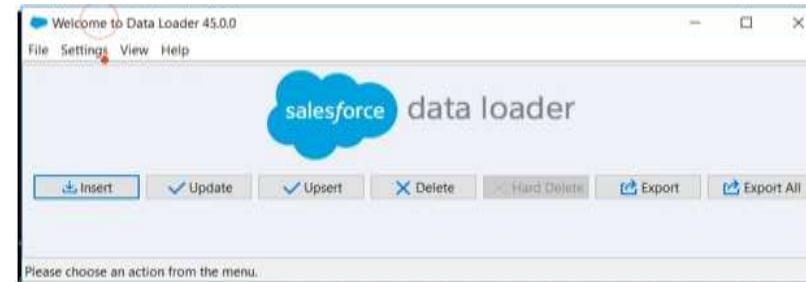
Click **Start Import**

Step 5. Check import status. From Setup, enter “Bulk Data Load Jobs” in the Quick Find box, then select **Bulk Data Load Jobs**. The user who starts the data import receives a status email when the import is completed.

Data Management

ii) Data Import Wizard

- The Data Loader tool is the most **efficient external tool** to move data rapidly into the platform. It is used not only for **importing and exporting data** but also does **updating and deleting data**.
- The Data Loader is a complex tool that supports all standard objects and custom objects.



- a client application that can **import up to 150 million records** at a time, of any data type, either from files or a database connection the ability to handle large volumes of data, the Data Loader is ideal for complex data migration tasks.
- It can be operated either through the **user interface or the command line**.
- You need to specify data sources, field mappings, and other parameters via configuration files.
- This makes it possible to automate the import process, using API calls.

Use Data Loader When:

You need to load 50,000 to 150 million records.

You need to load into an object that is not supported by the Data Import Wizard.

You want to schedule regular data loads, such as nightly imports.

Data Import Wizard	Data Loader
Used for simple data imports.	Used for complex data imports.
Can load up to 50,000 records.	Can load up to 5,00,000 records.
Supports all the custom objects and only a few standard objects like Account, Contact, Campaign members, person accounts, Leads, and Solution.	Supports all custom and standard objects.
Supports schedule export.	Doesn't support scheduled export.
Delete operation is not available.	Delete operation is available.
Cannot import cases and opportunities.	Can import cases, events, tasks, and opportunities.
Duplicates can be ignored while importing.	Duplicates cannot be ignored while importing.
Doesn't require installation.	Requires installation.

Data Management

2. Data Export :

- Easily export data from Salesforce, either manually or on an automatic schedule.
- The data is exported as a set of **comma-separated values (CSV) files**.
- Data export tools provide a convenient way to obtain a copy of your Salesforce data, either for backup or for importing into a different system.

Salesforce offers two main methods for exporting data.

I. Data Export Service

II. Data Loader

Data Management

Data Export Service:

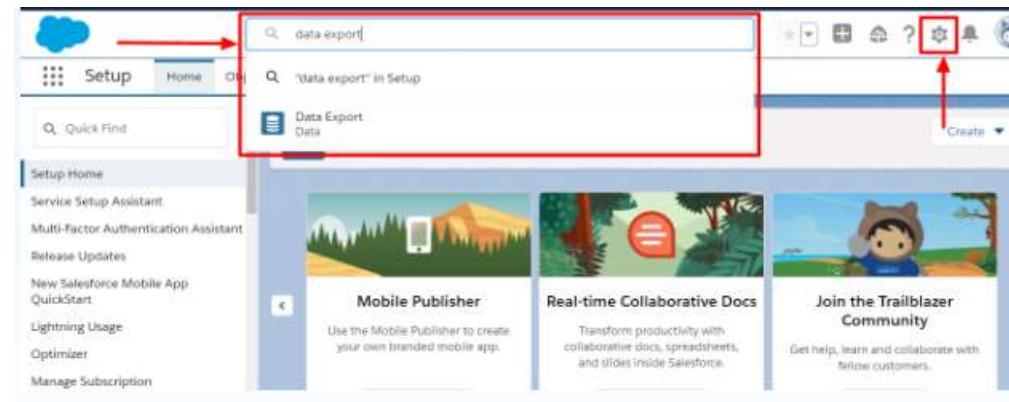
- Accessible through the **Setup menu**.
- It allows you to **export data** manually **once every 7 days** (for weekly export) **or 29 days** (for monthly export).
- In **Professional Edition and Developer Edition**, you can **generate backup files only every 29 days**, or automatically at **monthly intervals** only.

Data Management

Steps to Use the Data Export Service:

Step 1: Navigate to the Data Export option on Salesforce

On the Salesforce interface, navigate to *Setup* and enter *Data Export* in the search box.



Choose either Export Now or Schedule Export.

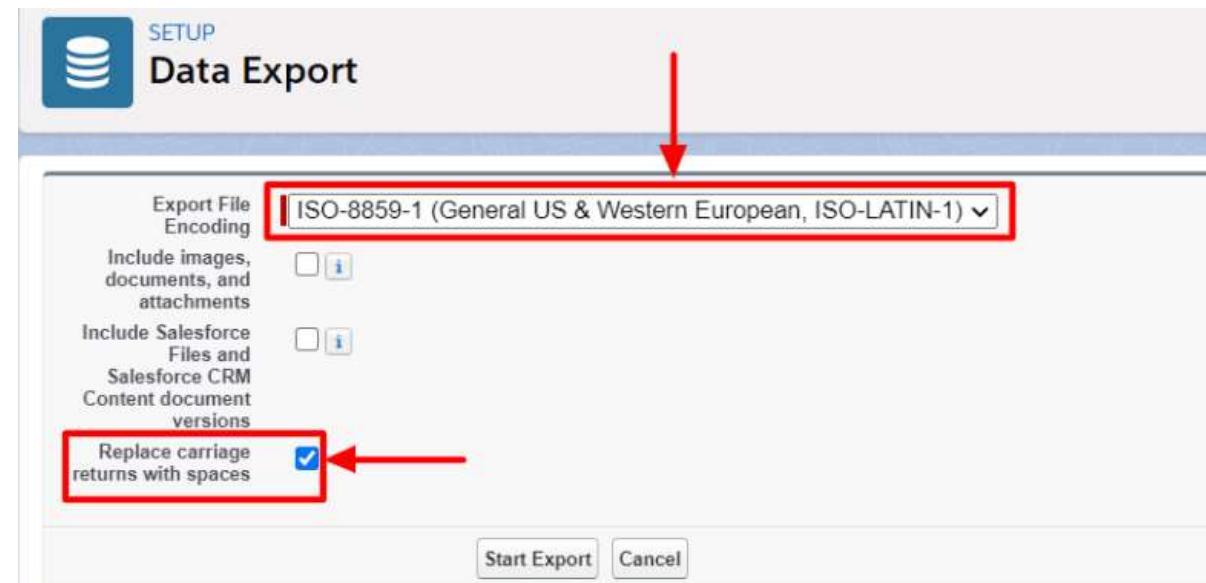
- The **Export Now** option prepares your files for **export immediately**. This option is only available if enough time has passed since your last export.
- The **Schedule Export** option allows you to schedule the export process to **run at monthly intervals**.

Data Management

Step 2: Choose the desired encoding for your export file

If you want images, documents, attachments, and so on included in your data, select the appropriate options.

Select **Replace carriage returns with spaces** to have spaces instead of carriage returns or line breaks in your export files. This is useful if you plan to use your export files for importing or other integrations.



Data Management

Select the types of info you want to include in your export. If you're not familiar with Apex API names, choose *Include all data* and click *Start Export*.

Start Export Cancel

Exported Data
Select what type of information you would like to include in the export. The data types listed below use the Apex API names. If you are not familiar with these names, select *Include all data* for your export.

Include all data

Contract Order OrderItem
 Approval ContractContactRole RecordType
 BusinessProcess EntityHistory FieldHistory
 EmailRoutingAddress Campaign CampaignMember
 Account Contact Lead

Step 3: Schedule your export

Configure your export schedule by setting the frequency, (only available for organizations with monthly exports), start and end dates, and time of day for your scheduled export.

SETUP Data Export

with spaces
Schedule Data Export

Frequency

On day 1 of every month On the 1st Sunday of every month

Start: 3/29/2021 End: 4/29/2021

Preferred Start Time: -None-

Exact start time will depend on job queue activity

Save Cancel

Data Management

Click **Save** when you're done (or **Start Export** if you're not scheduling, but instead exporting your Salesforce files immediately)..

Salesforce **creates a zip archive of CSV files and emails you when it's ready.**

Exports will complete as soon as possible. However, we can't guarantee the date and time the export will complete.

Large exports are broken up into multiple files.

Follow the link in the email or click **Data Export** to download the zip file.

Zip files are deleted 48 hours after the email is sent.

Data Management

Data Loader:

- A client application that you must install separately.
- It can be operated either through the user interface or the command line.
- The latter option is useful if you want to automate the export process, or use APIs to integrate with another system.

Salesforce Picklist Fields

- A picklist is a **list box of pre-defined values**. The user can only select one of the pre-defined values.
- Only the Administrator has the ability to add or delete picklist values. It is a **simple dropdown menu** of options.

Values in Salesforce Picklist

In salesforce value can be defined in three ways:

- We can **use a predefined picklist** which is a **standard picklist field** by salesforce.com.
- When we **create a picklist** we need to **set individual values** (specific to a single picklist field).
- Create a **global value set**.
When we need to share with more than one picklist field, we can **set a global value set**.

Picklist Administration

There are three different types of picklists available in Salesforce:

1. Standard
2. Custom
3. Custom Multi-Select

1. Standard Picklists:

- These picklists are the **default options provided within your Salesforce org**, requiring no customization.
- They typically **consist of common fields essential for standard objects** within your organization.
- For example,

In a **Lead object**, there is a **picklist field named LeadSource** which is a standard picklist. It has some standard value sets.

It also utilized for the Account Source picklist field on the Account object. Changes made to these standard picklist values are reflected across related fields.

Standard Picklists

The screenshot shows the Salesforce Object Manager interface for the Lead object. The top navigation bar includes 'Setup', 'Home', 'Object Manager', and a dropdown menu. Below the navigation is a breadcrumb trail 'SETUP > OBJECT MANAGER' and the object name 'Lead'. On the left, a sidebar lists various configuration tabs: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, and Object Limits. The main content area is titled 'Fields & Relationships' with a sub-label '32 Items, Sorted by Field Label'. It contains a table with columns for Field Label, API Name, and Type. A red oval highlights the 'Lead Source' row, which has 'LeadSource' as its API name and 'Picklist' as its type. Other visible rows include 'Last Modified By', 'LastTransferDate', 'OwnerId', and 'Status'.

	Field Label	API Name	Type
Last Modified By	LastModifiedById	Lookup(User)	
Last Transfer Date	LastTransferDate	Date	
Lead Owner	OwnerId	Lookup(User,Group)	
Lead Source	LeadSource	Picklist	
Lead Status	Status	Picklist	

Picklist Administration

2. Custom Picklists: (**Global value set**)

As their name suggests, these picklists are **user-created** according to specific requirements. Users can add their own values and configure custom picklists' behavior to align with their needs.

A Global Picklist Value Set in Salesforce means a picklist which **you can access for all salesforce objects**.

Steps to create a Global Picklist value set:

Step 1: Go to '**Picklist Value Sets**' under Setup → Home → Quick find.

The screenshot shows the Salesforce Setup interface. At the top, there's a navigation bar with a cloud icon, 'Setup', 'Home', and 'Object Manager'. A search bar says 'Search Setup' with the word 'picklis' typed in. Below the navigation is a sidebar with sections like 'Data' (containing 'Picklist Settings' and 'State and Country/Territory Picklists') and 'Objects and Fields' (containing 'Picklist Value Sets'). The 'Picklist Value Sets' section is highlighted with a blue bar. The main content area has a title 'SETUP Picklist Value Sets' with a blue icon. It says 'Picklist Value Sets' and explains that global picklist value sets let you share values across objects. It shows a table header for 'Global Value Sets' with columns for 'Label' and 'Description'. A 'New' button is at the top right of the table. Below the table, it says 'No records to display.' and 'Deleted Global Value Sets (0)'. At the bottom left, there's a note: 'Didn't find what you're looking for? Try using Global Search.'

Picklist Administration

Step 2: Click on ‘New’ button.

Picklist Value Sets Help for this Page 

Global picklist value sets let you share the values across objects. Base custom picklist fields on a global value set to inherit its values. The value set is restricted so users can't add unapproved values through the API.

View: All ▾ Create New View

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z | Other All

Global Value Sets		New
Label 		
No records to display.		
Deleted Global Value Sets (0)		

Step 3: Enter a label, name, and description. Then put all the Picklist values in the provided Text area as one value per line.

Global Value Set

Global Value Set Edit Save  Save & New  Cancel 

Information

Label	<input type="text" value="Country"/>
Name	<input type="text" value="Country"/>
Description	<input type="text" value="This contains all the Country names"/>

Values

Enter values, with each value separated by a new line

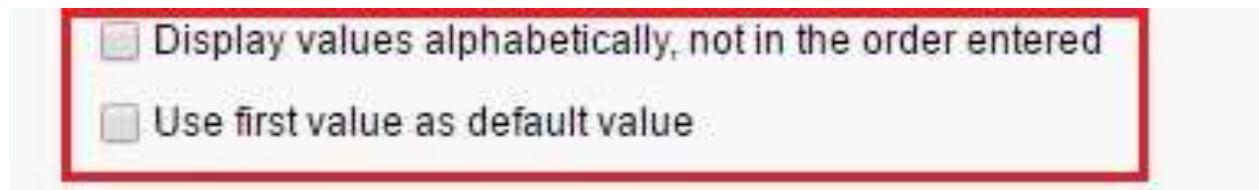
```
India
USA
Australia
United Kingdom
```

Display values alphabetically, not in the order entered
 Use first value as default value

Save  Save & New  Cancel 

Picklist Administration

Step 4: Optionally choose to sort the values alphabetically or to use the first value as the default value, or both. You can't change these settings later.



Step 5: Click Save

Global Value Set

[« Back to List](#)

[Printable View](#) | [Help for this Page](#)

Global Value Set Detail

[Edit](#) [Delete](#)

Information

Label:	Country
Name:	Country
Description:	This contains all the Country names

[Edit](#) [Delete](#)

Values

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	India	India	<input type="checkbox"/>	Assigned dynamically	Salesforce Training, 6/20/2017 11:07 PM
Edit Del Deactivate	USA	USA	<input type="checkbox"/>	Assigned dynamically	Salesforce Training, 6/20/2017 11:07 PM
Edit Del Deactivate	Australia	Australia	<input type="checkbox"/>	Assigned dynamically	Salesforce Training, 6/20/2017 11:07 PM
Edit Del Deactivate	United Kingdom	United Kingdom	<input type="checkbox"/>	Assigned dynamically	Salesforce Training, 6/20/2017 11:07 PM

Inactive Values

No Inactive Values values defined.

Fields Where Used

No records to display

Picklist Administration

Step 6: Choose “Use global picklist value set”

Step 2. Enter the details

Field Label	<input type="text" value="CountryName"/> i
Values	<input checked="" type="radio"/> Use global picklist value set <input type="radio"/> Enter values, with each value separated by a new line <input style="border: 1px solid black; padding: 2px; width: 100px; height: 25px; font-size: 10px;" type="button" value="Country"/>
	<input type="checkbox"/> Display values alphabetically, not in the order entered <input type="checkbox"/> Use first value as default value i <input checked="" type="checkbox"/> Restrict picklist to the values defined in the value set i
Field Name	<input type="text" value="Country"/> i
Description	<input type="text"/>
Help Text	<input type="text"/>
Required	<input type="checkbox"/> Always require a value in this field in order to save a record
Auto add to custom report type	<input checked="" type="checkbox"/> Add this field to existing custom report types that contain this entity i
Default Value	<input type="text" value="Show Formula Editor"/> i Use formula syntax: Enclose text and picklist value API names in double quotes : ("the_text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: SCustomMetadata.Type__mdt.RecordAPIName.Field__c

Picklist Administration

2. Custom Picklists: (Enter values, with each value separated by a new line)

- Select Enter values, with each value separated by a new line.

Product
New Custom Field

Help for this Page 

Step 2. Enter the details Step 2 of 4

Previous Next Cancel

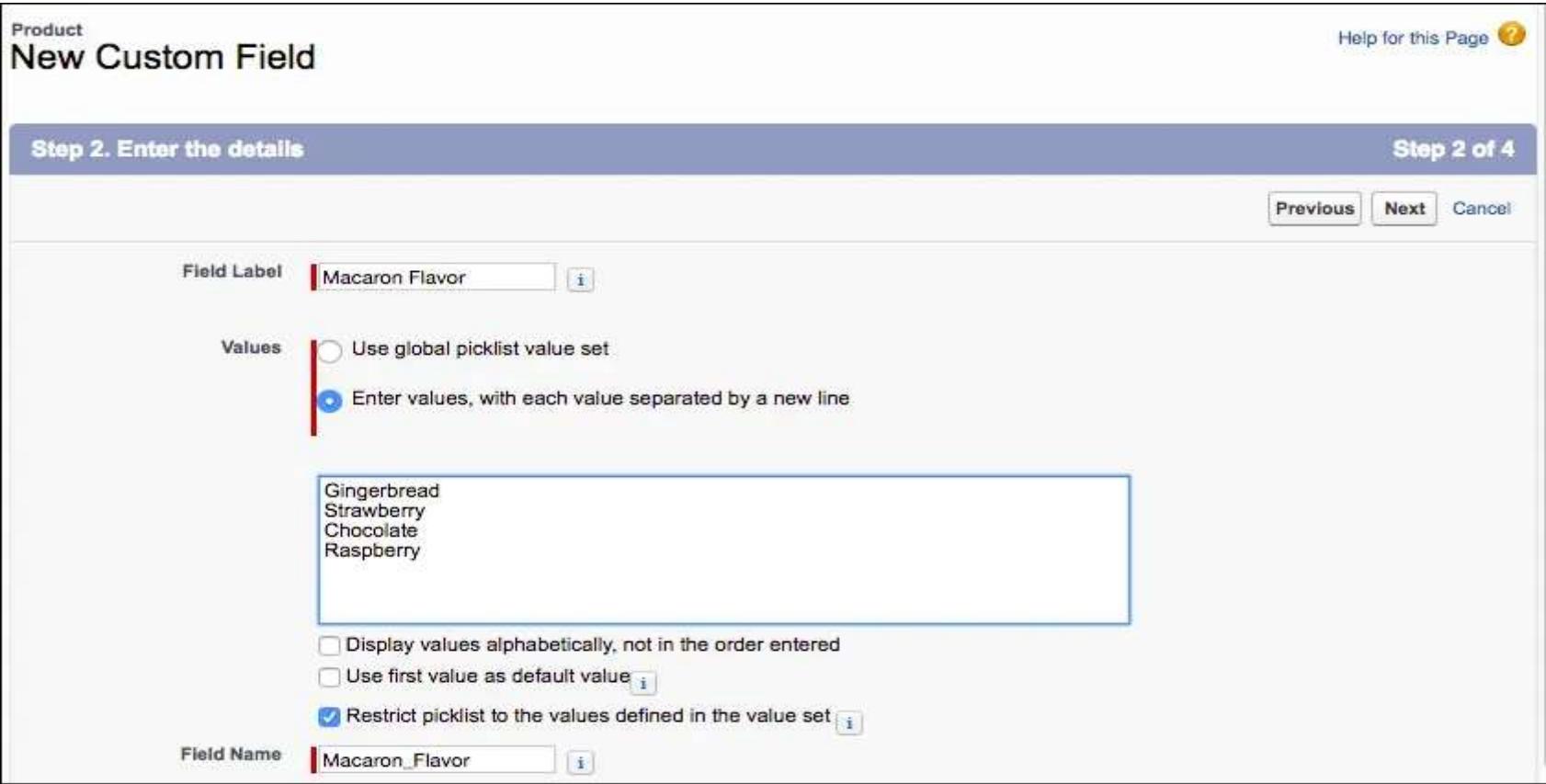
Field Label 

Values Use global picklist value set
 Enter values, with each value separated by a new line

Gingerbread
Strawberry
Chocolate
Raspberry

Display values alphabetically, not in the order entered
 Use first value as default value 
 Restrict picklist to the values defined in the value set 

Field Name 



Picklist Administration

3. Custom Multi-Select Picklists:

This feature allows users to **select more than one value from the picklist simultaneously**. Once enabled, the selected values are displayed in the field, separated by a semicolon. Unlike standard and custom picklists, which permit selection of only one value at a time, multi-select picklists offer the flexibility of choosing one or more values simultaneously, enhancing data selection capabilities.

The screenshot shows the Salesforce Object Manager interface. At the top, there are navigation tabs: Setup, Home, and Object Manager. Below the tabs, the path is shown as SETUP > OBJECT MANAGER. The main area displays a table for creating a new object named 'Product'. The left sidebar lists various configuration tabs: Details, Fields & Relationships (which is selected), Page Layouts, Lightning Record Pages, Buttons and Links, Compact Layouts, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. In the main table, the 'Fields & Relationships' section contains a 'Type' column with several options: Geolocation, Number, Percent, Phone, Picklist, Picklist (Multi-Select) (which is highlighted with a red box), Text, Text Area, Text Area (Long), Text Area (Rich), Text (Encrypted), and URL. To the right of each type, a brief description is provided. On the far right, a preview panel shows a field labeled 'Macaron Flavors' containing the value 'Gingerbread;Strawberry;Chocolate'.

Formulas and Validations - Use Formula Fields

- Users need to access and understand this data at a glance without doing a bunch of calculations in their heads.
- Enter formula fields, the powerful tool that gives you control of how your data is displayed.

Example:

- You wanted to take two numeric fields on a record and divide them to create a percentage.
- You want to turn a field into a clickable hyperlink for easy access to important information from a record's page layout.
- Maybe you want to take two dates and calculate the number of days between them.

Formulas and Validations - Use Formula Fields

Find the Formula Editor:

You can create custom formula fields on any standard or custom object.

1. From Setup, open the **Object Manager** and click **Object**.
2. In the left sidebar, click **Fields & Relationships**.
3. Click **New**.
4. Select **Formula** and click **Next**.
5. In **Field Label**, type **Annual Tax** as Field. Notice that **Field Name** populates automatically.
6. Select the **type of data** you expect your **formula to return**.

For example, if you want to write a formula that calculates the **Annual tax** a Account has to pay from his Annual Revenue. you select **Currency**.

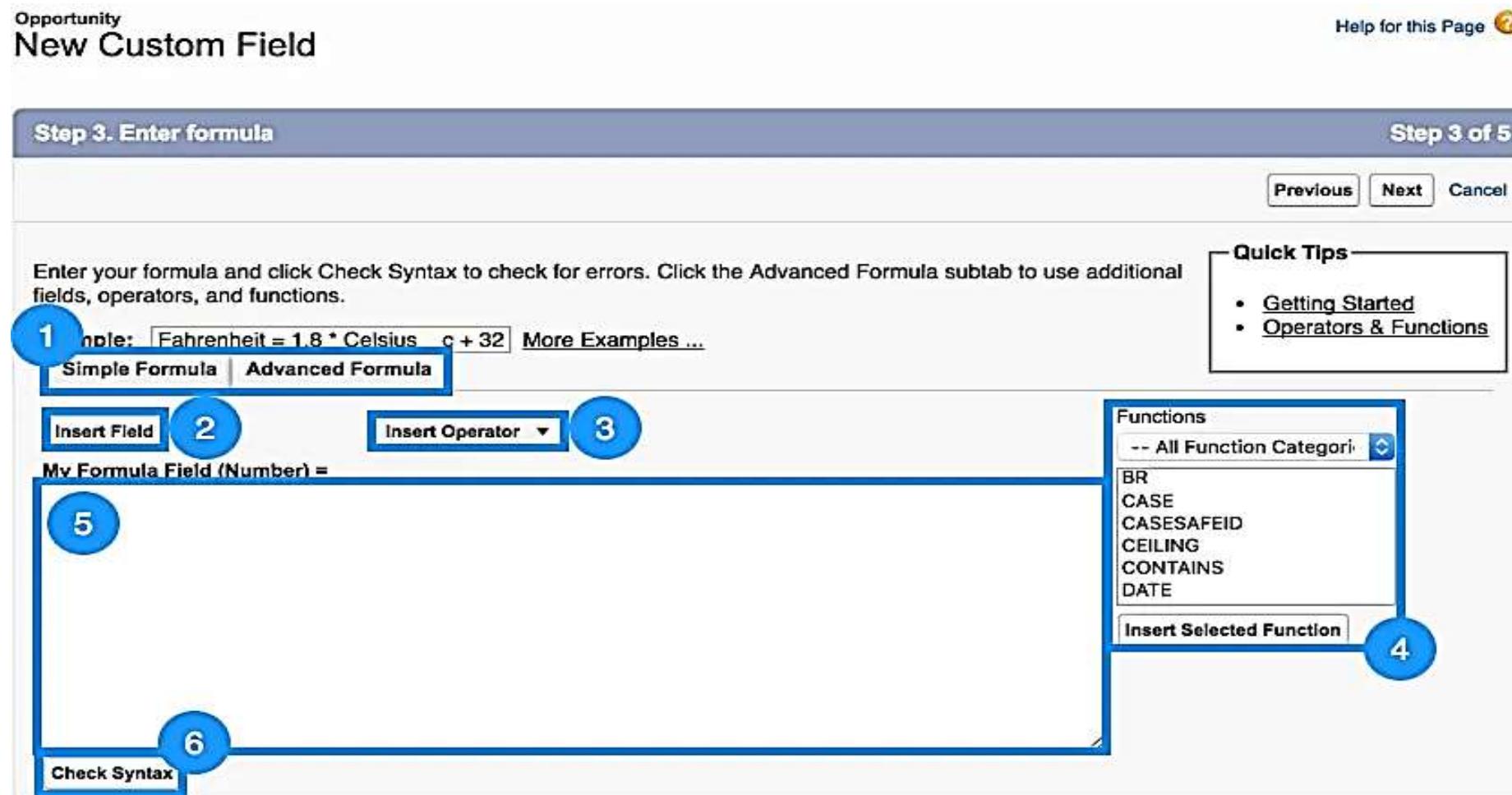
7. Click **Next**. You've arrived at the formula editor!



Formulas and Validations - Use Formula Fields

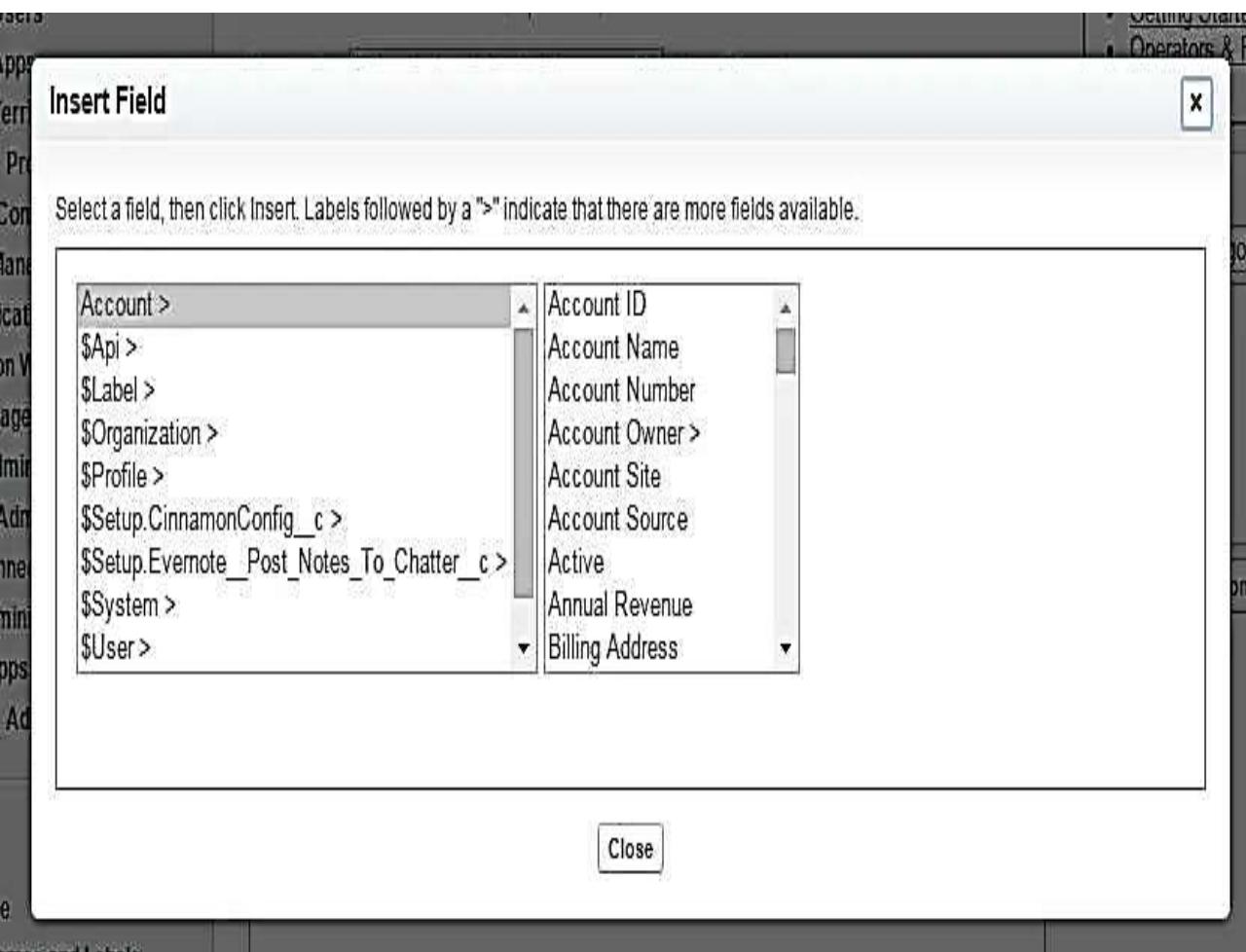
Use the Formula Editor:

1. The formula editor comes in two flavors: **Simple** and **Advanced**. It is always recommended using the Advanced editor. It means more tools for you to create powerful formulas.



Formulas and Validations - Use Formula Fields

2. The **Insert Field** button opens a menu that allows you to **select fields to use** in your formula. Inserting from this menu automatically generates the correct syntax for accessing fields.



EXAMPLE:

Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula subtab to use additional fields, operators, and functions.

Example: Gross Margin = Amount - Cost _c [More Examples...](#)

Simple Formula Advanced Formula

Select Field Type Insert Field

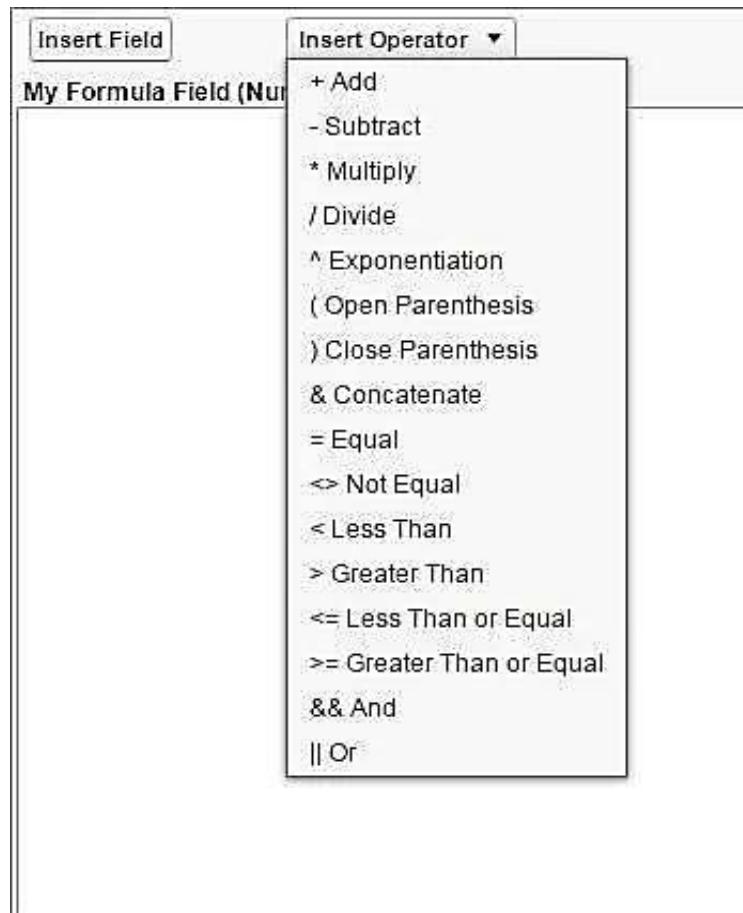
Account

Annual Tax (Currency) Annual Revenue

Billing Latitude
Billing Longitude
Employees
Number of Locations
Shipping Latitude
Shipping Longitude

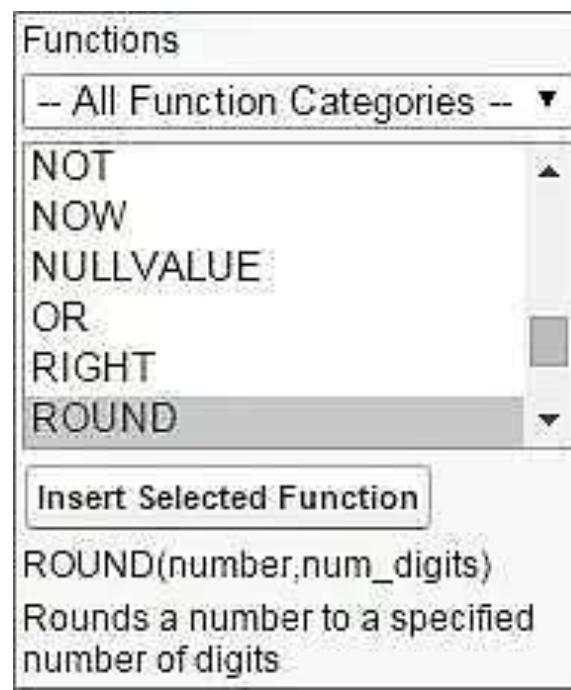
Formulas and Validations - Use Formula Fields

3. The **Insert Operator** button opens a dropdown list of the available mathematical and logical operators. For example- (AnnualRevenue * 10) / 100



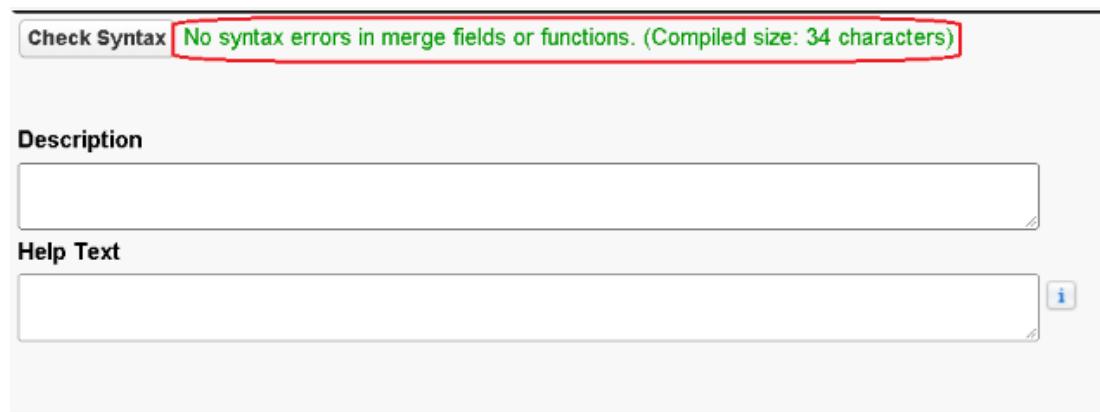
Formulas and Validations - Use Formula Fields

4. The Functions menu is where you view and insert formula functions. Functions are more complicated operations that are **pre-implemented** by Salesforce. Some functions can be used as-is (**for example**, the **TODAY()** function returns the current date), while others require extra pieces of information, called **parameters**. The **LEN(text)** function, for instance, finds the length of the text you input as a parameter.



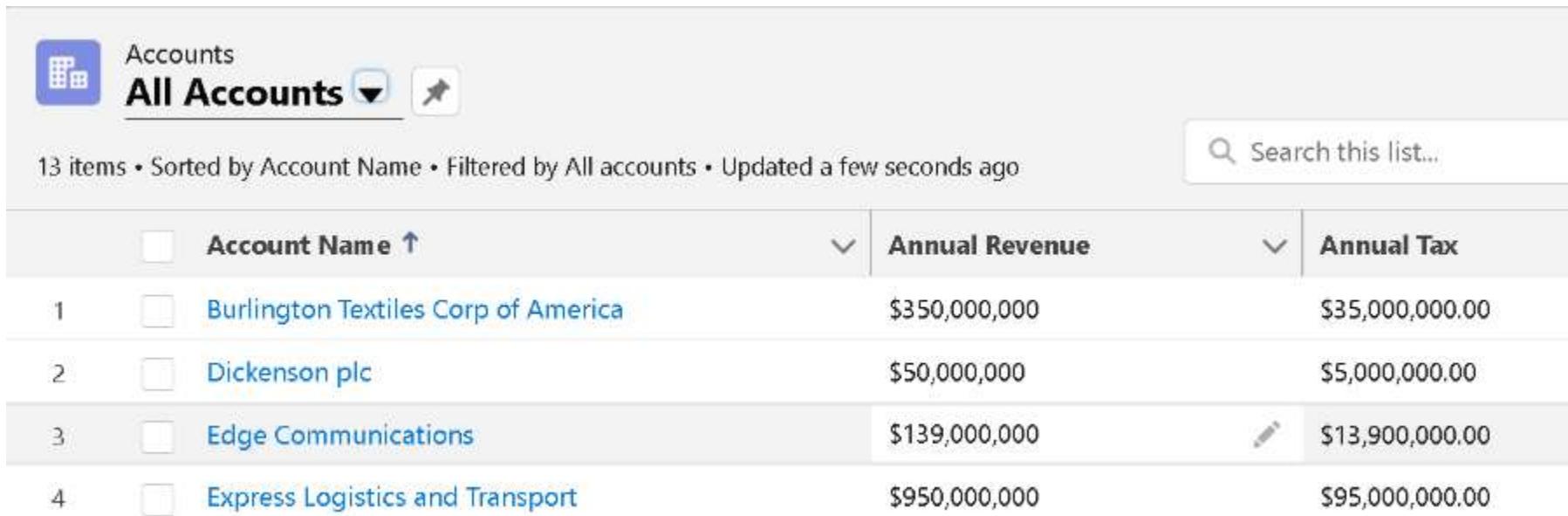
Formulas and Validations - Use Formula Fields

5. The text area is where you enter your formula. When writing formulas, keep in mind that:
- **Whitespace doesn't matter.** You can insert as many spaces and line breaks as you want without affecting the formula's execution.
 - **Formulas are case sensitive.** Pay attention to capitalization of field and object names.
 - When working with numbers, **the standard order of operations applies.**
6. Once you've written a formula, you can use the **Check Syntax** button to ensure that everything is in working order before saving. If your formula has issues, the **syntax checker alerts you to specific problems.**



Formulas and Validations - Use Formula Fields

Now when you see records of the Account Object,
We can see the “Annual Tax” is calculated as 10% of the Annual Revenue.



The screenshot shows the Salesforce interface for managing accounts. At the top, there's a blue header bar with the title "Accounts" and a sub-header "All Accounts". Below the header is a search bar with the placeholder "Search this list...". The main area displays a table with four columns: "Account Name", "Annual Revenue", and "Annual Tax". The "Account Name" column includes a sorting icon. The table lists four accounts:

	Account Name ↑	Annual Revenue	Annual Tax
1	Burlington Textiles Corp of America	\$350,000,000	\$35,000,000.00
2	Dickenson plc	\$50,000,000	\$5,000,000.00
3	Edge Communications	\$139,000,000	\$13,900,000.00
4	Express Logistics and Transport	\$950,000,000	\$95,000,000.00

Formulas and Validations - Use Formula Fields

Another Example

Formula:

Days To Close = Close Date – Today()

The screenshot illustrates the integration of formulas and validations across different Salesforce components:

- Opportunities View:** Shows a list of opportunities with columns for Opportunity Name, Amount, Expected Revenue, Days to Close, and Close Date. The Days to Close column is highlighted with a blue border. A blue arrow points from the "Days to Close" formula in the Opportunities Page Layout to this column.
- Page Layouts:** Shows the "Days to Close" field on the Opportunity page layout, also highlighted with a blue border.
- List Views:** Shows a list view of opportunities with columns for Opportunity Name, Amount, and Days to Close. The Days to Close column is highlighted with a blue border.
- Reports:** Shows a report with columns for Opportunity Name, Amount, and Days to Close. The Days to Close column is highlighted with a blue border.

Data from Opportunities View:

Opportunity Name	Amount	Expected Revenue	Days to Close	Close Date
BettaBurger	\$423,100.00	\$253,860.00	7	5/25/2017
		\$97,000.00	13	5/31/2017
		\$14,000.00	21	6/8/2017
		\$163,450.00	22	6/9/2017
		\$30,950.00	55	7/12/2017

Data from List View:

Opportunity Name	Amount	Days to Close
BettaBurger	\$423,100.00	7
Cloud Kicks	\$194,000.00	13
DreamHouse Realty	\$140,000.00	21
Ursa Major Solar	\$233,500.00	22
Get Cloudy Consulting	\$309,500.00	55

Data from Report:

Opportunity Name	Amount	Days to Close
BettaBurger	\$423,100.00	7
Cloud Kicks	\$194,000.00	13
DreamHouse Realty	\$140,000.00	21
Ursa Major Solar	\$233,500.00	22
Get Cloudy Consulting	\$309,500.00	55

Formulas and Validations - Roll Up Summary Fields

- *Rollup Summary field* is a **read only field that aggregates data from a child object to a parent object** that shares a Master-Detail relationship.
- **Roll Up Summary Field can be created in One(Master) Object**
- Rolling up the **child records and computing the value on the Parent record.**
- Roll-Up Summary field **working on the child object.**
- Rollup Summary Field **cannot be created in lookup relationship.**

Formulas and Validations - Roll Up Summary Fields

Roll-up summary field has the following **functions**:

- 1. Count:** Used to count the number of child records that exist for the parent record.
- 2. Sum:** Calculates the sum of the values of a specified field on a child record. Number, currency, and the percent fields are available for sum calculations. As ‘Sum’ is selected, you must provide a ‘Field to Aggregate’. It contains the list of numeric fields in the Child object. It will calculate the sum of the Quantity of Opportunity Line Items and put it on the rollup field.
- 3. Min:** Used to calculate the field’s minimum value in all child records.
- 4. Max:** Used to calculate the maximum value of a field in all child records.

Formulas and Validations - Roll Up Summary Fields

Roll-up summary fields are only available for objects in a Master-Detail relationship, and are not available for those that have a Lookup relationship.

Master-detail Relationship

BRANCH

(One) Master Object

**Roll Up Summary Field can be created in
One(Master) Object**

STUDENT

(Many) Detail Object



Formulas and Validations - Roll Up Summary Fields

Branch (Master)

B. Id	BName
B1	CSE
B2	AIML
B3	DS

Student (Detail)

S. Id	S.Name	FEE	Year	Rel-B.Id
S-01	Student1	50000	II	B2
S-02	Student2	45000	III	B1
S-03	Student3	48000	III	B1

After Roll up Summary Field created on Master(One) Object

Branch (Master)

B. Id	BName	Rel-S. Id	SCount
B1	CSE	S-02 , S-03	2
B2	AIML	S-01	1

Formulas and Validations - Roll Up Summary Fields

Roll-up summary fields are only available for objects in a Master-Detail relationship, and are not available for those that have a Lookup relationship.

EXAMPLE

Master-detail Relationship

Book (One) Master Object



Book History (Many) Detail Object

**Roll Up Summary Field can be created in
One(Master) Object**

Formulas and Validations - Roll Up Summary Fields

BOOK History

BH.No	B.Name	Iss Date	Ret Date	Rel-BH.No
H-01	Python	21/4/24	28/4/24	B2
H-02	C++	12/6/24	28/6/24	B1
H-03	Python	22/7/24	30/7/24	B2

BOOK

B.No.	BName	Price	Rel-BH.No
B1	C++	1200	H-02
B2	Python	1700	H-01

After Roll up Summary Field created on Master(One) Object

BOOK

B.No.	BName	Price	Rel-BH.No	Book Count
B1	C++	1200	H-02	1
B2	Python	1700	H-01	2

Formulas and Validations - Validation Rules

- Validation rules **verify that the data a user enters** in a record **meets the standards** you specify before the user can save the record.
- A validation rule can **contain a formula or expression that evaluates the data** in one or more fields and **returns a value of “True” or “False”**.
- Validation rules also **include an error message** to display to the user **when the rule returns a value of “True” due to an invalid value**.

Formulas and Validations - Validation Rules

After you have defined validation rules:

- The **user chooses to create a record** or edit an existing record.
- The user clicks **Save**.
- All validation rules are verified.
- If all **data is valid**, the **record is saved**.
- If any **data is invalid**, the **associated error message displays** without saving the record.
- The user **makes the necessary changes** and clicks **Save again**.

Formulas and Validations - Validation Rules

Creating a Validation Rule

1. From Setup, go to **Object Manager** and click **Account**.
2. In the left sidebar, click **Validation Rules**.
3. Click **New**.
4. Enter the following properties for your validation rule:
 - a. Rule Name: **Account_Number_8_Characters**
 - b. Error Condition Formula: **LEN(AccountNumber) < > 8**
5. Error Message: **Account number must be 8 characters long.**
6. To check your formula for errors, click **Check Syntax**.
7. Click **Save** to finish.

Formulas and Validations - Validation Rules

The screenshot shows the Salesforce Setup interface for managing the Account object. The top navigation bar indicates the path: SETUP > OBJECT MANAGER. The main title is "Account". On the left, a sidebar lists various configuration options: 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, Hierarchy Columns, Scoping Rules, Triggers, Flow Triggers, and Validation Rules. The "Validation Rules" option is highlighted with a red oval. The right side of the screen displays the "Details" section for the Account object, which includes fields for Description, API Name (set to "Account"), Singular Label (set to "Account"), Plural Label (set to "Accounts"), Enable Reports, Track Activities, Track Field History, Deployment Status, and Help Settings (linking to the Standard salesforce.com Help Window).

SETUP > OBJECT MANAGER

Account

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

Hierarchy Columns

Scoping Rules

Triggers

Flow Triggers

Validation Rules

Details

Description

API Name
Account

Custom

Singular Label
Account

Plural Label
Accounts

Enable Reports

Track Activities

Track Field History

Deployment Status

Help Settings
Standard salesforce.com Help Window

Formulas and Validations - Validation Rules

SETUP > OBJECT MANAGER
Account

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
Hierarchy Columns
Scoping Rules
Triggers
Flow Triggers

Validation Rules
0 Items. Sorted by Rule Name

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
No items to display.				

New

Formulas and Validations - Validation Rules

Account 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 the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit [Save](#) [Save & New](#) [Cancel](#)

Rule Name: Active

Description:

Error Condition Formula

Example: [More Examples...](#)

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 ▾](#)

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

[Check Syntax](#) [Help on this function](#)

Formulas and Validations - Validation Rules

Account 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; the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit Save Save & New Cancel

Rule Name Save Save & New Cancel

Active

Description

Error Condition Formula

Example: [More Examples...](#)

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 ▾](#)

Functions

-- All Function Categories -- ▾

LEFT
LEN
LN
LOG
LOWER
PAD

[Insert Selected Function](#)

LEN(text)
Returns the number of characters in a text string

[Help on this function](#)

[Check Syntax](#)

Formulas and Validations - Validation Rules

Account 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 evaluates to true, the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit Save Save & New Cancel

Rule Name: Account_Number_8_Characters

Active:

Description:

Error Condition Formula

Example: Discount_Percent__c>0.30 [More Examples](#)

Display an error if Discount is more than 30%

If this formula expression is true, display the text defined in the Validation Message field.

Insert Field Insert Operator ▾

LEN()

Insert Field dialog:

Select a field, then click Insert. Labels followed by a ">" indicate that there are more fields available.

Account >

- \$ObjectType >
- \$Organization >
- \$Profile >
- \$System >
- \$User >
- \$UserRole >

Account ID
Account Name
Account Number
Account Owner >
Account Site
Account Source
Active
Annual Revenue
Billing Address

You have selected:
AccountNumber
Type: Text
API Name: AccountNumber

Insert

Close

LEN(text)
Returns the number of characters in a text string

Help on this function

Formulas and Validations - Validation Rules

Error Condition Formula

Example: `Discount_Percent_c>0.30` [More Examples...](#)

Display an error if Discount is more than 30%

If this formula expression is true, display the text defined in the Error Message area

Functions

`-- All Function Categories --`

`LEFT`
`LEN`
`LN`
`LOG`
`LOWER`
`LPAD`

Insert Selected Function

`LEN(text)`
Returns the number of characters in a text string

[Help on this function](#)

Insert Field

Insert Operator

- + Add
- Subtract
- * Multiply
- / Divide
- ^ Exponentiation
- (Open Parenthesis
-) Close Parenthesis
- & Concatenate
- = Equal
- <> Not Equal**
- < Less Than
- > Greater Than
- <= Less Than or Equal
- >= Greater Than or Equal
- && And
- || Or

Check Syntax

Error Message

Example: `Discount_Percent_c>0.30`

This message will appear if the formula is true

Error Message

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 [i](#)

Formulas and Validations - Validation Rules

Rule Name:

Active:

Description:

Error Condition Formula

Example: [More Examples...](#)

Display an error if Discount is more than 30%

If this formula expression is true, display the text defined in the Error Message area

`LEN(AccountNumber) <> 8`

Functions: [-- All Function Categories --](#)

- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

ABS(number)
Returns the absolute value of a number, a number without its sign

[Help on this function](#)

Check Syntax: No errors found

Error Message

Example:

This message will appear when Error Condition formula is true

Error Message:

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 [i](#)

Formulas and Validations - Validation Rules

Edit Pyramid Construction Inc.

Review the errors on this page.

Account number must be 8 characters long.

* Account Name

Pyramid Construction Inc.

Phone

(014) 427-4427

Parent Account

Search Accounts

Fax

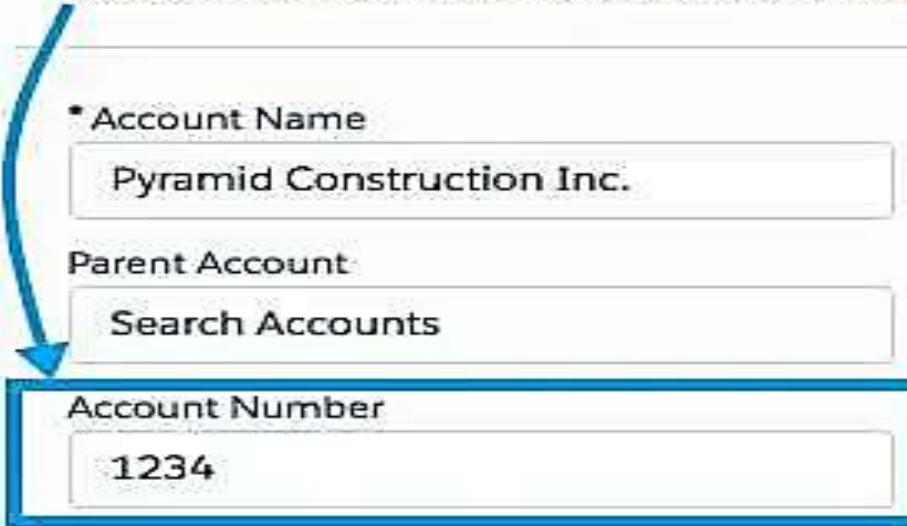
(014) 427-4428

Account Number

1234

Website

www.pyramid.com



A blue curved arrow originates from the validation message "Account number must be 8 characters long." and points directly to the "Account Number" input field, which contains the value "1234".

Salesforce String functions

Here are some frequently used string methods in Salesforce:

1. **toUpperCase()**

- The ‘toUpperCase()’ method transforms all characters in a text string into their uppercase equivalents.

EXAMPLE: If the given string is “**salesforce**”, applying the ‘**toUpperCase()**’ method will result in “**SALESFORCE**”.

2. **toLowerCase()**

- The ‘toLowerCase()’ method performs the reverse of toUpperCase(). It converts each character within a string to its respective lowercase counterpart.

EXAMPLE: If you have a string “**SALESFORCE**”, employing the ‘**toLowerCase()**’ method will yield “**salesforce**”.

Salesforce String functions

3. contains()

- The ‘contains()’ method verifies if a string includes a **specific substring**.
- The method provides a **boolean value: true** when the **substring exists**, and **false** when **it does not**. This function is beneficial for locating particular text within a more extensive string.
- You can create search functionality within your Salesforce applications.
- Check for the “@” symbol in **Email Validation**.

EXAMPLE:

```
String myProductName1 = 'HCL';
```

```
String myProductName2 = 'NAHCL';
```

```
Boolean result = myProductName2.contains(myProductName1);
```

```
System.debug('O/p will be true as it contains the String and Output is:'+result);
```

Salesforce String functions

4. trim()

- The ‘trim()’ method erases whitespace at both the starting and ending points of a text string.
- This function is especially valuable for cleaning up user input, such as eradicating extra spaces before storing data in Salesforce. It cleans up data before saving it to Salesforce.

5. substring()

- The ‘substring()’ method retrieves a segment of a string based on the provided start and end index values.
- This function is handy for decomposing strings into smaller parts or extracting specific information from a larger text.

EXAMPLE: Verify the **domain** extension in **Email validation**.

Salesforce String functions

6. replace()

- The ‘replace()’ method replaces a specific substring with another substring within the original string.
- This method is useful for **correcting misspellings** or **updating specific text** within a string.

EXAMPLE: You can **update a query** with new criteria or filters based on user input.

7. replaceAll()

- The ‘replaceAll()’ method is similar to ‘replace()’, but it replaces all occurrences of the specified substring with another substring.

Salesforce String functions

8. equals()

- This method will return true if the given string and the string passed in the method have the same binary sequence of characters and they are not null.
- This method is case-sensitive.
- You can **compare the SFDC record id** as well using this method.

EXAMPLE:

```
String myString1 = 'MyString';
```

```
String myString2 = 'MyString';
```

```
Boolean result = myString2.equals(myString1);
```

```
System.debug('Value of Result will be true as they are same and Result is:'+result);
```

Salesforce Application

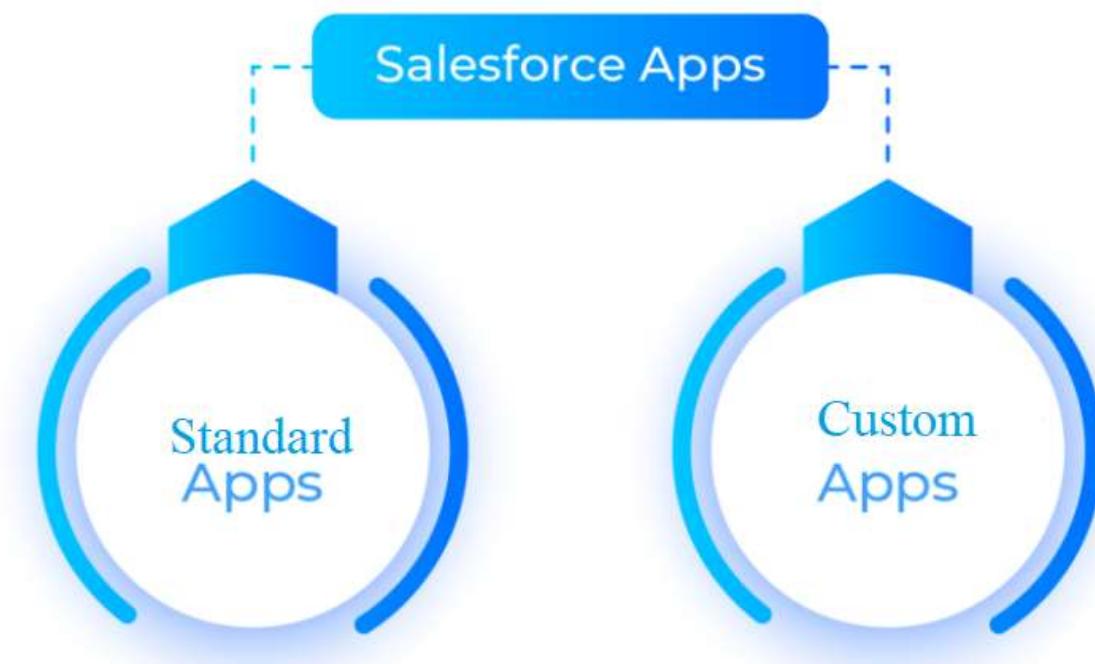
Application, commonly referred to as App, is a **particular tool or program used to complete a specific set of functions.**

We can use as a container to **store all the tabs and objects along with their functionality.**

Various tabs form a group that eventually **works together and performs multiple responsibilities** of the app.
An App simply consists of 3 major components,

1. Logo
2. Name
3. Ordered Set of Tabs

We have two types of apps in Salesforce,



Standard Apps

Standard apps come with **every occurrence of Salesforce as default**. These apps are already included in the Salesforce instant, and we get default access to them.

Users have a **little bit of control over making necessary changes** and get the best output. Thus, you get relevant results depending on the requirements of your organization.

However, some aspects like **description, label, and logo are fixed**, and we can't change them.

Example- App Launcher, Sales, Community, Salesforce Chatter, Marketing, etc.

Custom Apps

Custom apps are created according to the **needs of a company**. Custom Apps are user-oriented and are designed to fulfill specific demands of a business or organization.

On top of it, we can **combine standard and custom tabs to create a unique app** accordingly.

The best thing is that we can **change the logo and other details** in the future if necessary.

We can see a list of both standard and custom apps on the platform by clicking on the **“App Launcher”** on the Salesforce home page.

Setup

Home Object Manager

Search Setup

App Launcher

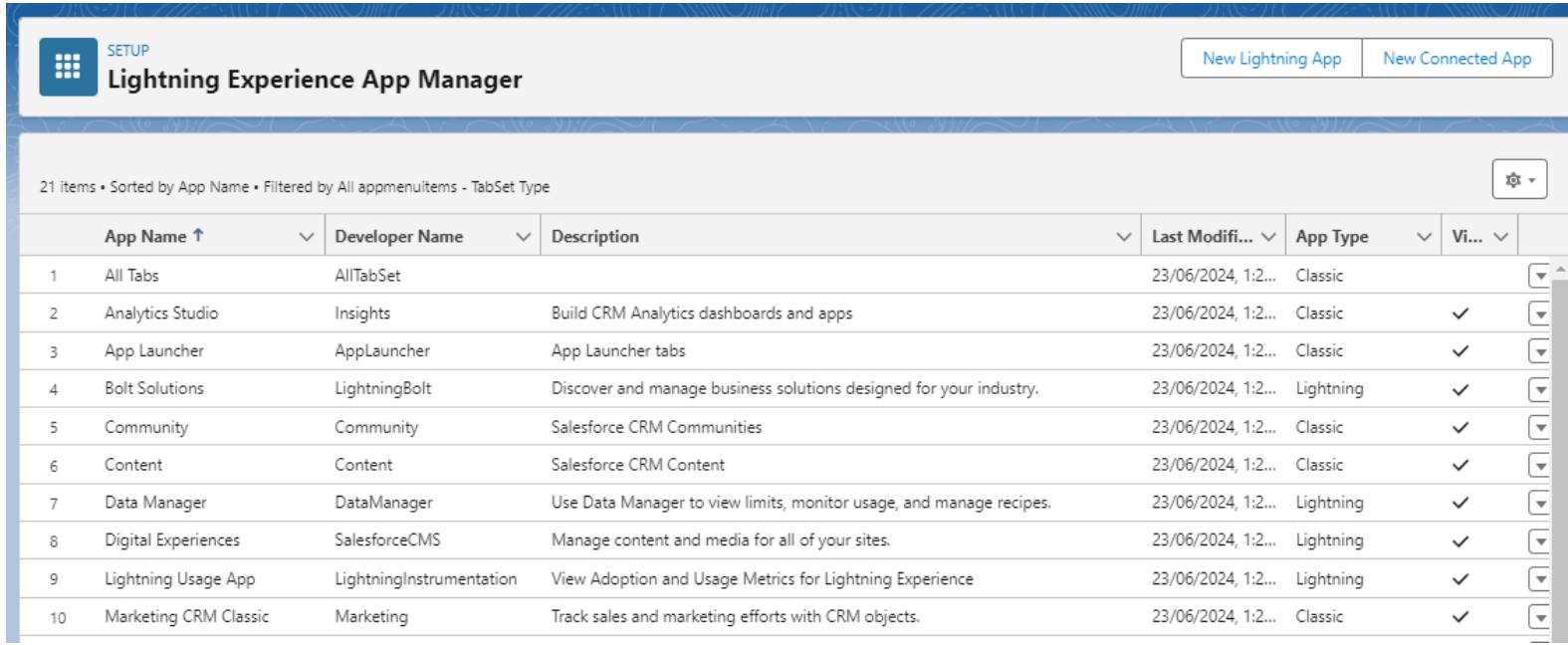
Search apps or items...

All Apps

Service Manage customer service with accounts...
Marketing Best-in-class on-demand marketing automation
Community Salesforce CRM Communities
Salesforce Chatter The Salesforce Chatter social network, including...
Content Salesforce CRM Content
Sales Console (Lightning Experience) Lets sales reps work with...
Service Console (Lightning Experience) Lets support agents work...
Sales Manage your sales process with accounts, leads...
Lightning Usage App View Adoption and Usage Metrics for Lightning...
Salesforce CMS Manage content and media for all of your sites
Salesforce Scheduler Setup Set up personalized appointment scheduling.
Bolt Solutions Discover and manage business solutions...
Test

Steps to Create a Custom Salesforce app

1. Search for “apps” on the “Quick Find” box and **select “App Manager.”**



The screenshot shows the "Lightning Experience App Manager" page. At the top, there is a header with the title "Lightning Experience App Manager" and two buttons: "New Lightning App" and "New Connected App". Below the header, a table lists 21 items, sorted by App Name. The columns in the table are: App Name, Developer Name, Description, Last Modified, App Type, and Visibility. The table includes rows for various built-in apps like All Tabs, Analytics Studio, App Launcher, Bolt Solutions, Community, Content, Data Manager, Digital Experiences, Lightning Usage App, and Marketing CRM Classic.

App Name ↑	Developer Name	Description	Last Modifi...	App Type	Vi...
1 All Tabs	AllTabSet		23/06/2024, 1:2...	Classic	✓
2 Analytics Studio	Insights	Build CRM Analytics dashboards and apps	23/06/2024, 1:2...	Classic	✓
3 App Launcher	AppLauncher	App Launcher tabs	23/06/2024, 1:2...	Classic	✓
4 Bolt Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	23/06/2024, 1:2...	Lightning	✓
5 Community	Community	Salesforce CRM Communities	23/06/2024, 1:2...	Classic	✓
6 Content	Content	Salesforce CRM Content	23/06/2024, 1:2...	Classic	✓
7 Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	23/06/2024, 1:2...	Lightning	✓
8 Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	23/06/2024, 1:2...	Lightning	✓
9 Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	23/06/2024, 1:2...	Lightning	✓
10 Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	23/06/2024, 1:2...	Classic	✓

2. It leads you to the Lightning Experience App Manager. Here, you can choose the **“New Lightning App”** option to begin the custom app creation.

3. On the window that pops up, enter the **app name** (**e.g. University Management**), developer name, and description.

We can also pick a color theme for the app or upload an image containing the app logo in this step.

App Details & Branding

Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.

App Details

* App Name

* Developer Name

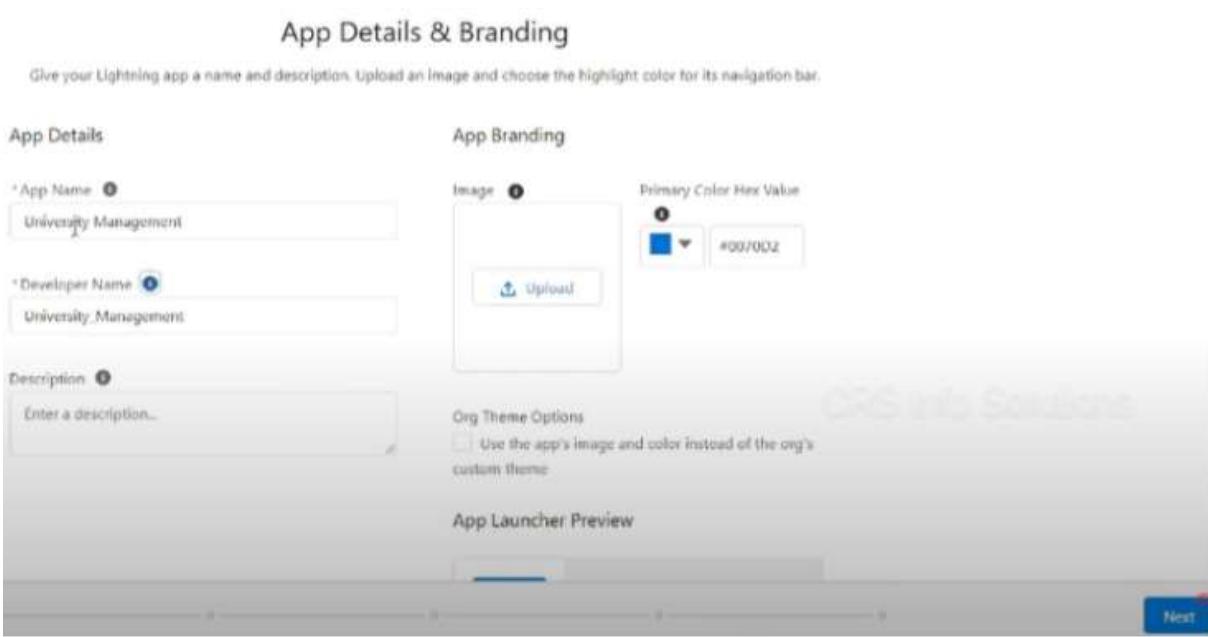
Description

App Branding

Image Primary Color Hex Value

Org Theme Options Use the app's image and color instead of the org's custom theme

App Launcher Preview



New Lightning App

Standard navigation shows items in a navigation bar at the top of the page. Console navigation shows a list of records in a sidebar and opens each record in a new workspace tab. Also, you can select the form factors where this app is available.

Navigation and Form Factor

* Navigation Style
 Standard navigation
 Console navigation

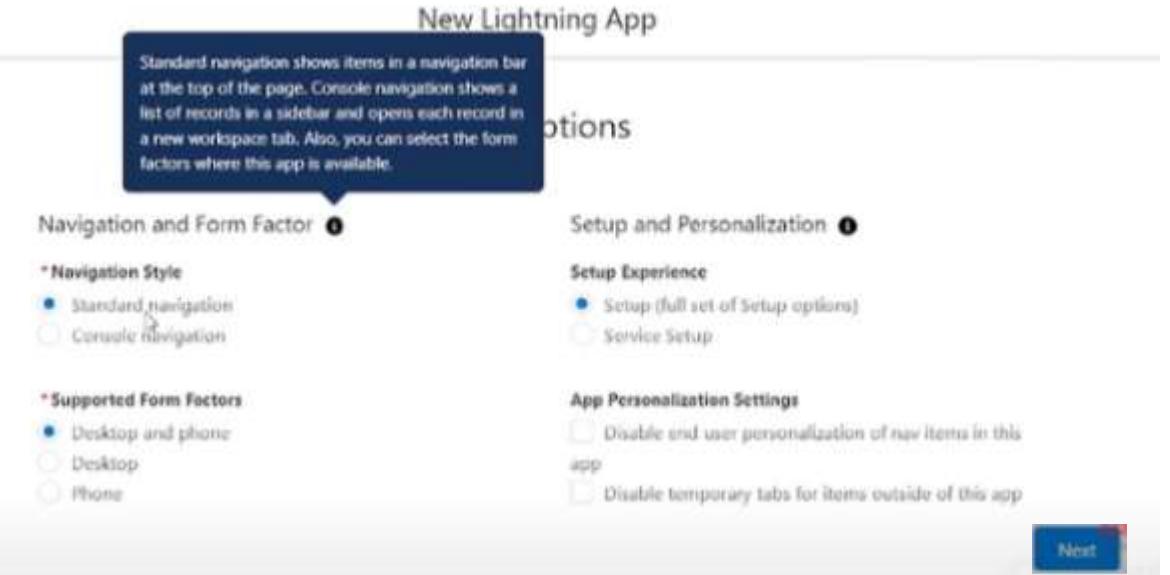
* Supported Form Factors
 Desktop and phone
 Desktop
 Phone

Setup and Personalization

Setup Experience
 Setup (full set of Setup options)
 Service Setup

App Personalization Settings

Disable end user personalization of nav items in this app
 Disable temporary tabs for items outside of this app

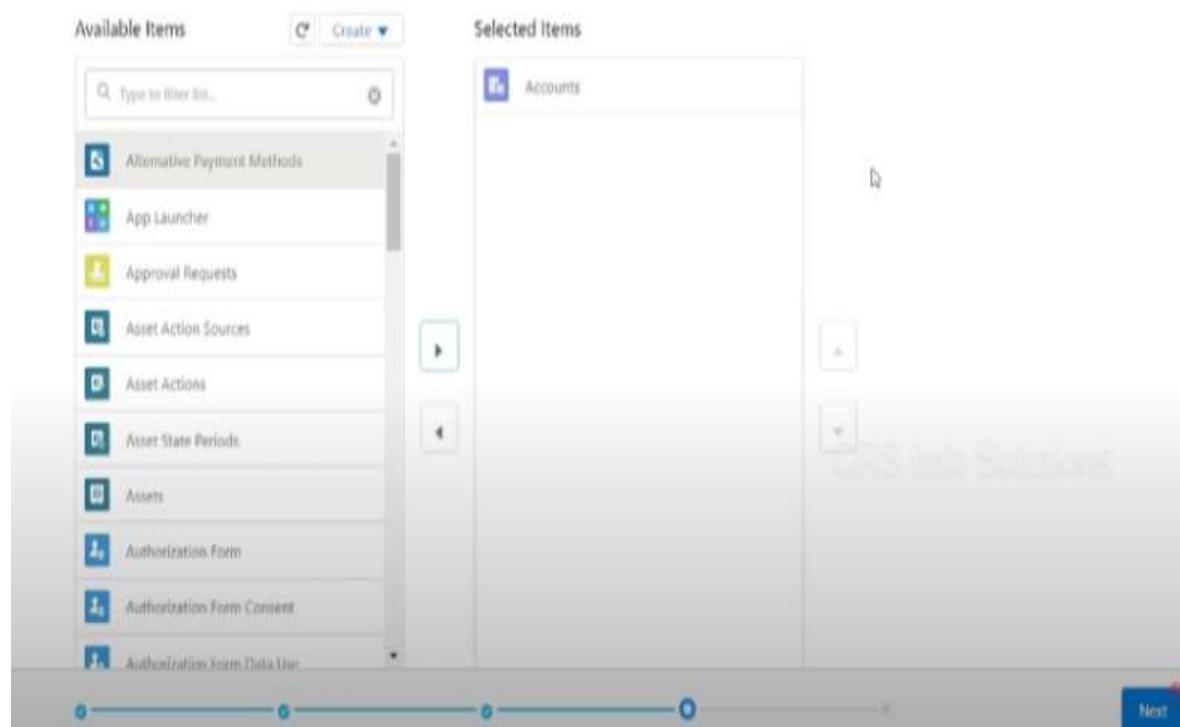


5. Now, click “Next” to go to the “**Utility Items**” page. You can keep **it as default** and move to the next page by click on to the “**Next**” button.

4. Then, click the “**Next**” button to start **setting up app options**. We can select the preferred navigation style, supported form factors, setup experience, and app personalization settings. For now, you can go with the **default options** provided here in this step.

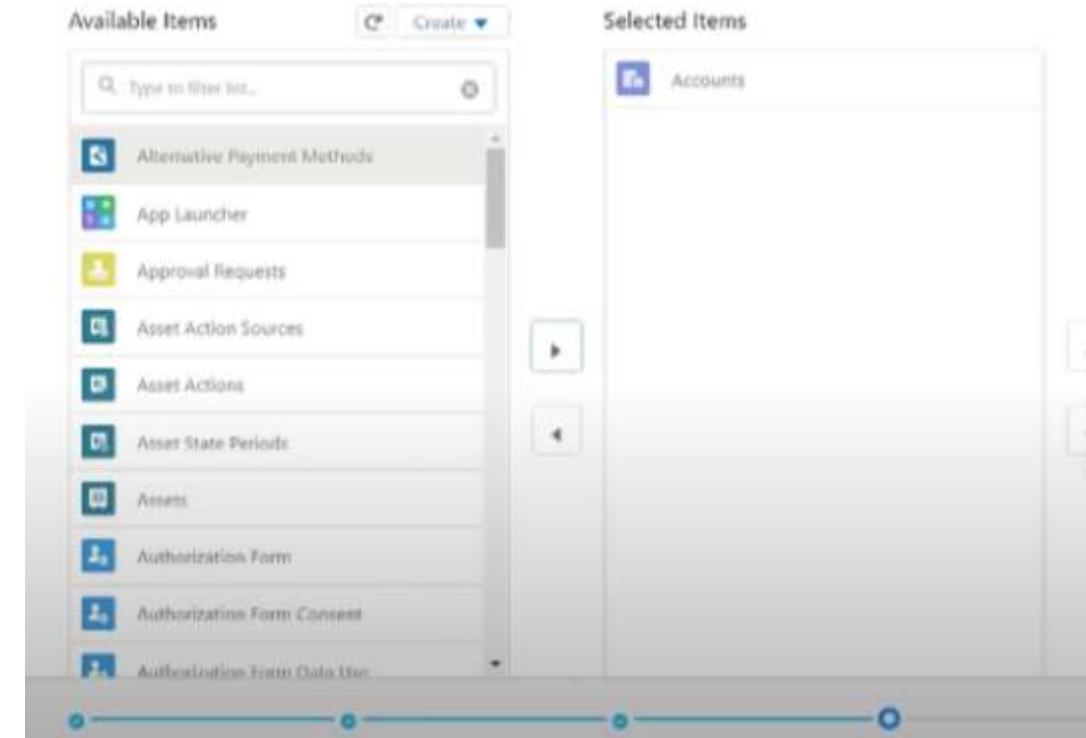
New Lightning App

Consider using your favorite search or filtering tools to quickly find the items you want to include in your app. You can also use the search bar at the top.



New Lightning App

Use only the "provider" or "target" part of each object's name to copy and paste it into the "Available Items" field. You can also use the search bar at the top.



6. Here, we have to choose a set of items to include in the app. The **left side of the page** shows a list of **standard objects** Salesforce provides, such as Accounts and Assets. For example, let's simply pick the Accounts object to add to the new app (University Management).

7. Then, go to the next page to select profiles for your app. You can search for the **default “System Administrator”** profile and add it to the app

8. Next, **save and finish** the app creation process. Now we can **view this new app (University Management) on the App Launcher**

