Table of Contents

Introduction to CRM 2

1.0 Introduction to Cloud Computing 3
1.1 What is Cloud Computing? 3
1.2 Top benefits of cloud computing 3
1.3 Types of cloud computing 3
1.4 Types of cloud services 4

2.0 What is CRM? 6

3.0 What is Salesforce? 6
3.1 Salesforce Economy 7
3.2 Why Learn Salesforce 7
3.3 Salesforce Products 10

4.0 Understand the Salesforce Architecture 13
4.1 Why Trust the Cloud? 13
4.2 Sharing Is Caring in the Multitenant Cloud 14
4.3 The Magic of Metadata 14
4.4 All About That API 14

5.0 Salesforce Editions 15

6.0 Getting Started with Salesforce 16

7.0 Sales Cloud Topics in Salesforce CRM 19
7.1 How to create New Lead in Salesforce 19
7.2 How to convert a Lead to opportunity in Salesforce 22
7.3 How to create web to lead forms in Salesforce | Lead Management 23
7.4 Creating a new account in Salesforce 26
7.5 Creating new Contact in Salesforce 27
7.6 How to create new Opportunity in Salesforce 28
7.7 Adding product and price book to an opportunity in Salesforce 30
7.8 Salesforce Activities, creating and viewing Salesforce activities 33
7.9 Modifying Quote Template in Salesforce 34
7.10 Email a Quote in Salesforce 40
7.11 How to create a lead assignment rules in Salesforce 42
7.12 How to Create a Lead Queue in Salesforce 45

8.0 Salesforce Marketing Administration 47
8.1 Salesforce campaign management 47
8.2 Salesforce Campaign planning 48
8.3 Configuring Campaign settings in Salesforce 49
8.4 How to create new campaigns in Salesforce CRM 52

Salesforce App Builder 55

9.0 Organization Administration 56
9.1 The Salesforce setup menu 56

9.2 Establishing your company profile within Salesforce 56
9.3 License information 57
9.4 User interface 58

10.0 Configuration in Salesforce 62
10.1 Objects 62
10.2 Fields 62
10.3 Object relationships 63
10.4 Apps 63
10.5 Creating custom objects 66
10.6 Object Limits 68
10.7 Building relationship fields 73
10.8 Building formulas 73
10.9 Page layouts 74
10.10 List views 76

11.0 User Management 78
11.1 Manage Users 78

12.0 Overview of Data Security 79
12.1 Introduction 79
12.2 Levels of Data Access 80
12.3 Control Access to the Organization 81
12.4 Control Access to Objects 83
12.5 Control Access to Fields 90
12.6 Control Access to Records 92

13.0 Implementing Business: Processes in Salesforce CRM 103
13.1 Data validation rules 103
13.2 Workflow rules 105
13.3 Approval process 108
13.4 Process Builder 117
13.5 Get Started With Flows 121

14.0 Data Management 148
14.1 Data Import Wizard 148
14.2 Data Loader 148

15.0 Data Analytics with Reports and Dashboards 151
15.1 Reports 151
15.2 Dashboards 158

Introduction to CRM

1.0 Introduction to Cloud Computing

1.1 What is Cloud Computing?

Cloud computing is the delivery of computing services—servers, storage, databases, networking, software, analytics, intelligence and more—over the Internet ("the cloud") to offer faster innovation, flexible resources and economies of scale. You typically pay only for cloud services you use, helping lower your operating costs, run your infrastructure more efficiently and scale as your business needs change.

1.2 Top benefits of cloud computing

Cloud computing is a big shift from the traditional way businesses think about IT resources. Here are six common reasons organisations are turning to cloud computing services:

1. Cost

Cloud computing eliminates the capital expense of buying hardware and software and setting up and running on-site data-centres—the racks of servers, the round-the-clock electricity for power and cooling, the IT experts for managing the infrastructure. This adds up to a significant cost reduction.

2. Speed

Most cloud computing services are provided as self service and on demand, so even vast amounts of computing resources can be provisioned in minutes, typically with just a few mouse clicks, giving businesses a lot of flexibility and taking the pressure off capacity planning.

3. Global scale

The benefits of cloud computing services include the ability to scale elastically. In cloud speak, that means delivering the right amount of IT resources—for example, more or less computing power, storage, bandwidth—right when it is needed and from the right geographic location.

4. Productivity

On-site data-centres typically require a lot of "racking and stacking"—hardware set up, software patching and other time-consuming IT management chores. Cloud computing removes the need for many of these tasks, so IT teams can spend time on achieving more important business goals.

5. Performance

The biggest cloud computing services run on a worldwide network of secure data-centres, which are regularly upgraded to the latest generation of fast and efficient computing hardware. This offers several benefits over a single corporate data-centres, including reduced network latency for applications and greater economies of scale.

6. Security

Many cloud providers offer a broad set of policies, technologies and controls that strengthen your security posture overall, helping protect your data, apps and infrastructure from potential threats.

1.3 Types of cloud computing

Not all clouds are the same and not one type of cloud computing is right for everyone. Several different models, types and services have evolved to help offer the right solution for your needs. First, you need to determine the type of cloud deployment or cloud computing architecture that your cloud services will be implemented on. There are three different ways to deploy cloud services: on a public cloud, private cloud or hybrid cloud.

1. Public cloud

Public clouds are owned and operated by a third-party cloud service providers, which deliver their computing resources like servers and storage over the Internet. Amazon AWS and Microsoft Azure are an example of a public cloud. With a public cloud, all hardware, software and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser.

2. Private cloud

A private cloud refers to cloud computing resources used exclusively by a single business or organisation. A private cloud can be physically located on the company's on-site data-centre. Some companies also pay third-party service providers to host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

3. Hybrid cloud

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives your business greater flexibility, more deployment options and helps optimise your existing infrastructure, security and compliance.

1.4 Types of cloud services

Most cloud computing services fall into four broad categories: infrastructure as a service (IaaS), platform as a service (PaaS), Server-less Computing and software as a service (SaaS). These are sometimes called the cloud computing stack because they build on top of one another. Knowing what they are and how they are different makes it easier to accomplish your business goals.

1. Infrastructure as a service (IaaS)

The most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

2. Platform as a service (PaaS)

Platform as a service refers to cloud computing services that supply an on-demand environment for developing, testing, delivering and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps, without worrying about setting up or managing the underlying infrastructure of servers, storage, network and databases needed for development.

3. Server-less Computing

Overlapping with PaaS, Server-less Computing focuses on building App functionality without spending time continually managing the servers and infrastructure required to do so. The cloud provider handles the setup, capacity planning and server management for you. Server-less architectures are highly scalable and event-driven, only using resources when a specific function or trigger occurs.

4. Software as a service (SaaS)

Software as a service is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure and handle any maintenance, like software upgrades and security patching. Users connect to the application over the Internet, usually with a web browser on their phone, tablet or PC.

5. How cloud computing works

While cloud computing services all work a little differently, many provide a friendly, browser-based dashboard that makes it easier for IT professionals and developers to order resources and manage their accounts. Some cloud computing services are also designed to work with REST APIs and a command-line interface, giving developers multiple options.

6. Uses of cloud computing

You are probably using cloud computing right now, even if you don't realise it. If you use an online service to send email, edit documents, watch movies or TV, listen to music, play games or store pictures and other files, it is likely that cloud computing is making it all possible behind the scenes. The first cloud computing services are barely a decade old, but already a variety of organisations—from tiny start-ups to global corporations, government agencies to non-profits—are embracing the technology for all sorts of reasons.

Here are a few examples of what is possible today with cloud services from a cloud provider:

1. Create new apps and services

Quickly build, deploy and scale applications—web, mobile and API—on any platform. Access the resources you need to help meet performance, security and compliance requirements.

2. Test and build applications

Reduce application development cost and time by using cloud infrastructures that can easily be scaled up or down.

3. Store, back up and recover data

Protect your data more cost-efficiently—and at massive scale—by transferring your data over the Internet to an offsite cloud storage system that is accessible from any location and any device.

4. Analyse data

Unify your data across teams, divisions and locations in the cloud. Then use cloud services, such as machine learning and artificial intelligence, to uncover insights for more informed decisions.

5. Stream audio and video

Connect with your audience anywhere, anytime, on any device with high-definition video and audio with global distribution.

6. Embed intelligence

Use intelligent models to help engage customers and provide valuable insights from the data captured.

7. Deliver software on demand

Also known as software as a service (SaaS), on-demand software lets you offer the latest software versions and updates around to customers—anytime they need, anywhere they are.

2.0 What is CRM?

CRM stands for Customer Relationship Management. It helps organizations get 360 degrees view of their customers. It helps connect to customers in a whole new way. Build more meaningful and lasting relationships — better understand their needs, identify new opportunities to help, address any problems faster and deploy customer-focused apps lightning fast. With a single view of every customer interaction you can sell, service and market like never before.

CRM does the following:

- Actively tracks and manages customer information.
- Connects your entire team from any device.
- Intelligently captures customer emails.
- Simplifies repetitive tasks so you can concentrate on leads.
- Delivers instant insights and recommendations.
- Extends and customizes as your business grows.

Why CRM Matters?

- CRM helps you ditch clunky processes and manual effort so you can get on with business.
- You'll find more leads, close more deals, keep more customers and grow your business.
- One place to store all customer information means your conversations are always personal, relevant, and up to date.
- Salesforce customers report 37% more sales revenue, 45% higher customer satisfaction, 43% better marketing ROI*.

3.0 What is Salesforce?

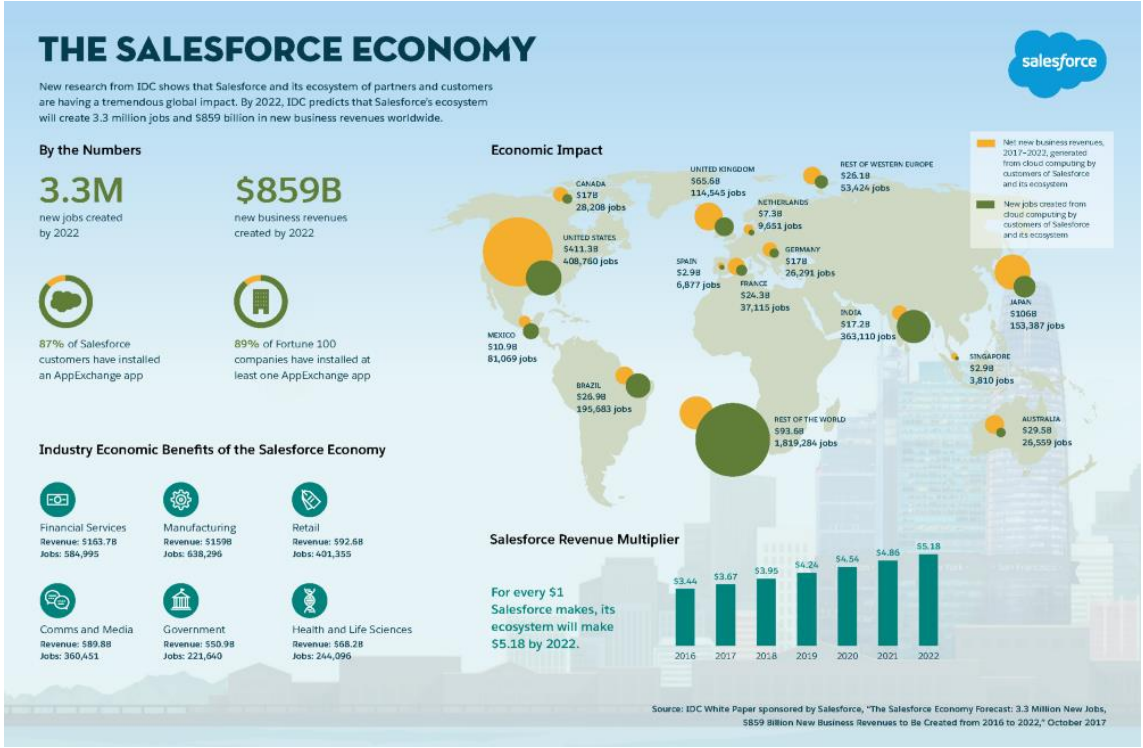
Salesforce is world's number #1 CRM platform. Salesforce began with the vision of reinventing Customer Relationship Management (CRM). Since then it has changed the way enterprise software is delivered and used, changing the industry forever. All Salesforce products run entirely in the cloud so there's no expensive setup costs, no maintenance and employees can work from any device with an internet connection – smartphone, tablet or laptop.

It makes CRM easy to use for small businesses and large-scale enterprises. This approach has helped to make Sales Cloud the world's number 1 CRM system. But Salesforce doesn't start and end with CRM for Sales and Marketing. The platform enables you to manage all interactions with customers and prospects, so that your organization can grow and succeed. That's why it is called the Customer Success Platform. It helps with the following:

- Get more leads, close more deals, and do more - faster
- Increase customer loyalty, retention, and satisfaction
- Connect 1-to-1 with customers on social, mobile, and more
- Anticipate customer wants based on past behavior
- Monitor and manage your business anytime, from anywhere

3.1 Salesforce Economy

Salesforce is world's #1 CRM. Salesforce eco-system is expected to generate 3.# Million jobs by the year 2022.

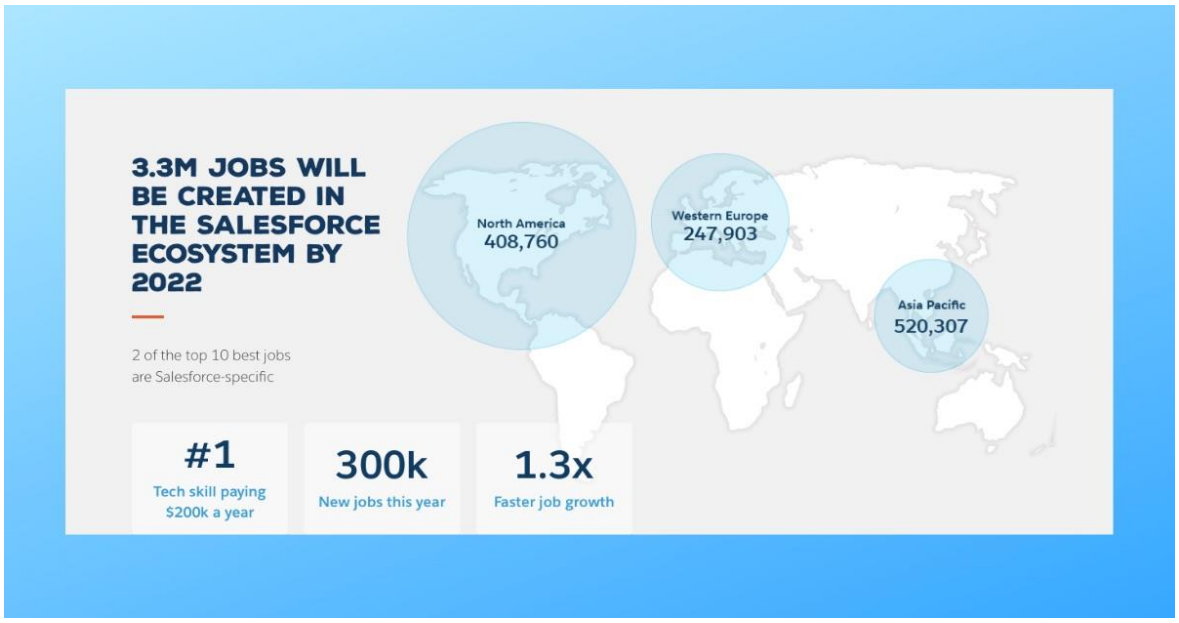


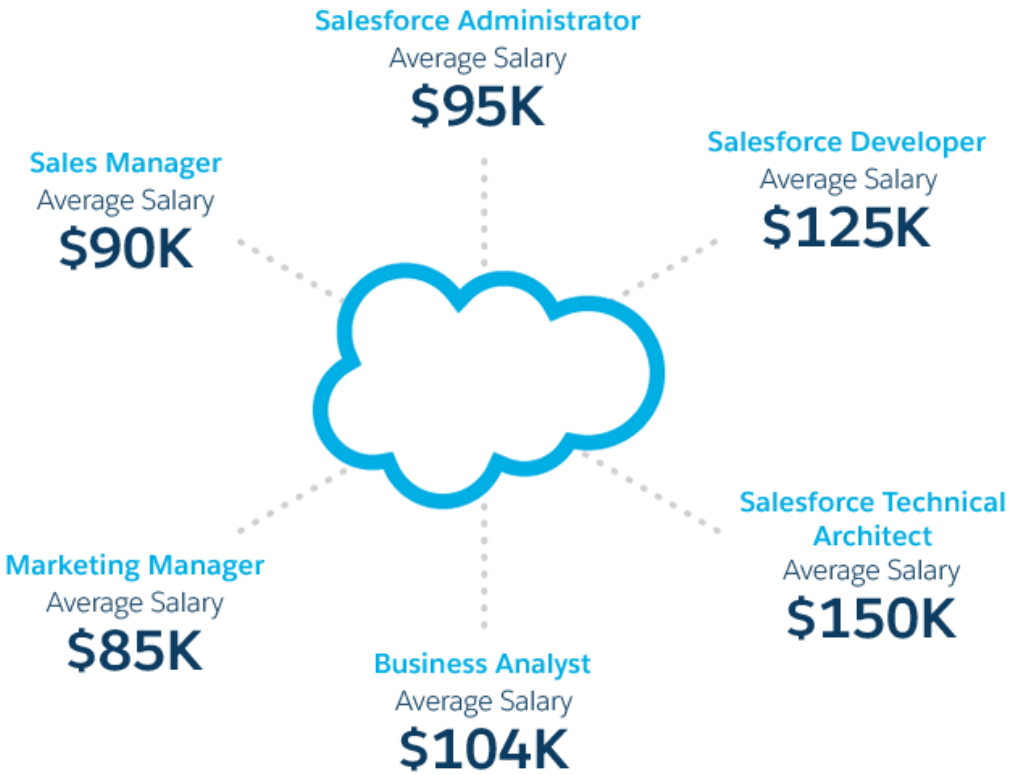
3.2 Why Learn Salesforce

It makes immense sense to learn Salesforce. Salesforce trained and certified professionals are in high demand and command premium salaries.

1. **Job Opportunities**

   Salesforce eco-system is expected to generate 3.3M jobs by the year 2022 globally. In India alone there are 4500+ jobs for Salesforce developer on the Naukri job portal. The demand for Salesforce professionals is increasing at a phenomenal rate.





2. Growth Opportunities

A career in Salesforce provides tremendous growth opportunities. One can start as a develop or an administrator to move to Business Analyst or Architect positions. Salesforce offers following career opportunities:

- Salesforce Administrator
- Salesforce Developer
- Salesforce Technical Architect
- Salesforce Business Analyst
- Salesforce Marketing Manager
- Salesforce Sales Manager
- Salesforce Consultant

3. Opportunity to work in big companies

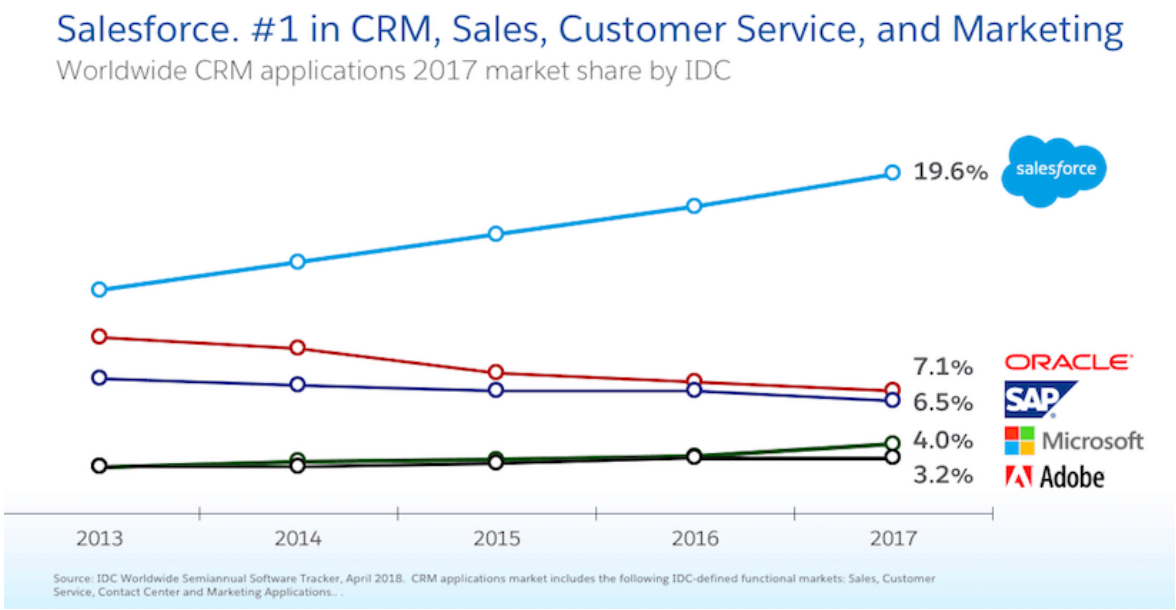
All the big companies have Salesforce as a major service practice. The top ten companies with strong Salesforce practice include: Accenture, Deloitte, IBM, PwC, Wipro, Cognizant Technologies, Capgemini, NTT Data and Tech Mahindra. These companies have huge Salesforce practice and employ thousands of Salesforce Developers, Administrators, Architects etc. They are also hiring Salesforce professionals in large scale. A Salesforce trained professional gets the opportunity to work in these large companies.

4. Global Demand

Salesforce is a global company. Salesforce trained professionals are in global demand. They get opportunities to work in different parts of the world including USA, Europe, Australia etc.

5. World' #1 CRM and growing

Salesforce is the undisputed leader in cloud based CRM. It is highly in demand and growing at a rapid rate. Many companies are switching from traditional CRM to cloud based CRM. Salesforce being the market leader becomes their first choice. While the traditional CRMs are losing market-share, Salesforce market share is increasing consistently.



6. Digital World

The world is increasingly going digital. Every company is increasing its digital footprint to stay relevant in the market. Companies that do not embrace digital technologies will get left behind. Digital technologies enable companies to reach the global market and showcase their goods and services. Marketing, Sales and Service with digital technologies enables companies to reach out to customers, understand them and delight them. There will be more and more demand for companies to embrace cloud-based platforms like Salesforce.

3.3 Salesforce Products

Salesforce has multiple products to address various requirements in an organization. These primarily consist of Sales Cloud, Service Cloud and Marketing Cloud. In addition, there are other products like App Exchange, Commerce Cloud, and Einstein etc.

Sales Cloud

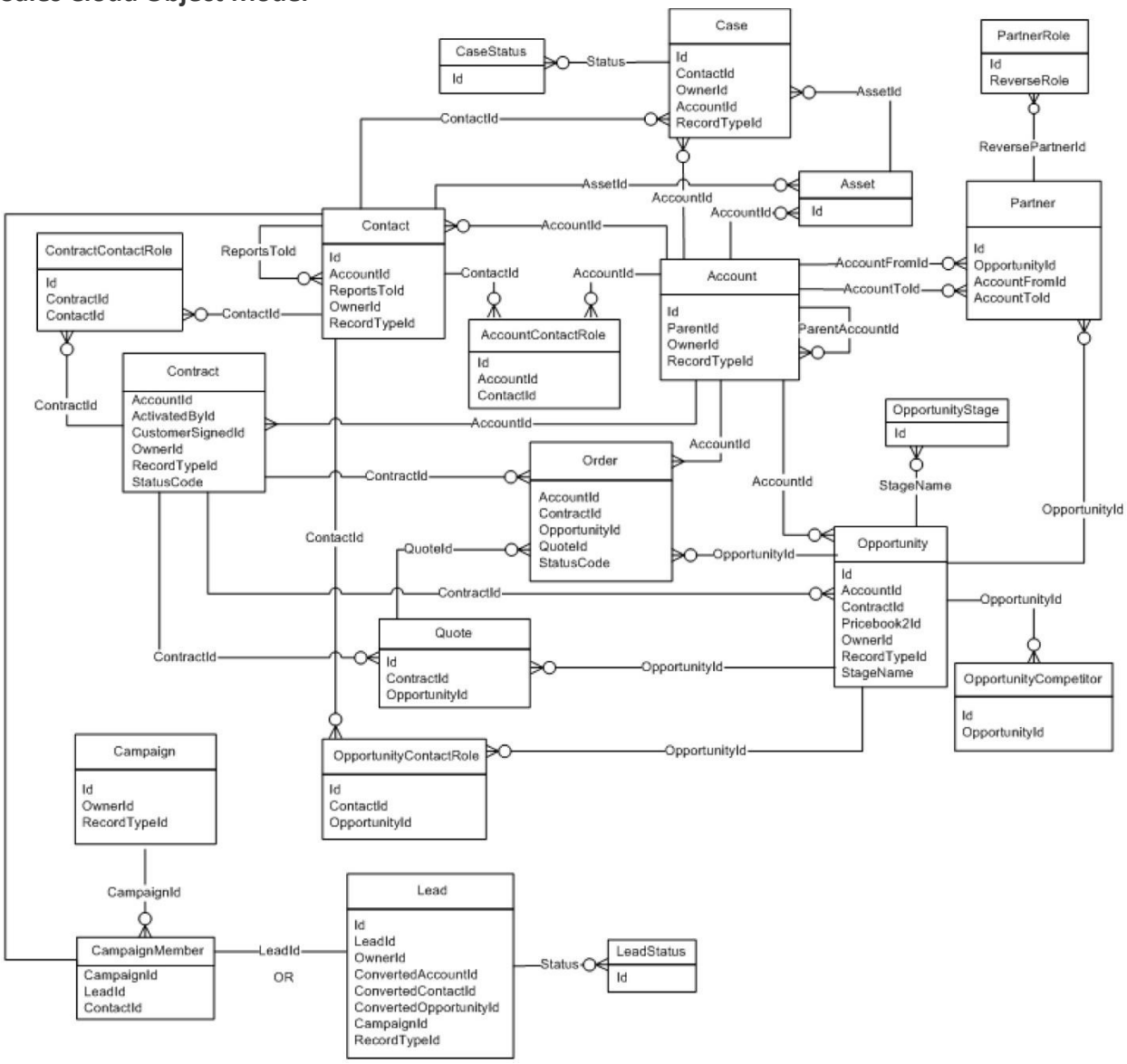
With Sales Cloud, you can grow your accounts, find new customers, and close deals faster — from anywhere. The path to success starts here.

- Close more deals, faster.
  Track all your customer information and interactions in one place, anywhere. Best practices and in-line intelligence help move deals along faster.
- Get more leads and make them sales-ready.
  Nurture leads so they're ready to convert, then automatically route them to the right reps at the right time. Build your own "lead machine" to improve conversion rates and grow your revenue.
- Accelerate sales productivity with a lightning fast console.
  Whether you're a rep in the field or you prefer to work inside your email inbox, now you can do more with AI, automatic data capture, and business process automation. All in one console.
- Make insightful decisions with up-to-date data.
  With Salesforce, information is in one place so it's easier to prioritise and make quick decisions based on the most up-to-date information. And do it all from any device.
- Give reps the power of automation and artificial intelligence.
  Make CRM data work harder for reps. Sales Cloud Einstein is artificial intelligence that learns what separates deals won from deals lost, and highlights top leads and next steps.

Sales important cloud Objects:

- **Account**
  Represents an individual account, which is an organization or person involved with your business (such as customers, competitors, and partners).
- **Contact**
  Represents a contact, which is a person associated with an account.
- **Lead**
  Represents a prospect or lead.
- **Opportunity**
  Represents an opportunity, which is a sale or pending deal.
- **Contract**
  A contract is a written agreement between parties. Many companies use contracts to define the terms for doing business with other companies.

Sales Cloud Object Model



Service Cloud

Support customers everywhere with the service platform that keeps the conversation flowing.

- Close cases faster with the help of AI and automation.
  Help your support agents solve customer problems faster by putting the right tools at their fingertips — all from one console.
- Provide faster, more intelligent self-service.
  Build self-service communities. Give your customers access to the answers they need to solve problems on their own terms and timeline.
- Personalize customer care and predict needs.
  Get more insight into your customers' behavior so you can personalize their service, predict future needs, and even provide conversational service with live messaging.
- Drive service productivity from the call center to the field.
  Improve your mobile service operations and deliver connected, intelligent, more personalized service from the phone to the field on one platform.

Marketing Cloud

- Build and manage personalized customer journeys with Marketing Cloud. Discover the complete solution to navigate the entire customer journey — on one platform. Explore the gear to help marketers deliver the right message at the right time — on any channel.
- Personalize your email marketing at scale.
  Build and manage any kind of email campaign. Use CRM and other data for personalization that drives engagement. Automate your marketing to scale your capabilities and reach customers with relevant messaging.
- Engage with mobile messaging.
  Use SMS, MMS, push notifications, and group messaging to reach customers anytime, anywhere. Go cross-channel and extend your digital marketing strategy by adding mobile to your email and social campaigns.
- Create cross-channel journeys.
  Create 1-to-1 customer journeys across email, mobile, social, ads, and the web. Connect experiences across marketing, sales, service, and every customer touchpoint.
- Guide exceptional social experiences.
  Connect social to marketing, sales, and service with social media marketing tools. Listen, engage, publish, and analyze data from over a billion sources while automating social workflow.

App Exchange

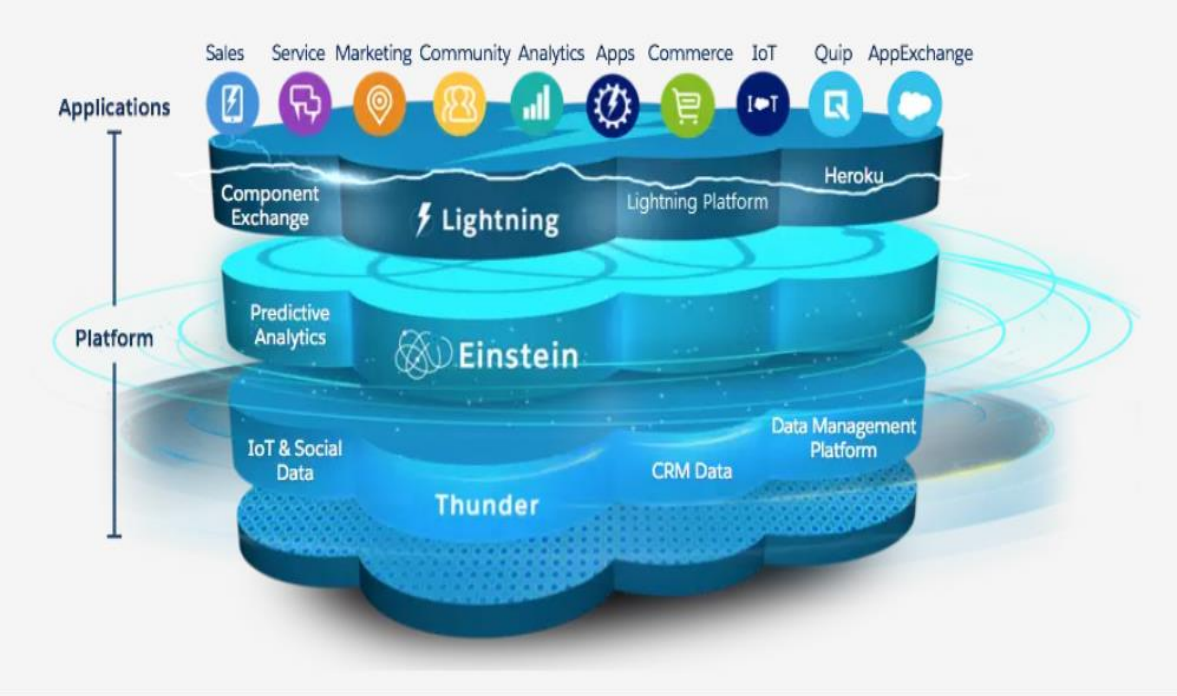
Salesforce also offers App Exchange which provides a platform for custom application building and sharing. This is essentially a marketplace where you can buy and sell Apps that work on the Salesforce platform. It has both free and paid Apps. The Apps are well categorised and offer different solutions to different industries.

4.0 Understand the Salesforce Architecture

What Is the Salesforce Architecture?

By now you know that you can use Salesforce to deliver a highly customized experience to your customers, employees, and partners. You can do it without writing much (or any) code, and you can do it fast. What's so special about Salesforce? It all starts with the architecture.

Before you close out this window in a frantic attempt to avoid learning about what seems like a really boring subject, sit tight. Learning about Salesforce architecture is quite interesting, and understanding it makes working with the platform a whole lot easier. When you think about the Salesforce architecture, imagine a series of layers that sit on top of each other. Sometimes it helps to think of it as a cake because cake is delicious, and it makes everything better.



There's a lot to unpack here, but let's focus on the most important points.

- Salesforce is a cloud company. Everything it offers resides in the trusted, multitenant cloud.
- The Salesforce platform is the foundation of services. It's powered by metadata and made up of different parts, like data services, artificial intelligence, and robust APIs for development.
- All the apps sit on top of the platform. The prebuilt offerings like Sales Cloud and Marketing Cloud, along with apps you build using the platform, have consistent, powerful functionality.
- Everything is integrated. The platform technologies like Einstein predictive intelligence and the Lightning framework for development are built into everything that Salesforce offers and everything you build
- There are few terms in here that are extra important for you to understand: trust, multitenancy, metadata, and the API.

4.1 Why Trust the Cloud?

- At Salesforce, **trust** is the top priority. Not only are you keeping your sensitive data in your org, you are also building functionality vital to your company's success on Salesforce platform. Salesforce keeps your data and functionality safe.
- Salesforce trust site, trust.Salesforce.com, is a vital resource. You can use it to view performance data and get more information about how Salesforce secures your data. It also shows you any planned maintenance that might impact your access to Salesforce.

4.2 Sharing Is Caring in the Multitenant Cloud

Salesforce is set up more like an apartment building. Your company has its own space in the cloud, but you have all kinds of neighbours, from mom-and-pop shops to multinational corporations.

This idea is **multitenancy**. Multitenancy is a great word for making you sound smart at dinner parties, but really all it means is that you're sharing resources. Salesforce provides a core set of services to all our customers in the multitenant cloud. No matter the size of your business, you get access to the same computing power, data storage, and core features.

Trust and multitenancy go hand in hand. Despite the fact that you're sharing space with other companies, you can trust Salesforce to keep your data secure. You can also trust that you're getting the latest and greatest features with automatic, seamless upgrades three times a year. Since Salesforce is a cloud service, you never have to install new features or worry about your hardware. All this is possible because of multitenancy.

4.3 The Magic of Metadata

To put it simply, **metadata** is data about data. Wait. That's not simple at all. When we say data about data, we're really talking about the structure of your Salesforce org. Let's think about an object like Property. User can input and view data about properties. For example, a property can be located in Boston, cost $500,000, and have 3 bedrooms.

Now, imagine you stripped away all that specific data. What are you left with? You are left with the Property object along with all its fields, like address, price, and number of bedrooms. You can also have page layouts, security settings, and any other customizations you've made.

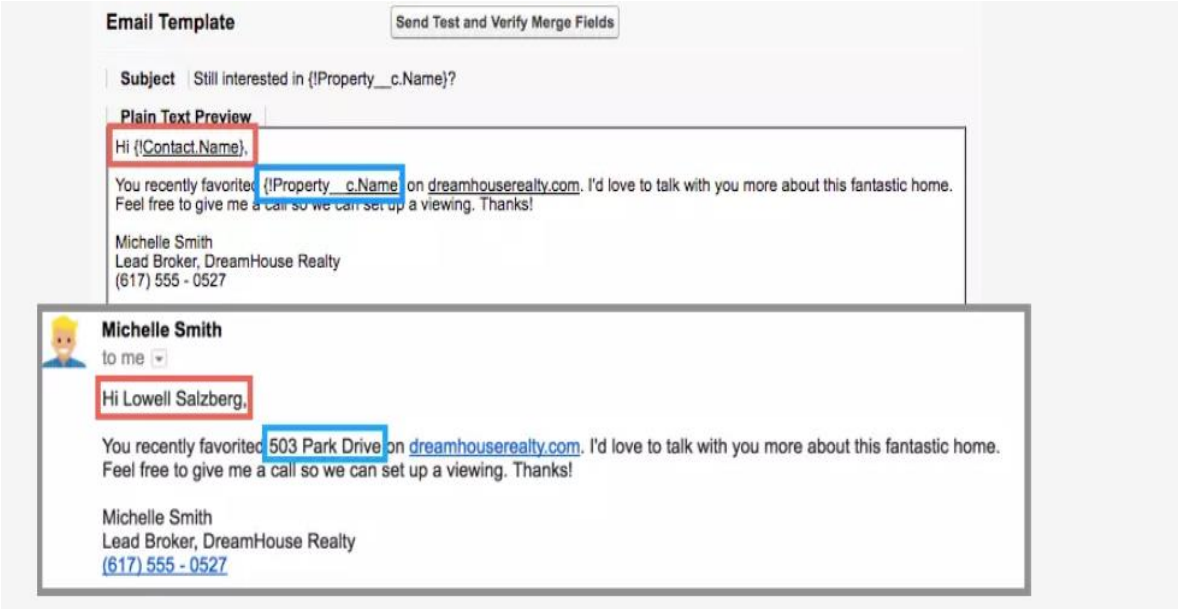
All of these standard and custom configurations, functionality, and code in your org are metadata. Part of the reason you can move so fast on the platform is that Salesforce knows how to store and serve you that metadata immediately after you create it.

4.4 All About That API

Fundamentally, **APIs** allow different pieces of software to connect to each other and exchange information. If that sounds kind of abstract, take a quick look at the computer you're working on right now. You can probably find a series of ports of various shapes and sizes that support different kinds of connections. These are like the hardware version of APIs. You don't have to know how the USB port works. All you have to understand is that when you plug your phone into a USB port, it

passes information to your computer. APIs are similar. Without knowing the details, you can connect your apps with other apps or software systems. The underlying technology takes care of the specifics of how information passes throughout the system.

So, what does this have to do with Salesforce?
When you add a custom object or field, the platform automatically creates an API name that serves as an access point between your org and the database. Salesforce uses that API name to retrieve the metadata and data you're looking for. For example, we can use a contact's Name field in a bunch of places, like the Salesforce mobile app, a custom page, or even an email template. That's all possible because of the API name.



The core of the API's power is that all of your data and metadata is API enabled. This might not seem like a big deal right now, but the API gives Salesforce a huge amount of flexibility. It lets you move beyond the normal idea of business software and build unique and creative solutions for your company.

5.0 Salesforce Editions

Salesforce offers bundles of features and services, each geared toward specific business needs. These are available in both Salesforce Classic and Lightning interface. The various Editions consist of: Essentials, Professional, Enterprise, Unlimited and Developer editions.
These bundles, called *editions*, share a look and feel but vary by functionality and pricing. Some customers start with a basic edition and upgrade to a more feature-rich edition as business requirements evolve. It's easy to learn which features are available in various editions. Just search for features in Salesforce Help. Toward the top of every topic, you can see the editions in which the feature's available. Here's a quick description and what it offers.

Editions

Essentials	Designed for small businesses that want to get started with a CRM system quickly. It includes intuitive walkthroughs and a setup assistant to get started, an easy-to-use UI, and administration tools to customize your deployment as you grow.
Professional	Designed for businesses requiring full-featured CRM functionality. It includes straightforward and easy-to-use customization, integration, and administration tools to facilitate any small to midsize deployment.
Enterprise	Meets the needs of large and complex businesses. It gives you advanced customization and administration tools, in addition to all the functionality available in Professional Edition, that can support large-scale deployments. Enterprise Edition also includes access to Salesforce APIs, so you can easily integrate with back-office systems.
Unlimited	Maximizes your success and extends it across the entire enterprise through the Lightning Platform. It gives you new levels of platform flexibility for managing and sharing all your information on demand. Includes all Enterprise Edition functionality, Premier Support, full mobile access, unlimited custom apps, increased storage limits, and other features.
Developer	Provides access to the Lightning Platform and APIs. It lets developers extend Salesforce, integrate with other applications, and develop new tools and applications. Developer Edition also provides access to many of the features available in Enterprise Edition.

6.0 Getting Started with Salesforce

Every Salesforce user can login into Salesforce by creating an account. We can create free developer account in Salesforce through the URL http://developer.Salesforce.com

Follow the following steps to create free developer account in Salesforce.com (Salesforce login).

1. First you need a valid email Id.
2. Login to www.developer.Salesforce.com
3. Select Sign Up.
4. Provide all the details specially valid email and username. Username must be in the form of a valid email format. Ex : abc@gmail.com.
5. Now click Sign me up.
6. An email will be sent to you email address.
7. Now activate your account by clicking the link provided in the email.
8. The link logs you into the site automatically.
9. Now set your password to your account.
10. Log out from your account.

After creating developer account go to login.Salesforce.com. Use your Username and password credentials for Salesforce.com login.

Salesforce login URL: login.Salesforce.com

- Enter your User name and Password to login.
- Click on Submit button.

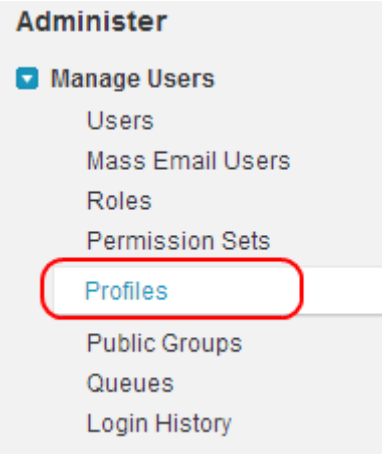
How to set login hour restriction in Salesforce

Salesforce user login restriction and authorisation can be made to check whether user' profile have any login restriction or does the user's profile have any profile IP ranges in an organization.

Follow the steps given below:

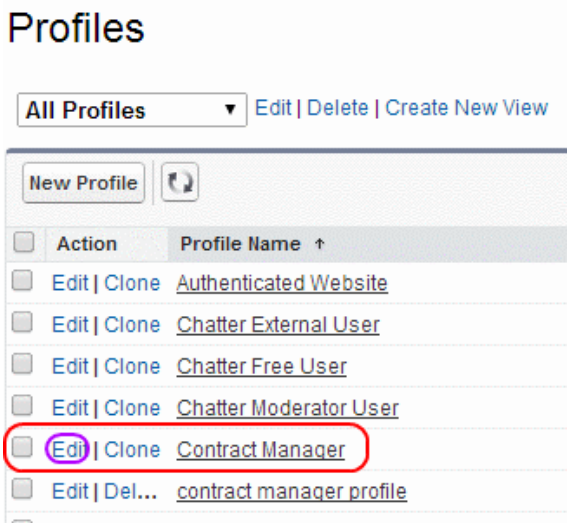
Salesforce Login hours are set in an organization to restrict the user's who tries to *Salesforce login* before or after login hours.

To set Salesforce.com login hours in an organization go to Setup=>Administration=>Manage users=>Profiles.



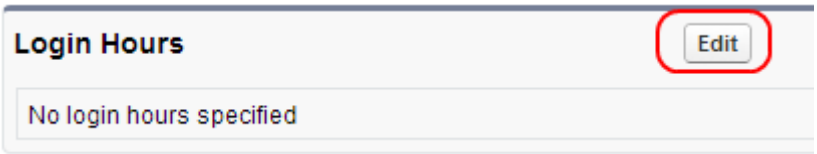
Salesforce login

Select your profile to set login hours and edit it



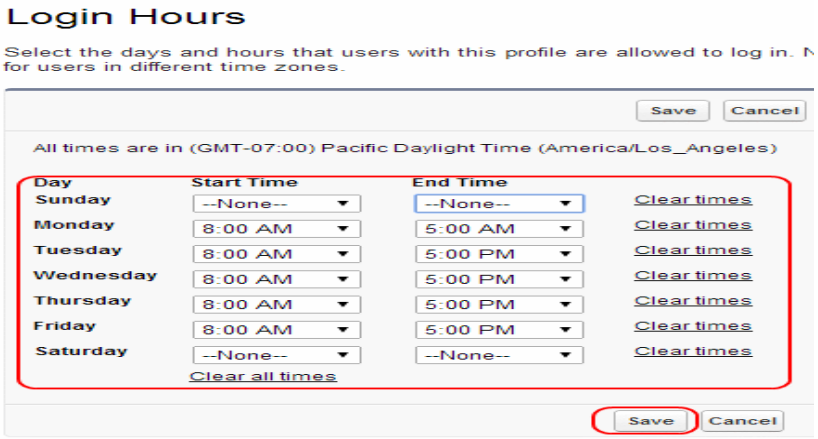
userSalesforce login

Select Edit Button.



Salesforce login

Now go to login hours and select Edit button.



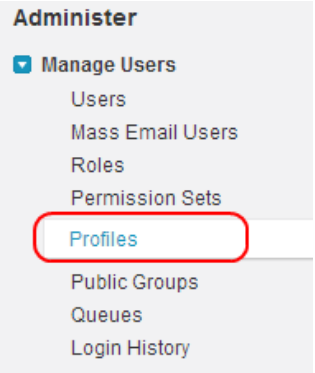
Salesforce login

Enter Start time and End time login time hours and finally save the Settings.

How to set IP address restrictions in Salesforce.

IP ranges are used to restrict any login attempt is done from unknown IP addresses. Usually organizations maintain login IP ranges.

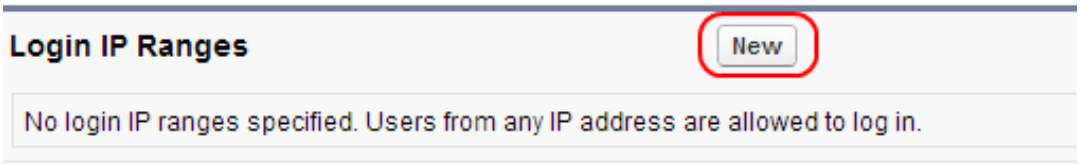
To set Login IP ranges in Salesforce go to Setup=>Administration Setup=>Manage Users=> Profiles.



Now select the profile to which you want to set IP address ranges.

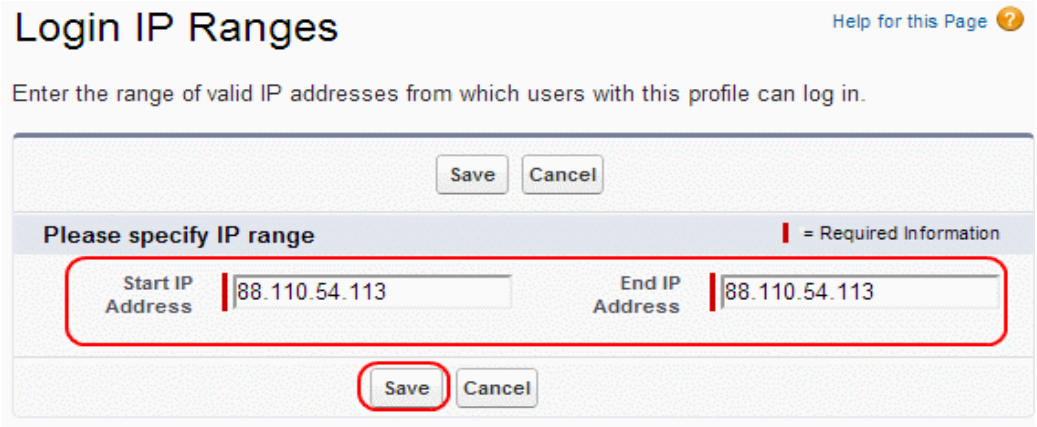
Salesforce login

After selecting edit button go to Login IP Ranges section.



Login IP Ranges

Select New Button to add new Login IP Ranges.



Setting login IP ranges

Enter start IP address and End IP Address as shown above and Finally click on Save.
Successfully we have learned about Salesforce login. In our upcoming Salesforce tutorials we learn *about* Salesforce login history, Salesforce login API, Salesforce developer login, Salesforce login test, Salesforce login history report, Salesforce login to sandbox.

7.0 Sales Cloud Topics in Salesforce CRM

7.1 How to create New Lead in Salesforce

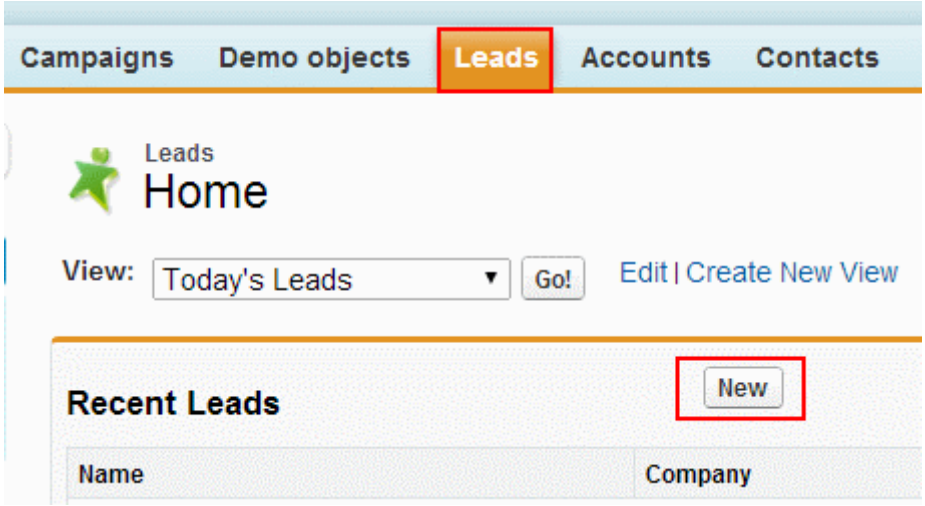
In Salesforce sales cloud is very useful for sales and marketing team in an organization. Basically in sales cloud we have many topics to cover such as campaign, lead, opportunity. Salesforce provided some automation tools which are used to track marketing returns, convert leads to opportunity.

What is a Lead?

When a person enters to data base to purchase some products then we enter this person as a lead.
In Easy understanding we say Leads as Tadpoles and Contacts as Frog.
In an organization when a customer call we have to enter his/her information detail. These information details will be used in future to contact that person. So we have to create new lead in this process.

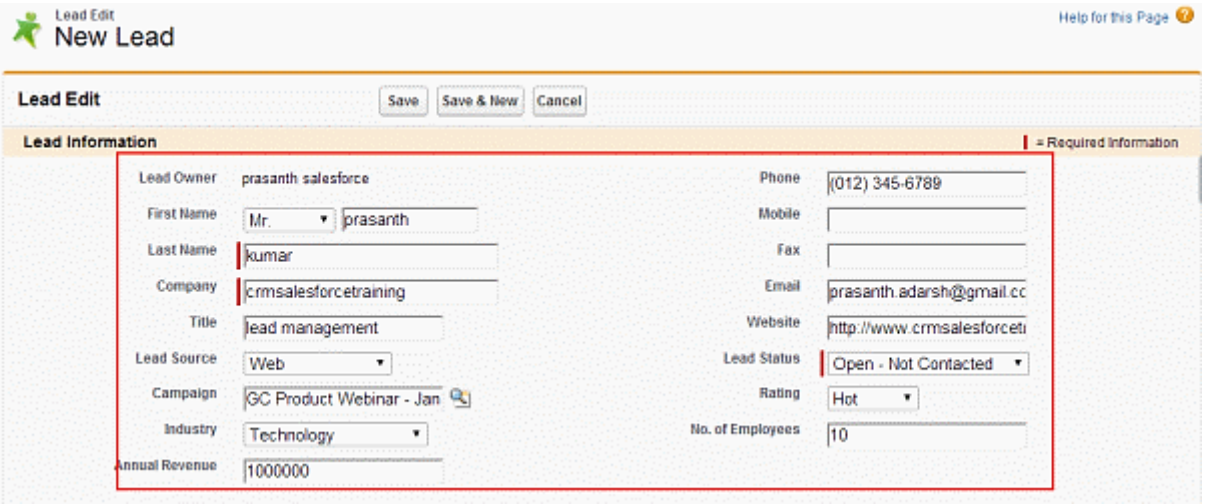
☞ Steps

To create new lead, go to Lead TAB and Select new.



Create New Lead in Salesforce

Enter all the Lead information as shown below.

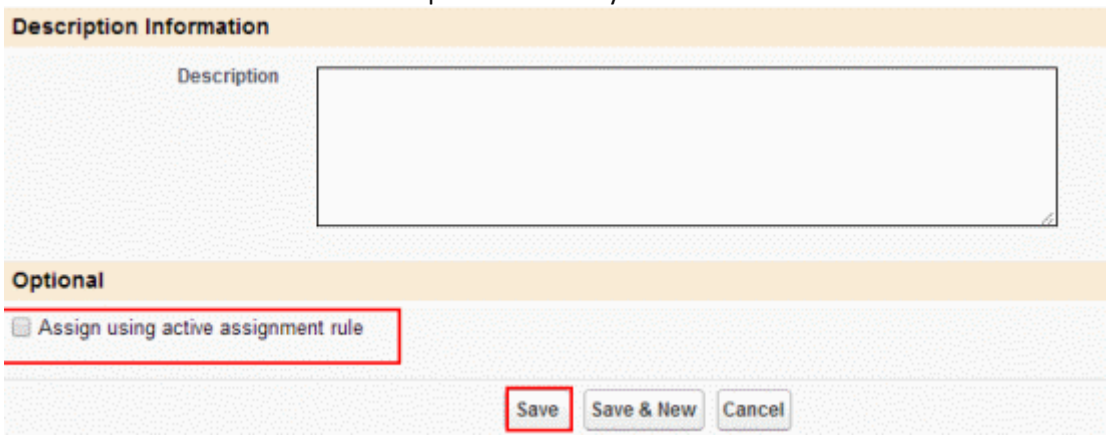


create New Lead in Salesforce

Enter Address information.

Create New Lead in Salesforce

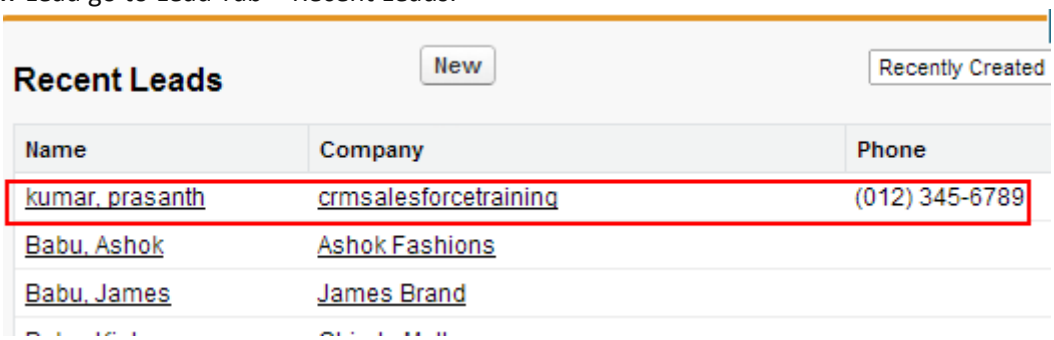
Enter Additional Information and Description and finally click Save Button.



Create New Lead in Salesforce

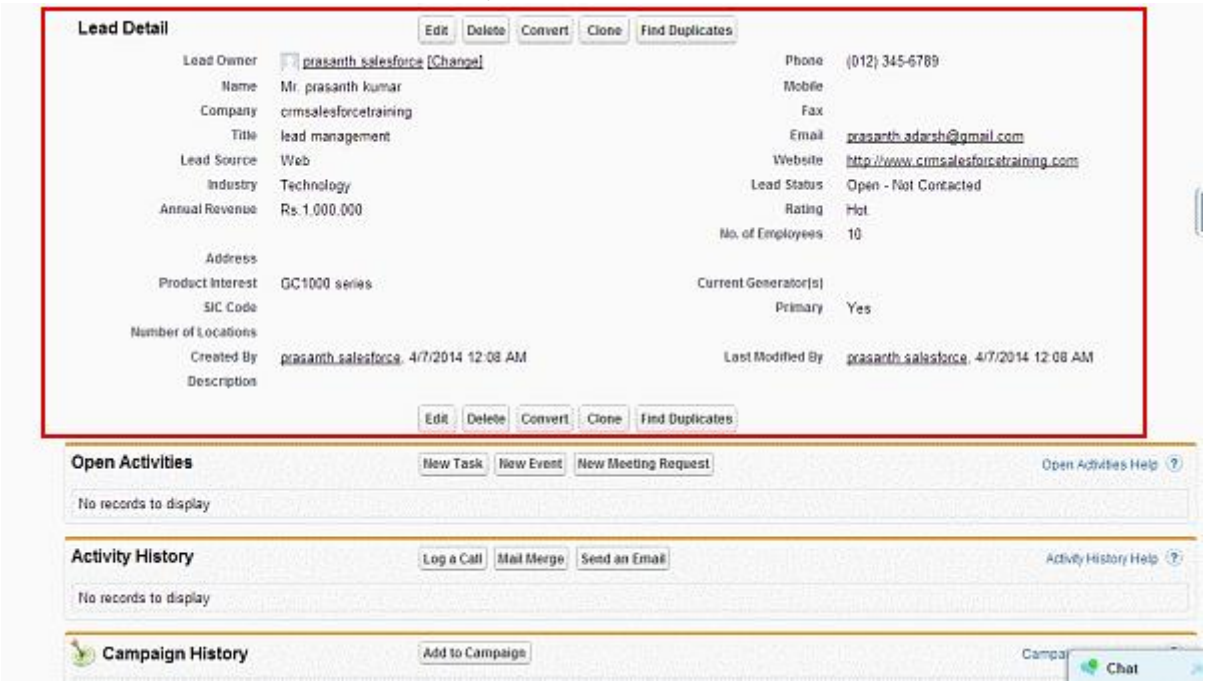
To view or Edit created Lead in Salesforce.

To view Lead go to Lead Tab=>Recent Leads.



Create New Lead in Salesforce

Select the lead which we created to view/edit.



Create New Lead in Salesforce

Successfully we have created new lead in Salesforce CRM. In our next Salesforce Training tutorial we are going to learn how to convert a Lead.

7.2 How to convert a Lead to opportunity in Salesforce

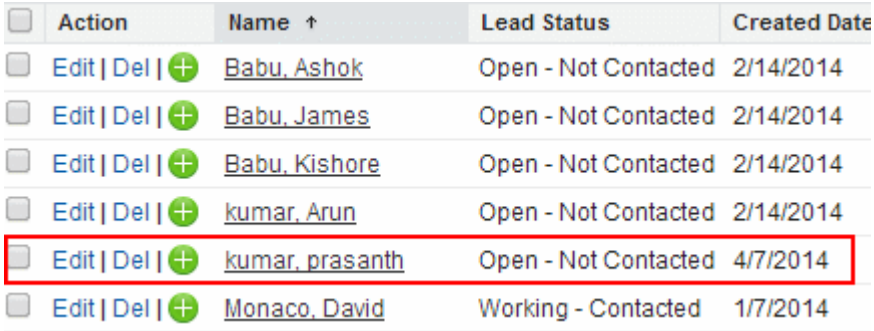
Convert a Lead to opportunity in Salesforce : In our previous chapter we have created new lead in Salesforce. Now in this Salesforce tutorial we are going to learn how to convert a lead to opportunity in Salesforce.

Converting a lead to opportunity takes place only when a lead is qualified in an organization then we convert that lead to an account, lead to contact and lead to opportunity. When a lead is converted to opportunity it appears on forecasting reports in Salesforce.

Convert a Lead to opportunity in Salesforce

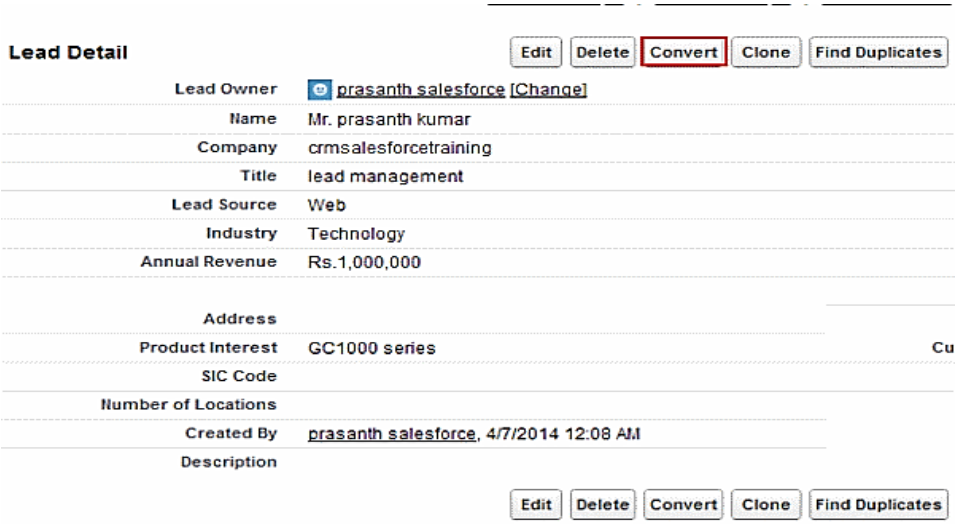
To convert a lead to opportunity in Salesforce go to Lead Tab.

Now select the lead that to be converted to opportunity.

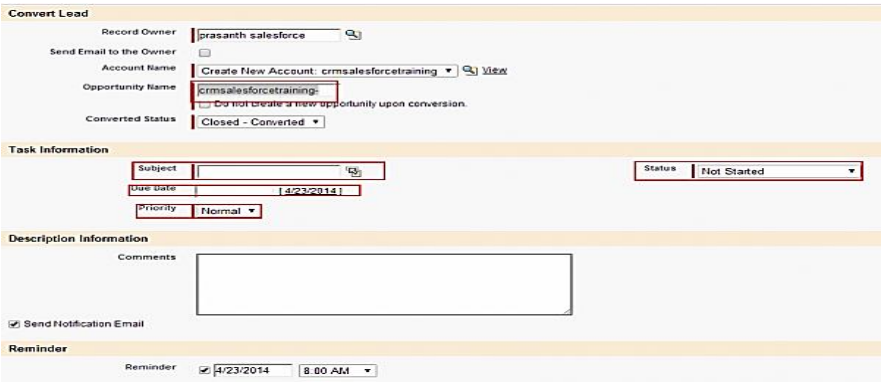


Action	Name ↑	Lead Status	Created Date
Edit | Del |	Babu, Ashok	Open - Not Contacted	2/14/2014
Edit | Del |	Babu, James	Open - Not Contacted	2/14/2014
Edit | Del |	Babu, Kishore	Open - Not Contacted	2/14/2014
Edit | Del |	kumar, Arun	Open - Not Contacted	2/14/2014
Edit | Del |	kumar, prasanth	Open - Not Contacted	4/7/2014
Edit | Del |	Monaco, David	Working - Contacted	1/7/2014

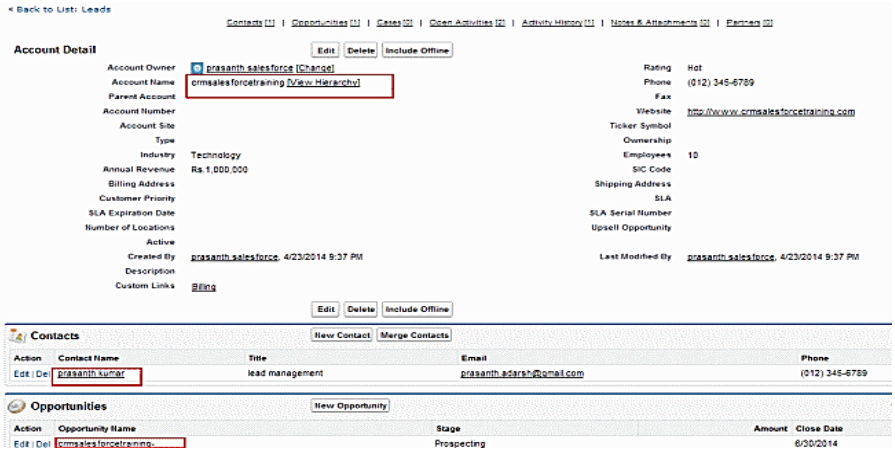
Click convert .



Enter the name for the opportunity, Enter subject name, priority and select status.



Select Convert.



After select convert button now we are taken to Account page as shown above. After converting lead to opportunity Account is automatically created with company name as account name and Account will be created with first and last name of the lead in the contact name field.

7.3 How to create web to lead forms in Salesforce | Lead Management

Web to lead forms in Salesforce: In this chapter we are going to learn how to create web to lead forms in Salesforce, what is lead management, different types of lead management software, sales lead generation software, sales lead management, lead tracking software lead distribution software. Every company needs to drive qualified leads that steadily improves and increase the conversion rates. Through this lead management we can analyse website visitor's data which helps to inform sales and marketing teams in an organization.

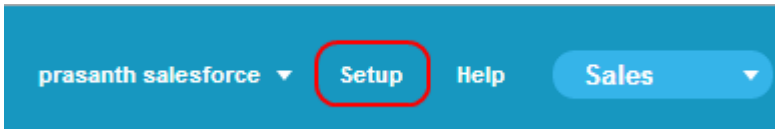
Advantages of Internet marketing or lead management on sales?

- In Salesforce web to lead forms captures keywords that customers use in both organic search and paid search to find your company product.
- It helps to bring website visitors data.
- It increases sales.

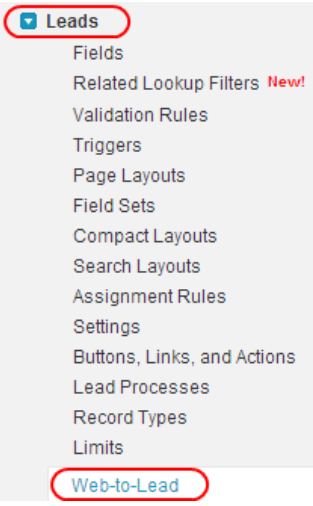
How to create web to lead forms in Salesforce?

Web to lead forms in Salesforce are the standard functions in Salesforce. These webs to lead forms are embedded in company website to gather all the leads from customers.

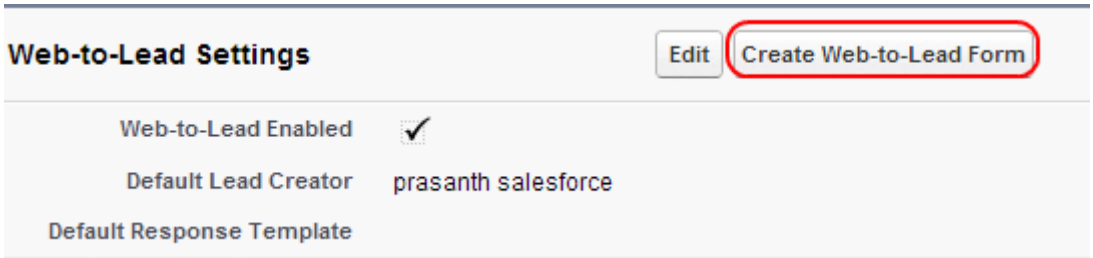
To create web to lead forms in Salesforce go to Setup.



Now go to Customize=>Leads=>Web to Lead.

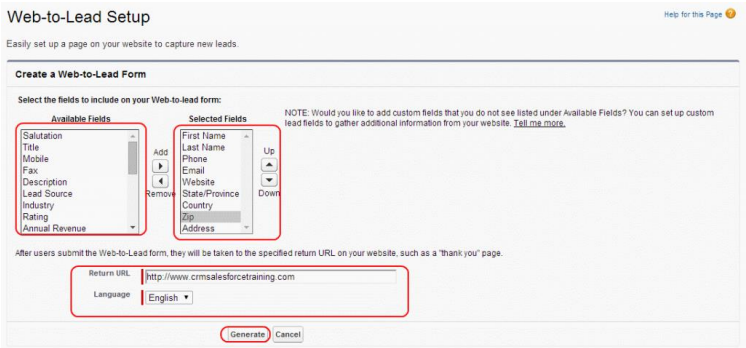


Select web to lead as shown above.



Web to lead forms in Salesforce

Now select create web-to-lead Form button as shown above.



Web to lead forms in Salesforce

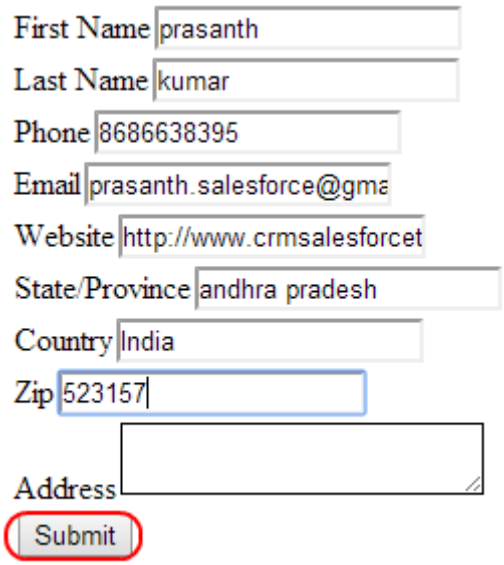
After selecting web to lead button we are taken to new page where we have to select fields that to be appear in Web to lead form. Select the fields from Available Fields. Selected fields will be available as shown above. After selecting fields enter the returning URL we have entered http://www.crmSalesforcetraining.com as returning URL. Select language.

Now select Generate Button.

Now HTML code will be generated. Now follow below steps.

- Copy HTML code and paste it on Notepad.
- Save that with .html extension (example web to lead.html)
- Go to web to lead from and open it on browser.
- Now web to lead form will be generated.

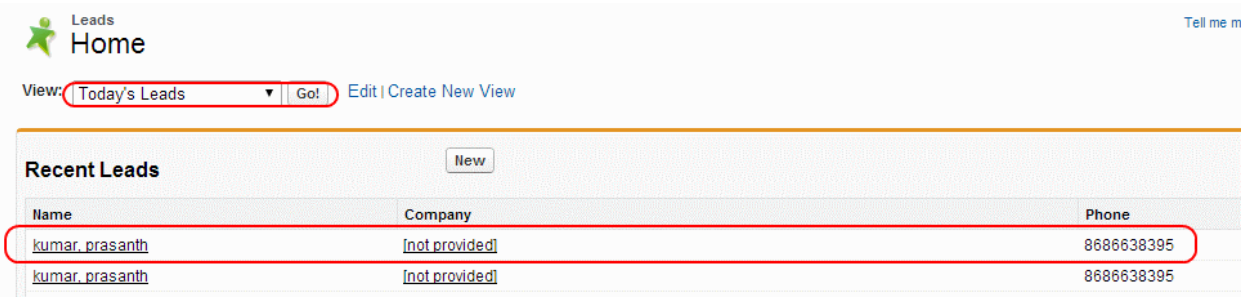
To check Web to lead forms in Salesforce code.
Copy the code and paste it on the website page. Now web to lead form will be formed as below.



Web to lead forms in Salesforce

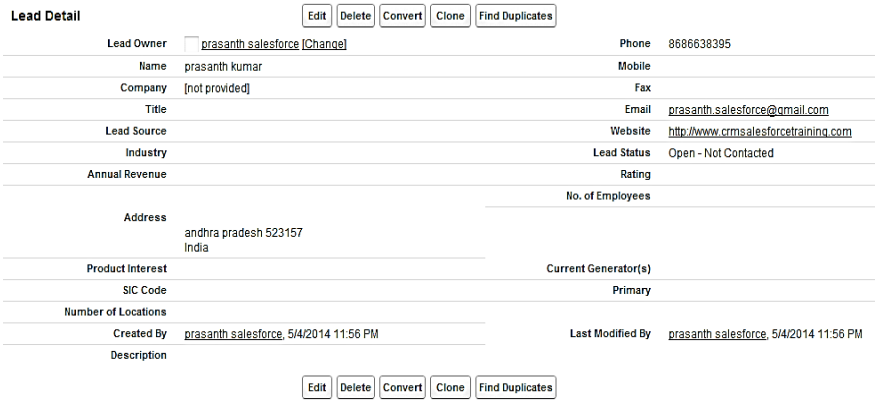
Enter all the details as show above and select Submit button. After selecting submit button new lead will be generated in Salesforce.

Now log in to Salesforce.com through login.Salesforce.com and go to Leads Tab.



Web to lead forms in Salesforce

Lead which is generated through web to lead form as shown above. Open that lead to view the details.



Web to lead forms in Salesforce

We have successfully created web to lead form in Salesforce. With the web-to-lead form feature we can create up to 500 leads per day.

7.4 Creating a new account in Salesforce

Creating a new account in Salesforce; in our previous chapter we have learned how to create new leads in Salesforce and how to convert lead to opportunity. In this Salesforce tutorial we are going to create new Account in Salesforce.

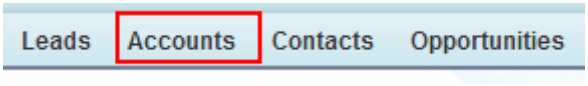
Accounts in Salesforce are used to track customers. For every customer to a company an account will be created. In that account we have opportunities, tasks, contracts are linked to that account. Accounts are typically business which have field about the organization.

Creating new Account in Salesforce can be done in two ways.

1. Creating new account by clicking Account Tab.
2. Importing Accounts and contacts from spreadsheets by selecting import my accounts and contacts.

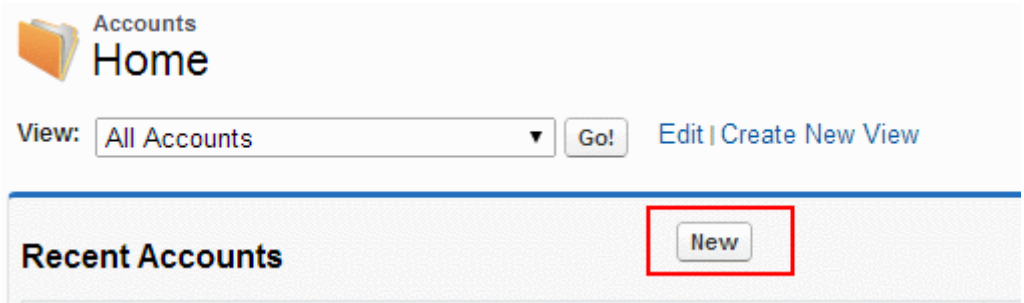
Creating new account in Salesforce.

To create new account go to Accounts Tab.



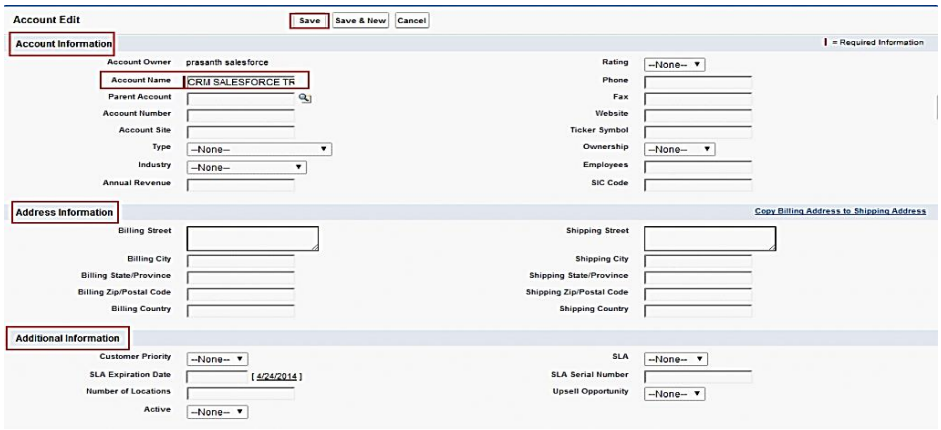
Creating a new account in Salesforce

Click on Accounts Tab and select New as shown below.



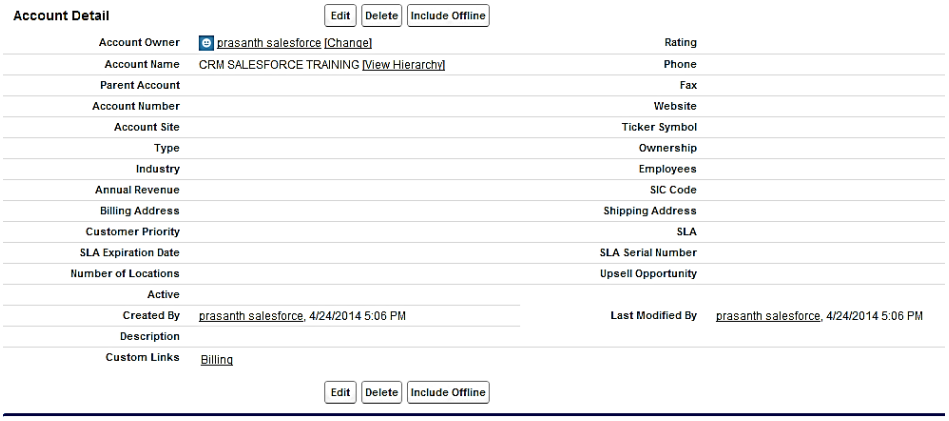
Creating a new account in Salesforce

After selecting new we are taken to new page where we are required to enter Account information, address information and Additional Information to create new account. Account name is important and required field it must be entered.



Creating a new account in Salesforce

Click Save to save all setting and the detailed view of contact is displayed as shown below.



Creating a new account in Salesforce

To merge duplicate accounts in Salesforce, go to Accounts tab and select Merge Accounts button and enter all the details like duplicate account name, master account. We can merge only 3 records at a time.

7.5 Creating new Contact in Salesforce

Creating new Contact in Salesforce; in previous module we have learned how to create new account in Salesforce and now we will create new contact in Salesforce.

Contact records are used for individuals to store personal and business information. In contacts we can store phone numbers, E-mail address. Department name, Fax etc. Last name field is required when creating new contact in Salesforce.

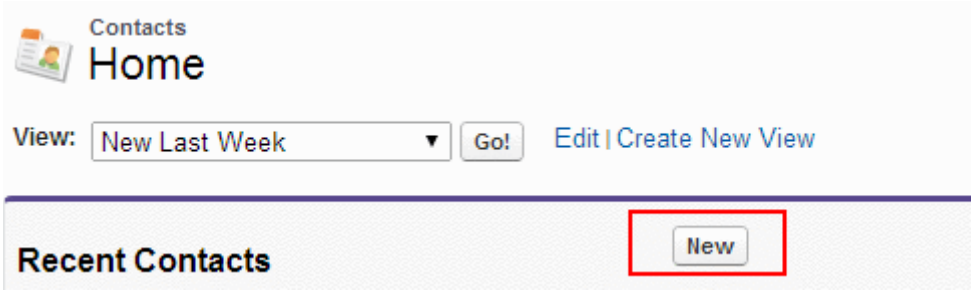
Both Contacts and accounts have relation with each other. Contacts are typically associating with an account. An account will have more than one contact.

Creating new Contact in Salesforce.

To create new contact in Salesforce, go to Contacts Tab.

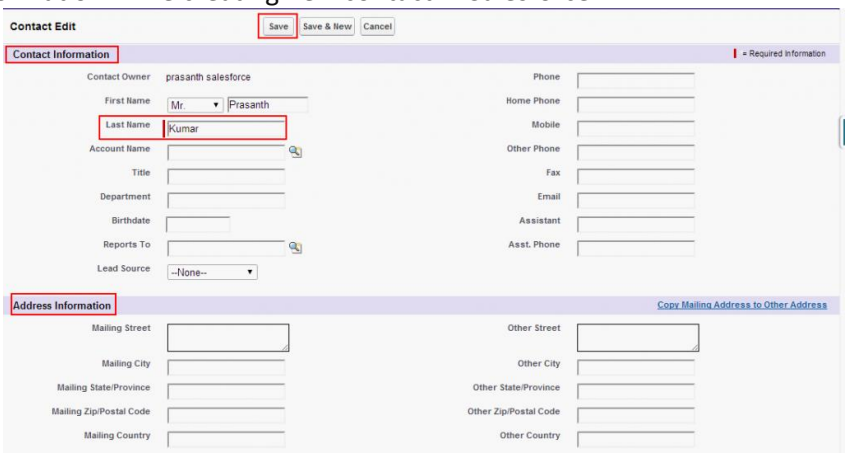
Creating new Contact in Salesforce

After selecting Contacts Tab select new as shown below.



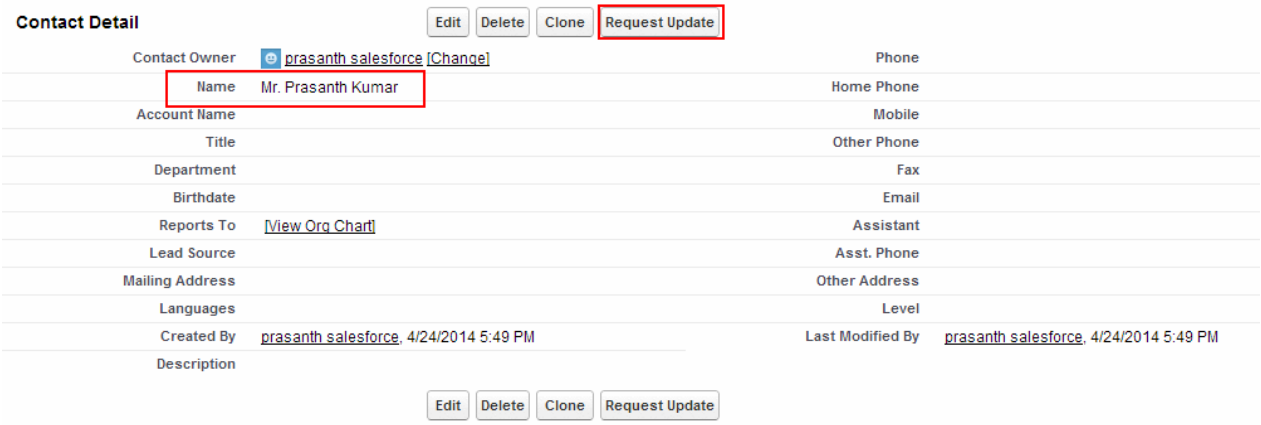
Creating new Contact in Salesforce

Now we are required to enter contact information, Address information, Additional information and description information while creating new contact in Salesforce.



Creating new Contact in Salesforce

Enter First name, Last name, name of the account for contact and finally Save it.
Contact details will be displayed in detailed view. If we want to change contact details in Salesforce select request update.



Creating new Contact in Salesforce

7.6 How to create new Opportunity in Salesforce

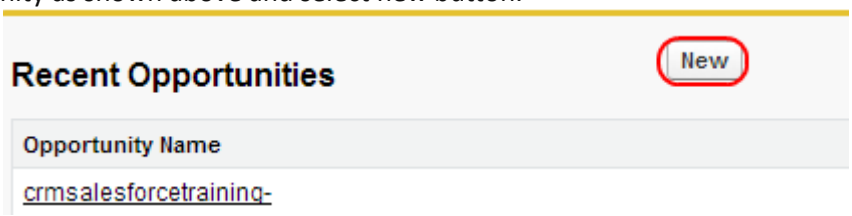
How to create new Opportunity in Salesforce: In this chapter we are going to create new opportunity in Salesforce. In our previous chapter we have learned how to create new email template in Salesforce. Opportunities plays major role in an organization because they represents sales and potential sales. Opportunities may have quotes, proposals and orders.

Creating opportunity takes place when a customer contacts salesperson for a product and interested to purchase the product. Then salesperson creates an opportunity with Account information, address information and additional information. This opportunity is linked to campaigns to increase the effectiveness of the sales in a company.

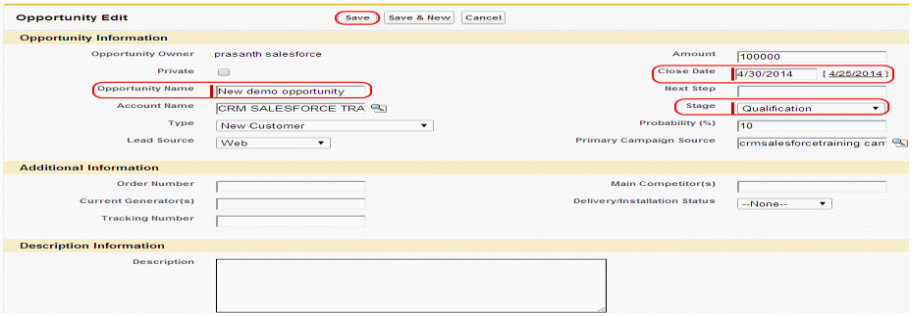
How to create new opportunity in Salesforce.
To create new opportunity in Salesforce go Opportunity tab.

How to create new Opportunity in Salesforce.

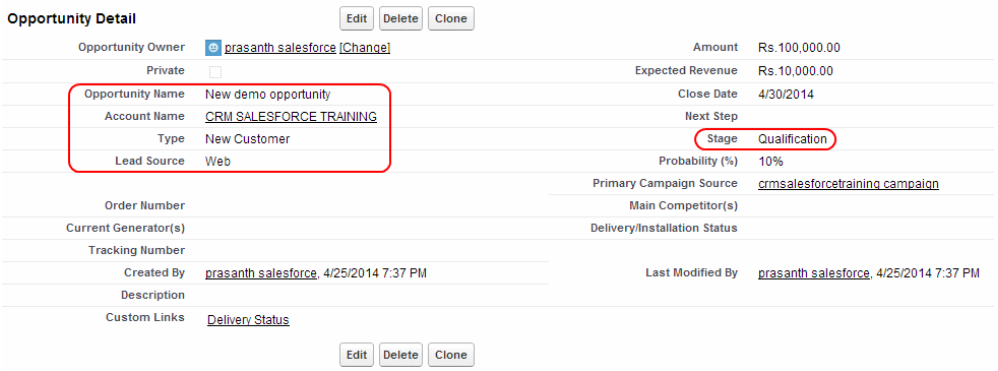
Select opportunity as shown above and select new button.



How to create new Opportunity in Salesforce.
To create new opportunity in Salesforce some required fields are to be entered. Required fields like Opportunity name, Close date, stage and the remain fields are not required field they may entered or may not be entered.



How to create new Opportunity in Salesforce.
Enter Opportunity name, close date, account name, opportunity stage. You may also enter additional information and description information.
Select Save



How to create new Opportunity in Salesforce.

After saving the opportunity we are taken to detailed view of the opportunity observe the field as shown above. While creating opportunity in Salesforce we can include stage and probability fields to estimate the likelihood of the completing a successful sale. The probability for various stages can be set by administrator.

Opportunity have been successfully created

7.7 Adding product and price book to an opportunity in Salesforce

In this chapter we are going to create and adding product and price book to an opportunity in Salesforce. Products are the individual items will belong to one universal product catalogue in an organization. After creating product, we can associate that product to multiple price books and also can be associates to multiple prices in Salesforce.

In Salesforce a product can be scheduled based on quantity, revenue or both quantity and revenue. We can add new product to an opportunity in Salesforce. In Salesforce a product may have different types of standard fields when we create new product

What are the different types of standard product fields in Salesforce?

- Product name: Product name must be specified clear and familiar.
- Product code: Product code is an internal code or product id which is used to identify a product.
- Product description: We have to enter text in product description area to distinguish products from each other.
- Product Family: Product family means it is the category of the product. By setting product family in Salesforce the product automatically maps to it product family.
- Quantity Scheduling Enabled: To enable quantity scheduling for a product the check box must be checked.
- Revenue scheduling Enabled: To enable revenue scheduling select the check box.

<u>Steps to add a product to an opportunity.</u>

- First create a product in Salesforce.
- Second go to opportunity detailed page to the product related list.
- Select price book button.
- Select price book from the price book drop down and select save.

Creating new product in Salesforce

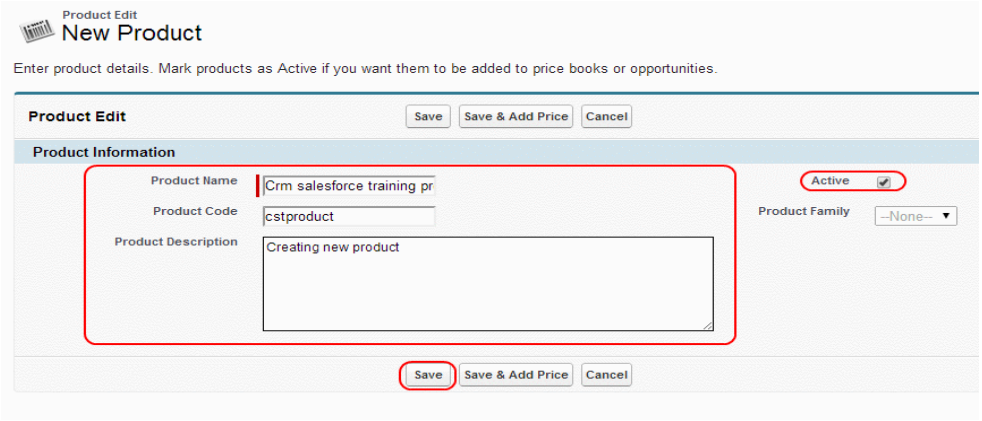
To create new product in Salesforce go to Product Tab as shown below.

Adding product and price book to an opportunity in Salesforce

Now select button to add new product.

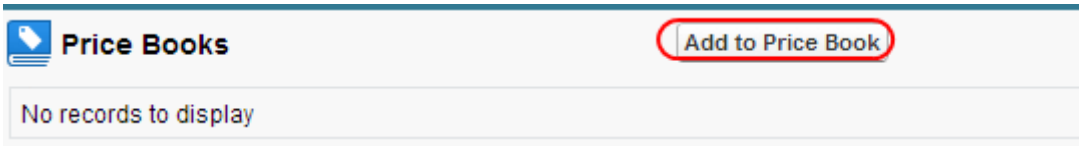
Adding product and price book to an opportunity in Salesforce

After selecting new product button, we have to enter product details like product name, product code and product description.



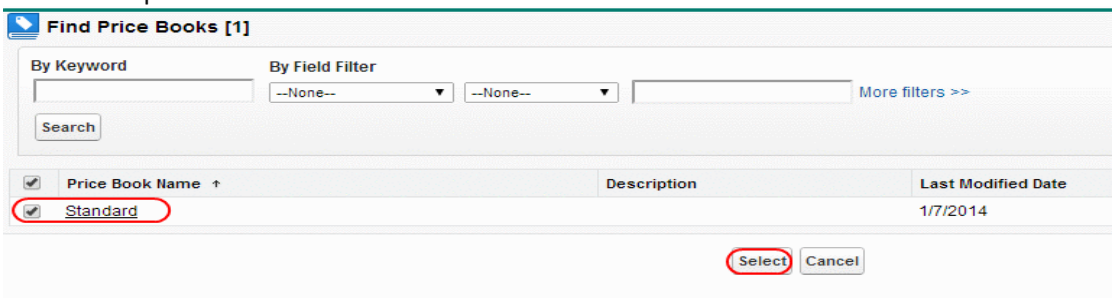
Adding product and price book to an opportunity in Salesforce

Select Save & Add Price button as shown above. Add price to the product in next step as shown below.



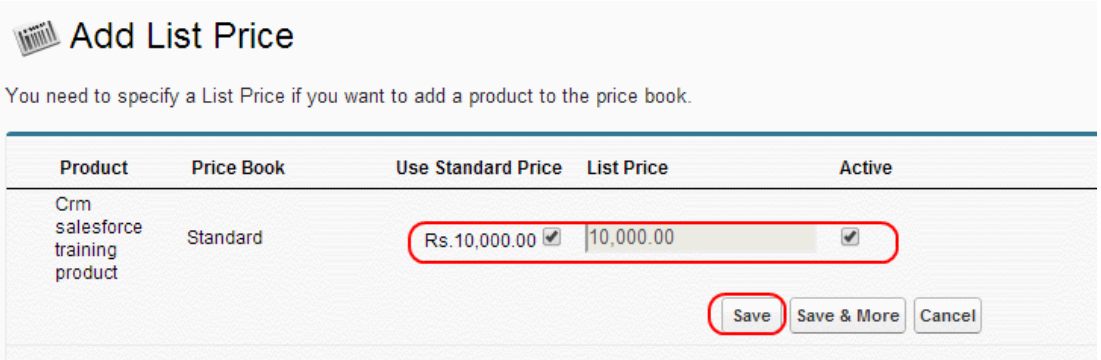
Adding product and price book to an opportunity in Salesforce

Now Select the price book Name as show below.



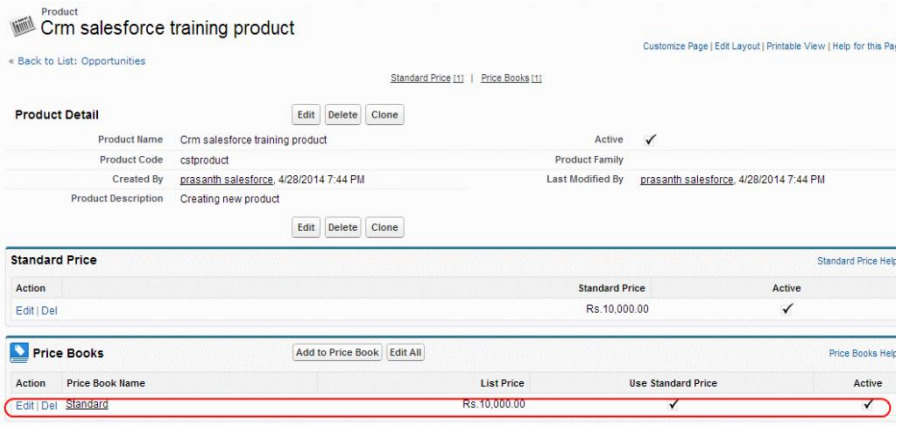
Adding product and price book to an opportunity in Salesforce

Now Select button as shown above .



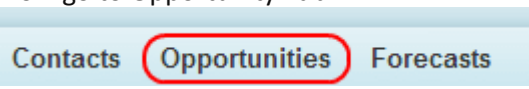
Adding product and price book to an opportunity in Salesforce

Enter list price if the product price is different to other places. If the product price book is equal for all select Use standard price check box and select Save as shown above.

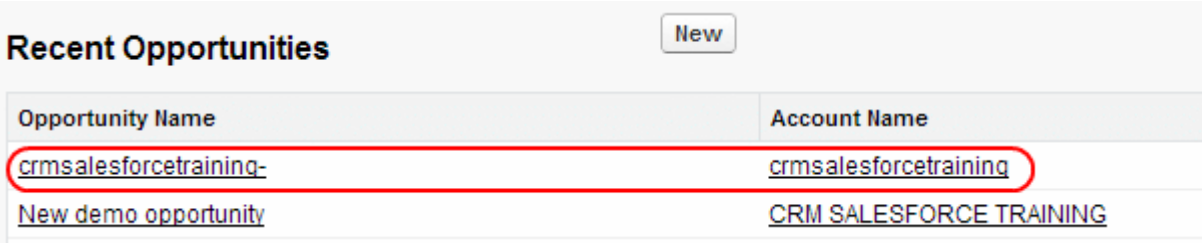


Adding product and price book to an opportunity in Salesforce

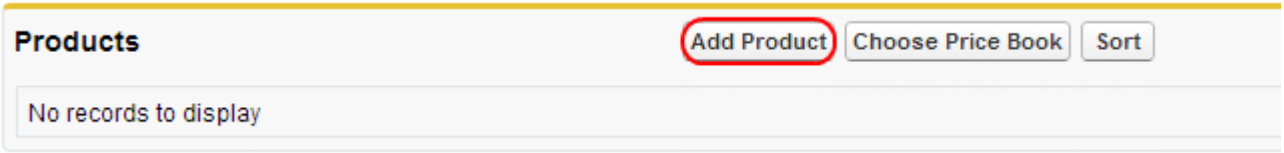
Now go to Opportunity Tab.



Now Select the opportunity to which we want to add product.



In detailed view of opportunity go to product section by scrolling down.

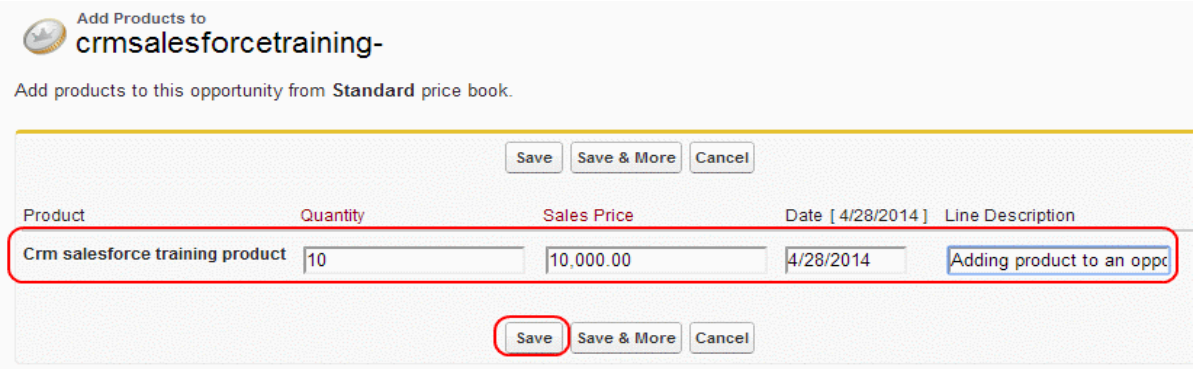


Adding product and price book to an opportunity in Salesforce

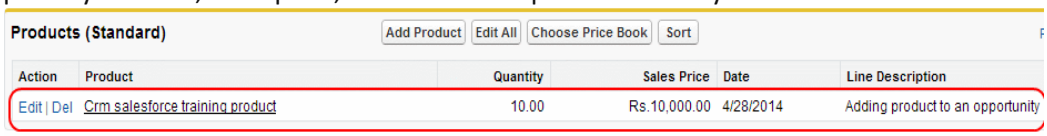
After selecting Add Product button we are taken to product list. We have to select the product name from the list.

Product Name	Product Code	List Price	Product Description	Product Family
Colgate fresh	cqf6358	Rs.50.00	product name is Colgate fresh wit...	
Crm salesforce training product	cstproduct	Rs.10,000.00	Creating new product	
GenWatt Diesel 1000kW	GC1060	Rs.100,000.00		
GenWatt Diesel 10kW	GC1020	Rs.5,000.00		
GenWatt Diesel 200kW	GC1040	Rs.25,000.00		
GenWatt Gasoline 2000kW	GC5060	Rs.150,000.00		

Select the product name as shown above.



Enter quantity number, sales price, date and description and finally select Save button.



Adding product and price book to an opportunity in Salesforce. Adding product and price book to an opportunity is successful.

7.8 Salesforce Activities, creating and viewing Salesforce activities

Salesforce Activities: In this chapter we are going to learn about *Salesforce activities, different types of Salesforce activities, creating Salesforce activities and viewing existing Salesforce activities.*

What are the activities in Salesforce?

Salesforce activities are of different types they are scheduled calendar, events and tasks. Salesforce activities are similar to the activities in Microsoft outlook and Google applications. Generally activities in Salesforce are used to schedule events, inviting people to a function, adding certain task to an employ and so on. Mainly these activities are used to get things done without any failure.

Events and tasks are related to records like accounts, contacts, leads, opportunities and cases. By using Salesforce activities, we can track all the tasks and events that are scheduled or involved in acquiring, selling, and servicing customers.

What is a task in Salesforce?

A task is an activity in an organization that needs to be completed. A task may not have any specific time period or time duration for completion. Example: - Suppose marketing team wants to send email to the existing customer about the new product they launched now the marketing team create a task called send letter.

What is an Event in Salesforce?

An Event is generally a calendar activity which has scheduled time, date and duration.

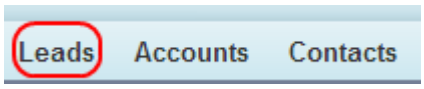
Log a Call: - In Salesforce Log a call is an activity where important data from the customer is gathered by making a call to customers.

Send an Email: - Send an Email is an activity in Salesforce where we can send emails from Salesforce and also from third parties like Microsoft Outlook and Gmail.

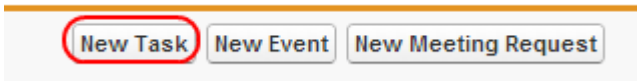
How to create new task in Salesforce (Salesforce activities)?

Now we are going to create Salesforce activities. Before creating activities in Salesforce, we must understand which type of activity we have to create. Suppose if we want to schedule an activity which have specific location, time and duration there we create events.

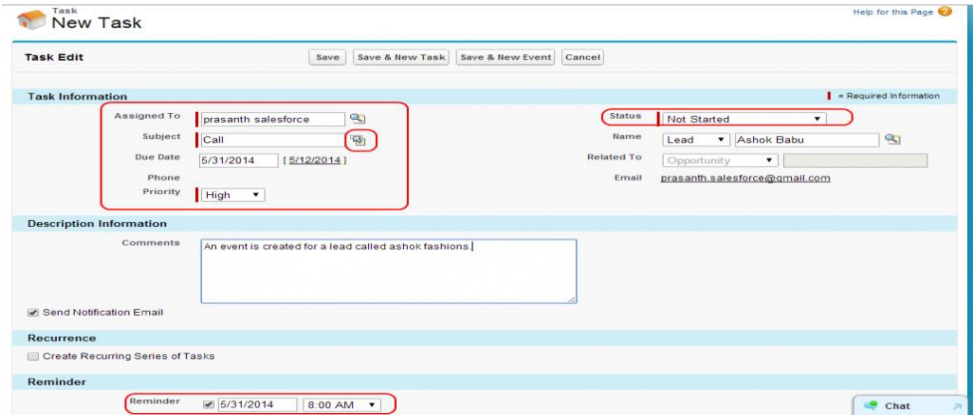
We are going to create an activity for Leads. To create activity select Leads Tab=> Select the Lead.



Select Leads Tab as shown above and from the list of leads select any of the lead to which you want to create new task. View the lead in detailed view and select New Task as shown below.

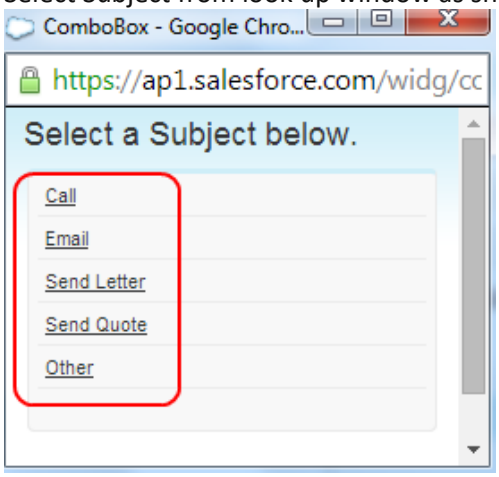


Now a new window will be display with different fields like Assigned To, Subject, Due Date, Priority, Status, Name. Enter all the details.

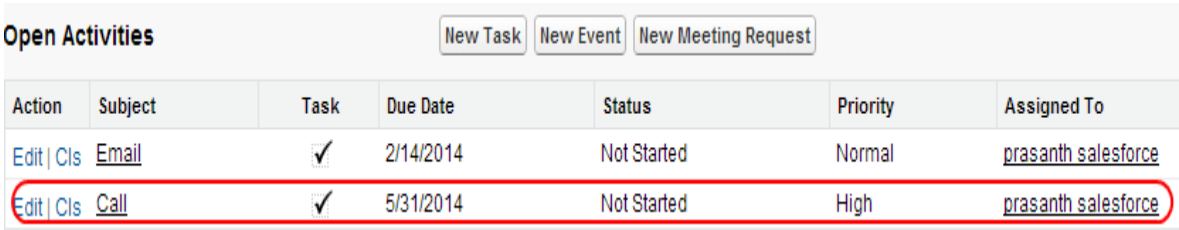


Salesforce Activities

Select Subject from look up window as shown below.



Now Select Save Button.



Successfully we have created new Task.

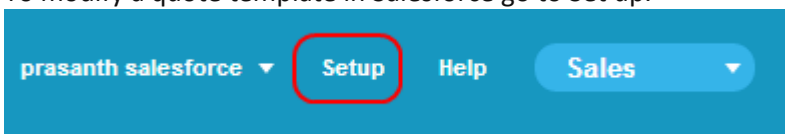
7.9 Modifying Quote Template in Salesforce

In this chapter we are going to learn how to modify a quote template in Salesforce. Salesforce.com provides us an easy way to generate quotes in PDF format. These PDFs are the copies of Contracts, quotes, proposals. Salesforce provides us the option to save the PDF quotes to a disk and also be sent to customers through an E-mail.

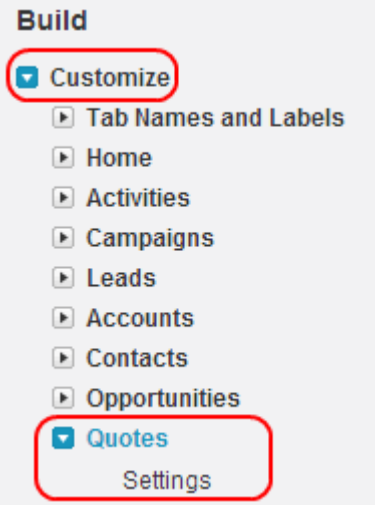
Modifying quote template is the way of designing the documents that matches to our standard branding by adding Logo, images, we can redesign header and footers and other design elements. In documents we can also add contract line items and signature if required.

☞ Steps

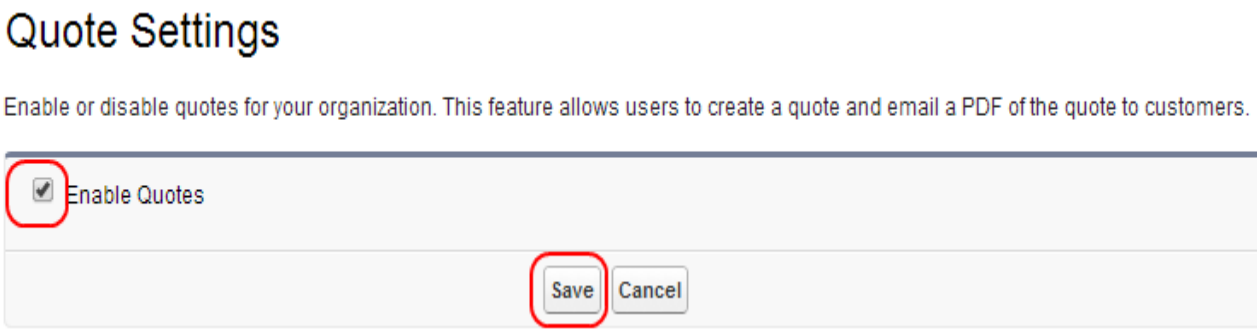
To modify a quote template in Salesforce go to Set up.



Now Go to Customize=>Quotes=>Settings.

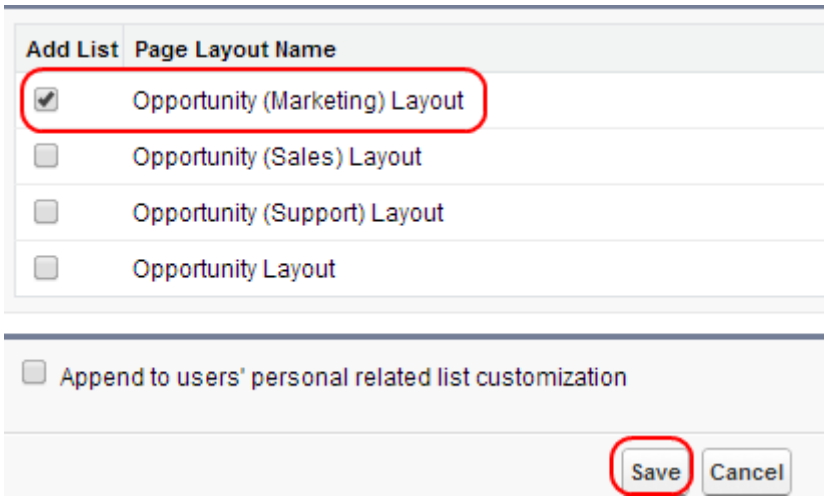


When Settings in Quotes are selected a new window will be opened where we have to select the check box to enable quotes in Salesforce.



Modifying Quote Template in Salesforce

Select Save Button.

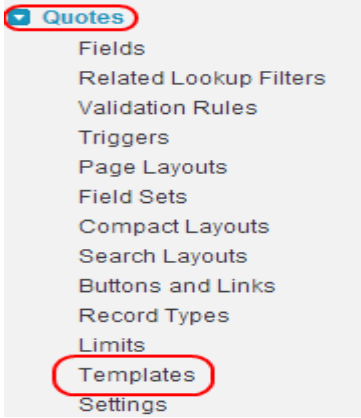


Modifying Quote Template in Salesforce

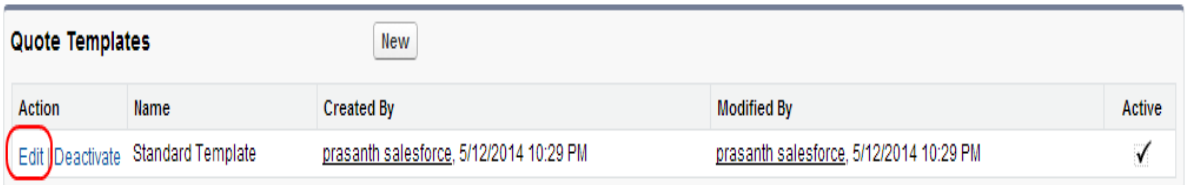
Now a list of options will be displayed to add a list. Select any of the page layout by selecting the check box as shown.

Select Save.

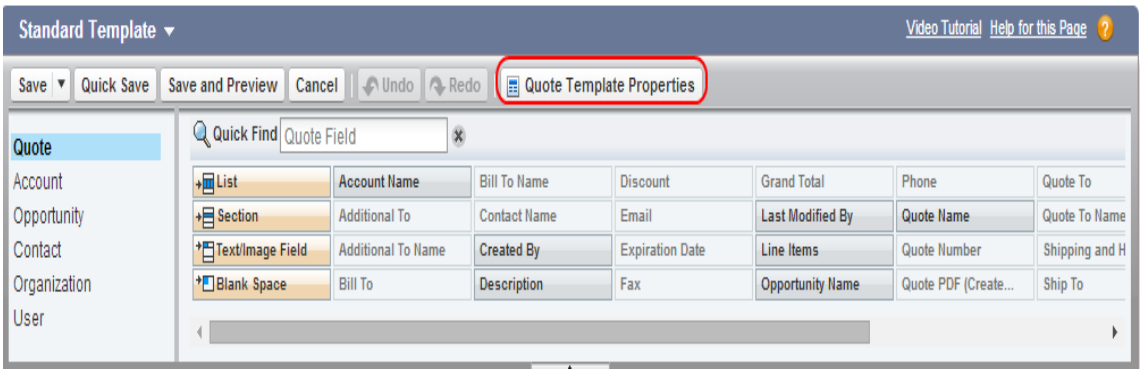
Now go back to Customize=>Quotes=>Templates.



Select Templates.

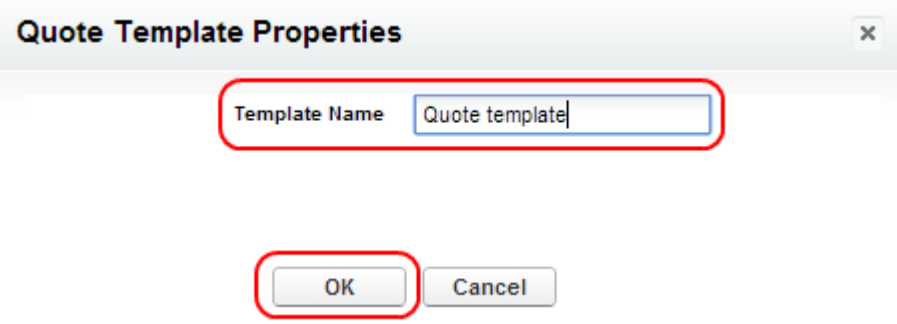


Select Edit.

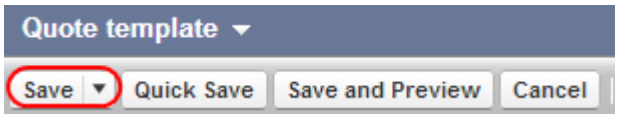


Modifying Quote Template in Salesforce

Select Quote template Properties to edit layout, to rename template name.



Select OK.

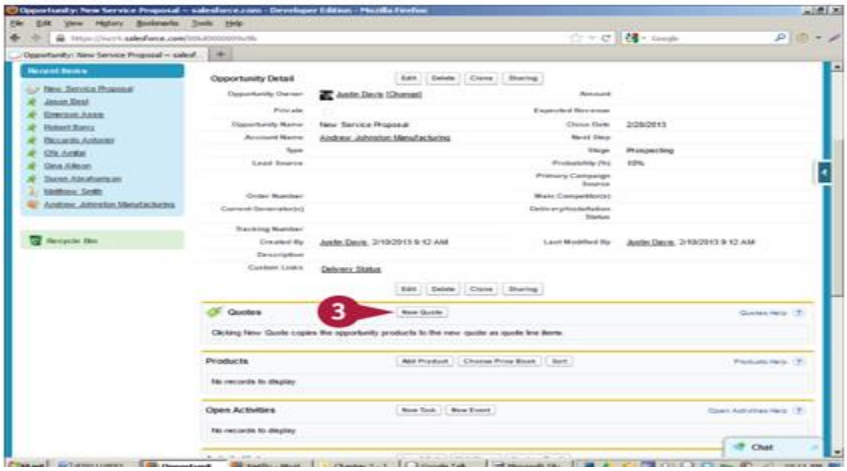


Now Finally select Save the settings.
Select the list of fields that to be appeared in the quote by the fields. Digital Signatures in Quotes can be inserted by using Salesforce App exchange applications such as DocuSign and EchoSign.
Generating quotes in Salesforce.com
Generating Quotes in Salesforce allows us to collect information from products and price books which are collected in to opportunities. Generating quotes is Salesforce are done only in PDF format and sending the PDF quotes through email to a customer. If we want to generate quotes in any other format, we have to use other tools like Congo Merge. We can also use mail merge templates to generate documents in Microsoft Word format.
In this chapter we are going to learn how to customize quotes in Salesforce, how to generate quotes in Salesforce, preparing quotes for customers, revising quotes, sending quotes.
Generating quotes in Salesforce.
Before generating quotes in Salesforce, we have to create Quote Template in Salesforce. After creating quote template in Salesforce only we have to generate quotes. In this Salesforce training tutorial, we are going to create Generate or Build quotes for Opportunities.
Quotes are created for only opportunities in Salesforce to generate quotes for opportunities go to Opportunity Tab and select the opportunity record.

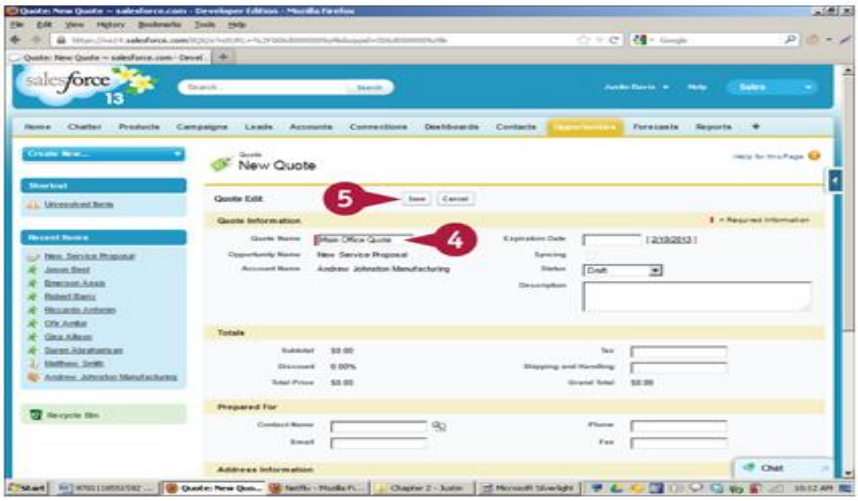
☞ *Steps*

Step 1 :- Go to Opportunity Tab.

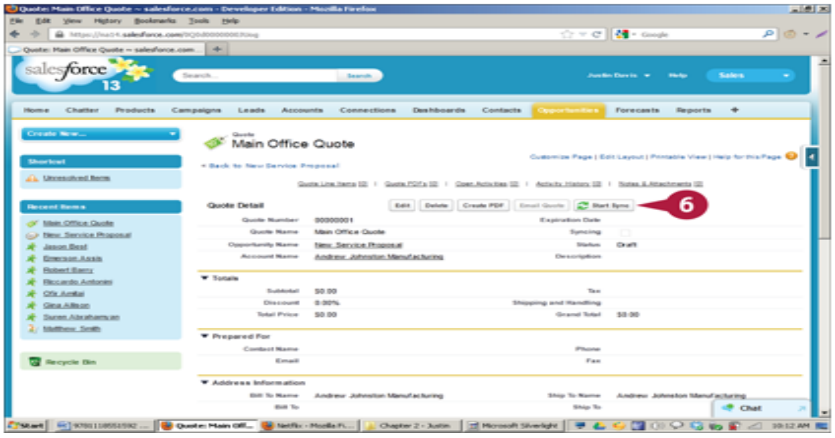
Step 2 :- Select the opportunity to which the quotes are to be received.



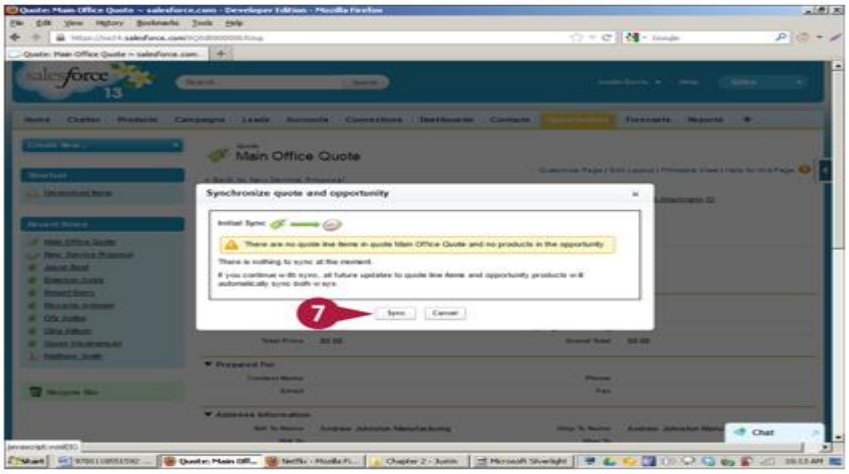
Step 3 :- Select New quote button as shown above.



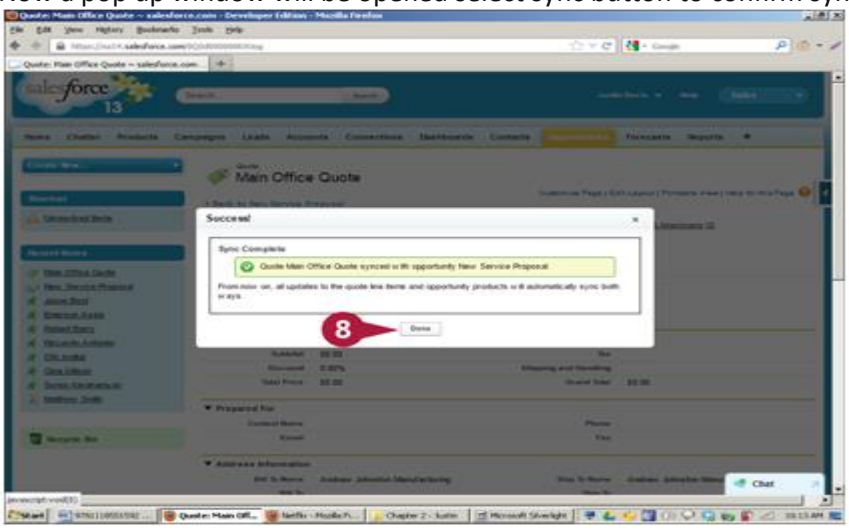
Enter Descriptive name for the quote and select Save button as shown above.



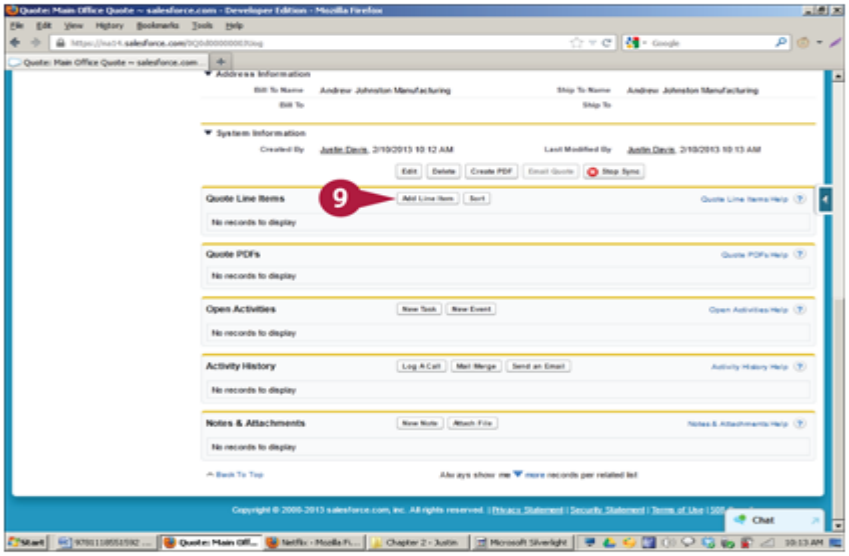
Now we have to sync. Select Start Sync Button as shown above.



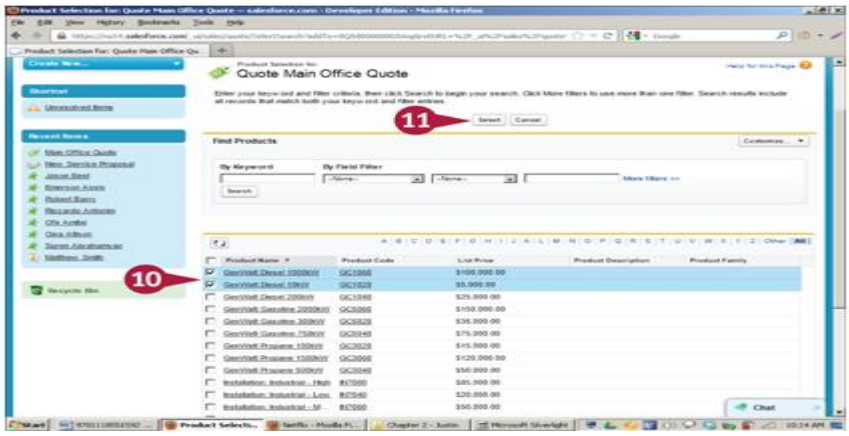
Now a pop up window will be opened select sync button to confirm syncing.



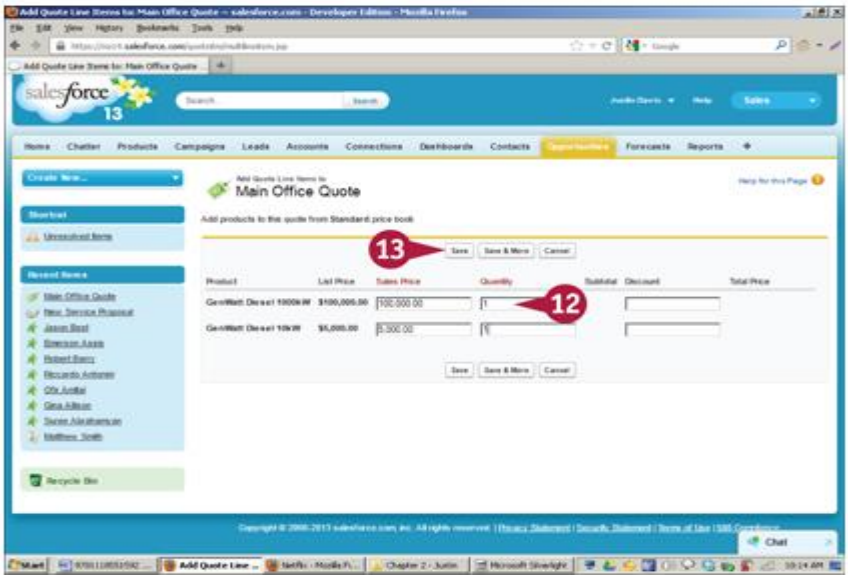
Syncing processes will be done after few seconds. Select Done.



Now select Add Line items button.



After selecting line items new window will be appeared to which we have to select the items that to be appeared in quote. Select the check boxes as shown above to add products in quotes.
Now Click on Select button.



Enter the product quantity for each item that we selected previously in the quote.
Finally select Save button and successfully we have created Quotes in Salesforce. Add discounts to quotes by defining discount percentage for a quote line item. Discount percentage will be automatically included in the final quoted price.

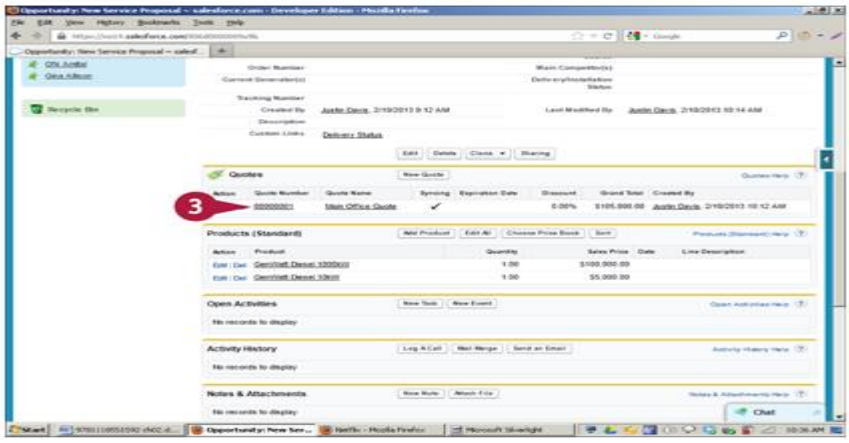
7.10 Email a Quote in Salesforce

Emailing a Quote in Salesforce for the customers is done for the processes from the customers. The quote must be approved by the customers. Emailing a quote is done by attaching a PDF Quote to the customers or recipients Email addresses. The recipients will download the attachments and view the downloads in normal PDF format. We can also create custom Quote Templates with different Quote Format which includes Quotes details in it.

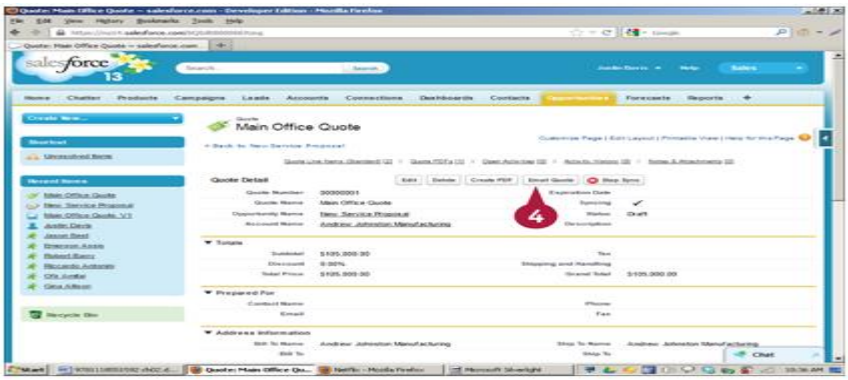
☞ Steps

Now we learn how to email a quote in Salesforce to customers or recipients. Click on Opportunities Tab.

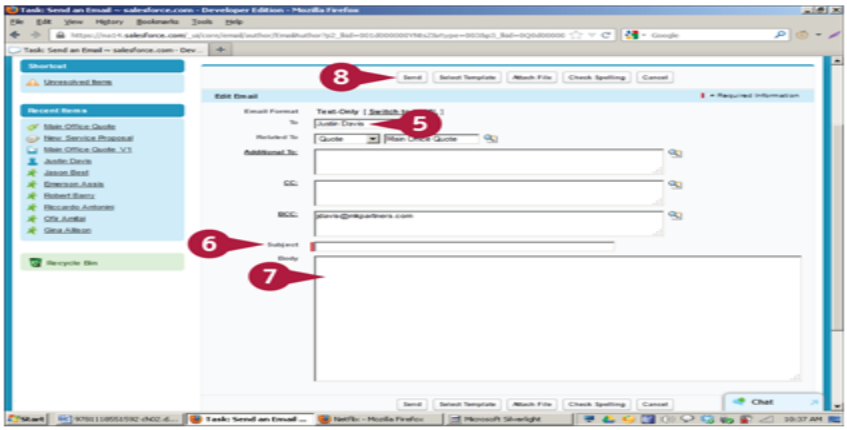
Select any opportunity to which you want to Email a quote.



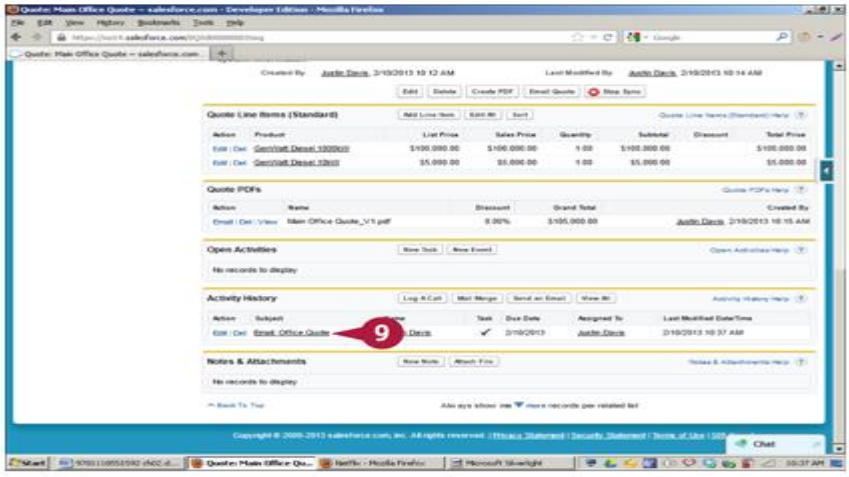
Now go to Quotes section in the opportunities and select the quote you want to send which we created in our previous Salesforce training tutorial.



Now select Email Quote as shown above. After selecting Email Quote an Email Template will be opened as shown below.



Here we are required to enter the name of the customer, Enter description subject for the Email, Enter Body text of the E-mail. We can also enter CC and BCC Email Addresses.



After enter all the details as shown above select Send Button. Emailing a quote in Salesforce has been done successfully.
After sending email Salesforce records activity on the quotes to which the email has to be sent to the customers. Other users in an organization are also able to know the data and the time of the email was sent.

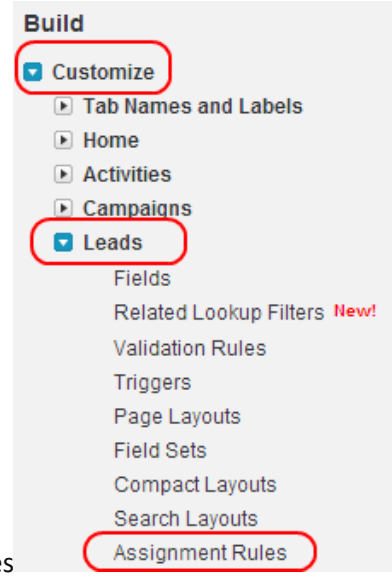
7.11 How to create a lead assignment rules in Salesforce

In previous chapter we have learned how to create lead queues in Salesforce. In this chapter we are going to create lead assignment rule in Salesforce.

Why do we want to create lead assignment rules in Salesforce?
Before creating lead assignment rules in Salesforce, we have to know what is the need of creating lead assignments in Salesforce. Suppose if a company generates huge amount of leads, managing that large pool of leads in an organization is very difficult. Generated leads in an organization must be assigned to right users to distribute the work load. To minimize the work load leads are assigned to users by using assignment rules.
What is a lead assignment rules in Salesforce?
Lead assignment rule is a feature in Salesforce which allows the administrators to define who should receive a lead with conditions. Example: - Depending on country Zip code we can assign the leads to the users who assigned to that Country.
Generally, sales manager uses lead assignment rules to assign new leads to the users in an organization like sales representatives automatically.

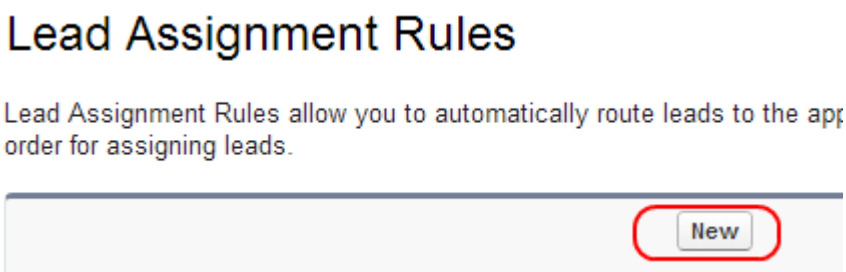
☞ Steps

To create lead assignment rules in Salesforce, follow the steps:
Go to Setup



Now go to Build=>Customize=>Leads=>Assignment Rules

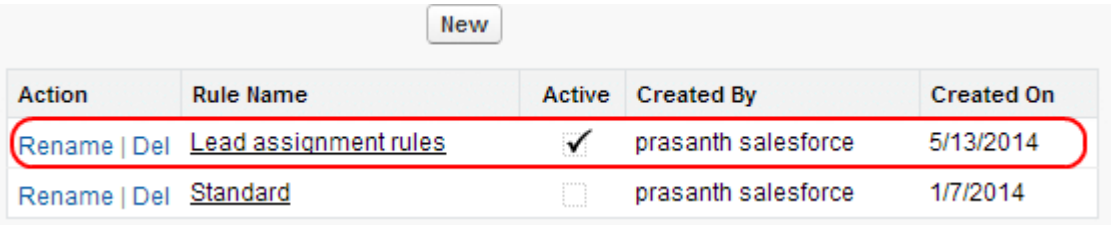
Now select create new assignment rule button.



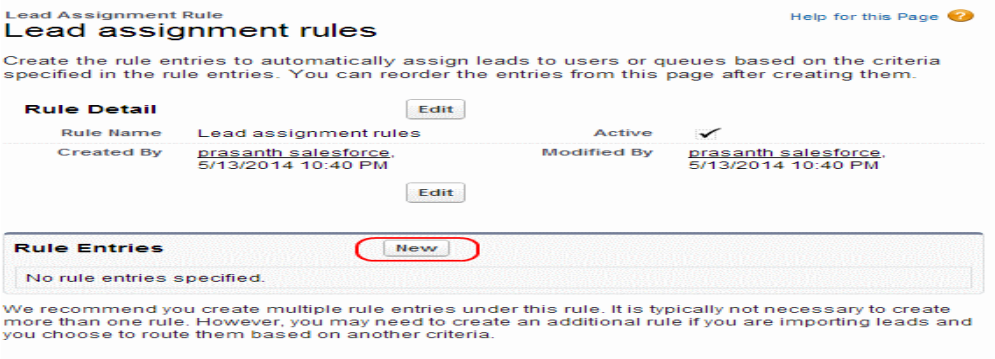
After selecting new button assignment rule page will be opened in editable mode.

Enter Rule name and make sure to check the box Set this as the active lead assignment rules check box. Assignment rule will be active only one at a time.

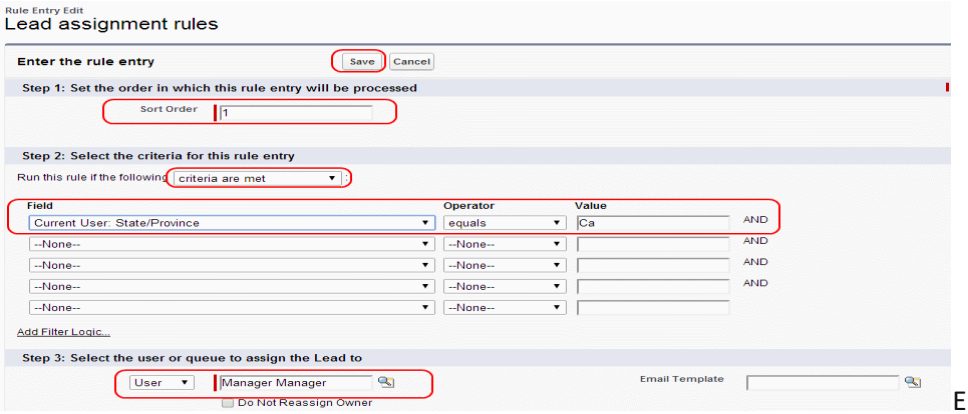
Select Save.



Now edit the Assignment rule which we created as shown above.

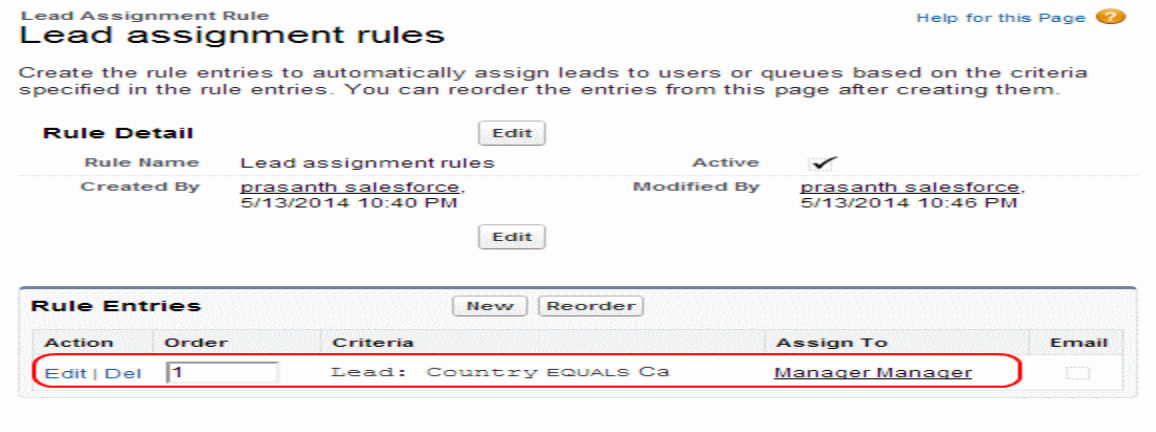


Now go to Rule Entries and select New.



Enter the details like Sort Order number, Enter Field name, operator and value as shown above, Select the user as shown above.

Select Save Button.



In this example lead assignment rule assigns a lead when the state or province value is equal to CA where the leads are from California.

7.12 How to Create a Lead Queue in Salesforce

One of the greatest challenges for a system administrator or a lead manager to manage large pool of leads in Salesforce. Due to large pool of leads Lead data base becomes unwieldy over time and we need to clean the lead data base. When involving with leads the biggest problem with leads is lead assignment, identifying duplicate leads, DE-duping the leads, transferring the leads, archiving the leads and deleting the leads.

To create a lead queue in Salesforce we use Queues to store records in Salesforce before assigning leads to the users. In Salesforce queues are used to manage leads by the users.

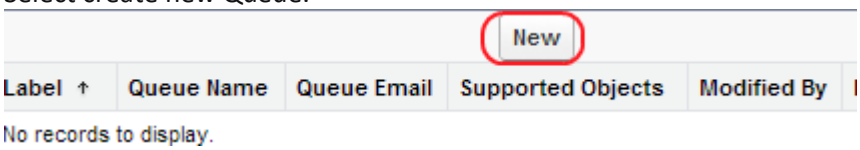
In an organization sales team is responsible for harvesting leads and setting up the lead queues.

☞ Steps

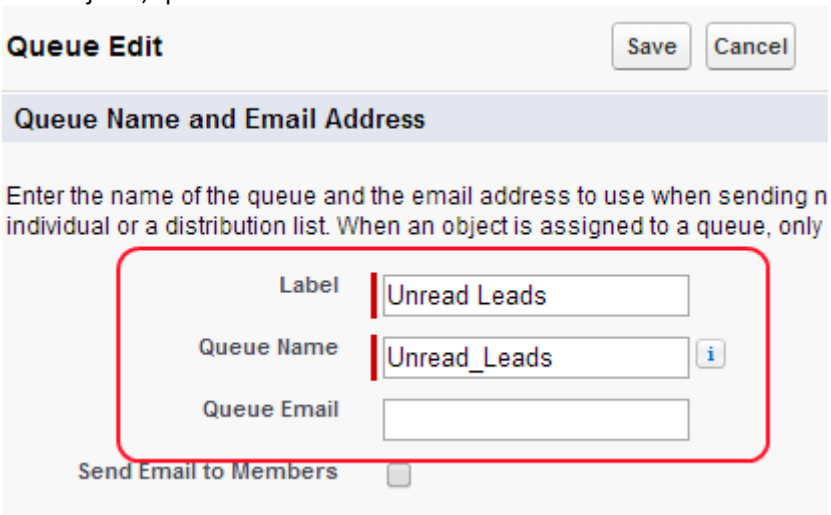
Go to Setup menu.

Now go to Administer=>Manage Users=>Queues.

Select create new Queue.

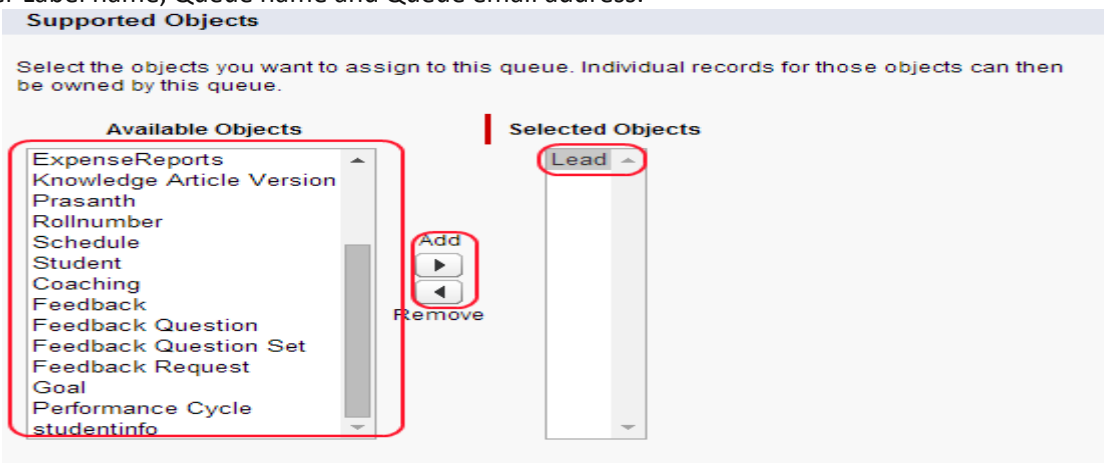


Now a Queue page will be appeared in editable mode which have three section Queuename and email address, Supported Objects, queue members as shown below.



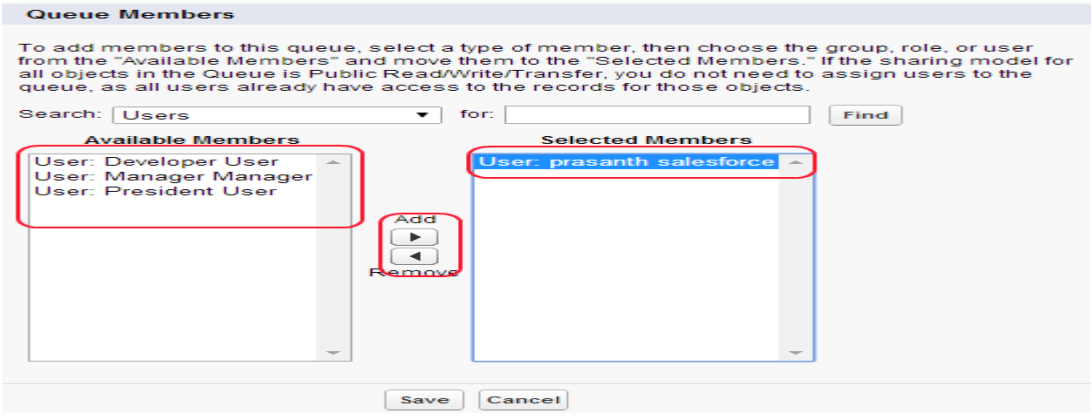
Create a Lead Queue in Salesforce

Enter Label name, Queue name and Queue email address.



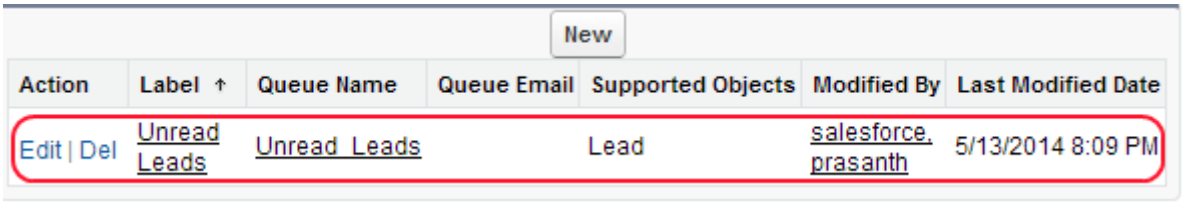
Create a Lead Queue in Salesforce

Select lead object from available objects to selected objects as shown above.



Create a Lead Queue in Salesforce

Now add members to the queue. Members can be users and roles in an organization or company. Now save the settings.



create a lead queue in Salesforce

Note: In Salesforce in addition to leads we can also add cases, knowledge articles, service contracts and custom objects to the queues. If you have a doubt like how to move records from queue to a user in Salesforce? Now go to leads and select change link text and name the owner name to which you want to move.

8.0 Salesforce Marketing Administration

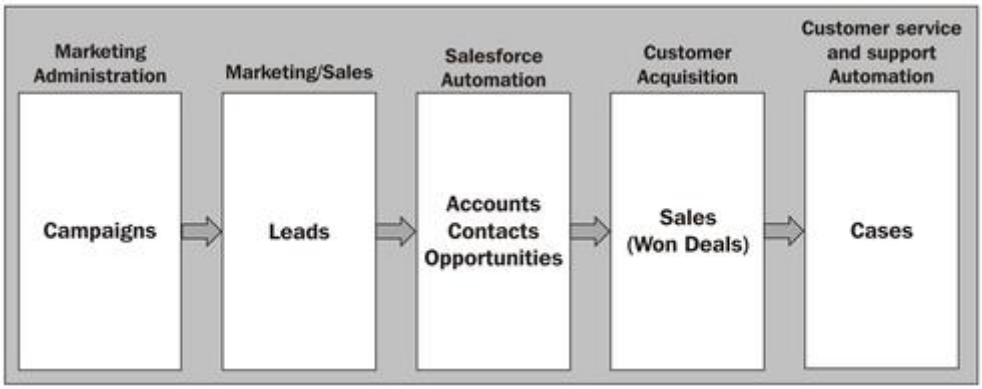
8.1 Salesforce campaign management

In this chapter we are going to learn about

- What is Salesforce campaign management?
- How to manage marketing campaigns
- Different type of campaigns.
- Marketing administration

In Salesforce CRM marketing administration is available. Marketing administration is helpful to manage marketing campaigns from initiation to lead management with the sales team in an organization. Through this marketing administration, the effectiveness of each campaigns can be analysed by leads and opportunities generated in an organization.

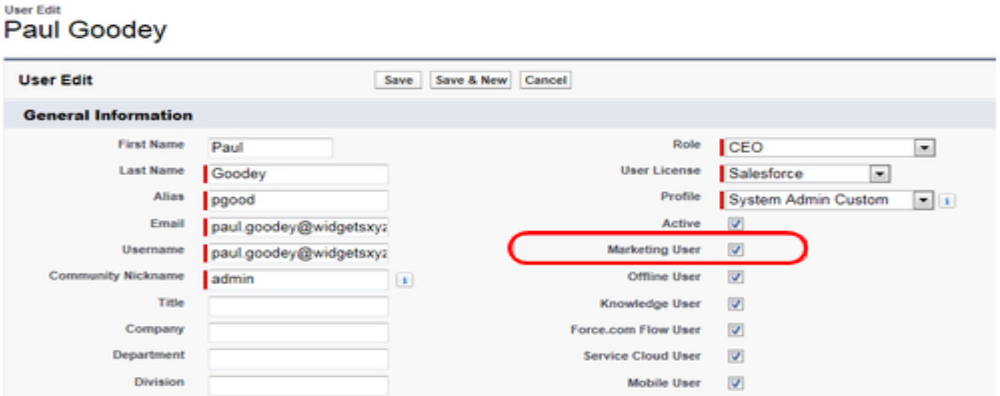
Salesforce CRM record life cycle will be as shown below.



Salesforce campaign management

Salesforce CRM life cycle process starts with marketing administration where campaigns are done through marketing administration. Different types of Marketing administration tools are used to carry Salesforce campaign managements. These Marketing administration tools links the leads which are generated in an organization.

In marketing administration different types of functions are enables like campaign management, lead management, reporting and analysis, response tracking, campaign effectiveness. We can provide different security levels when creating campaign management in Salesforce such as create, edit and delete campaigns. To set security and advanced campaigns settings user must have Marketing user Licence as shown below.



Marketing user Licence

What is Salesforce Campaign Management?

Through campaign management Salesforce allows to manage and track outbound marketing activities which are E-mail campaigns, targeted email campaigns, email marketing, Google adwords campaigns,

pay per click (PPC) management, Road show campaigns, Print advertisements and other different types of marketing campaigns.
Different types of steps are present when working with Salesforce campaigns they are:

- Campaign Planning.
- Campaign Setup.
- Campaign Creation.
- Campaign execution.
- Campaign responses.
- Campaign effectiveness.

8.2 Salesforce Campaign planning

When we are working with campaign, we have to follow six steps in Salesforce CRM hey are:

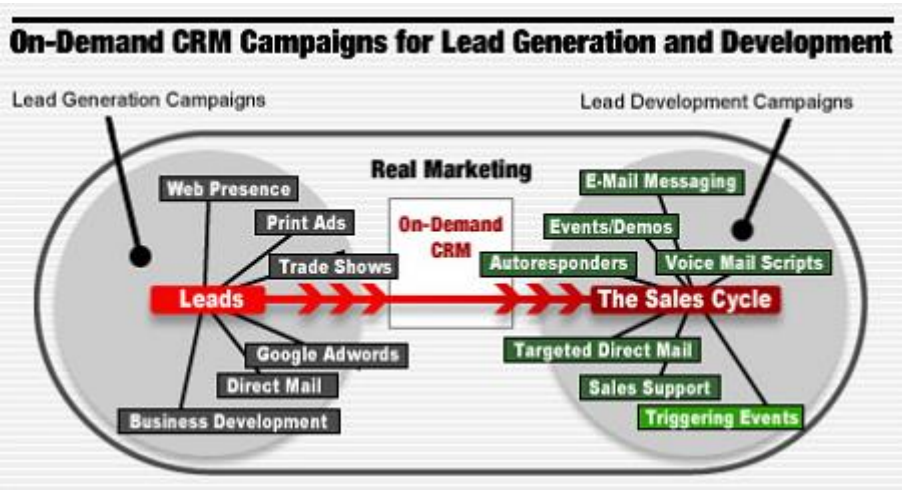
- Campaign Planning.
- Campaign Setup.
- Campaign creation.
- Campaign execution.
- Campaign responses.
- Campaign effectiveness.

Salesforce Campaign planning.

In an organization every marketing team must have overall plan about the goals and objectives of the campaigns like core processes, type of the campaign that the company is going to campaign before starting to build and run campaigns.
Different types of campaigns.

- Mass Marketing e-mails.
- Hosting a conference.
- Sending direct mails.
- Pay per Click advertisements.
- Road Shows.
- Online advertisements
- Google adwords.
- Social Networks advertisements (Facebook) and so on.

Every company must analyse the targets of the campaigning which helps in identifying whether they should be with existing customers, existing leads or new leads. With existing leads, we can use lead scoring and lead status to facilitate segmentation.



Salesforce Campaign planning

Generating Analysis and report on campaigns is very important. Create campaign fields in early stage to customize campaigns. Customizing campaigns helps us to improve the targeting and customer segmentation. Through this we can compare and analyse which types of campaigns are the most effective for our sales and marketing teams.
When targeting with new Leads, marketing team targets new leads through third party lists. These third-party lists suspects, prospects or leads can be flagged in Salesforce with a specific indicator. By flagging with different records or assigning to different queue, these lead records can be kept apart from existing prospects so that any pre-qualification or De-duplicating can be done before they are available for use in campaigns.

Important points before creating campaign in Salesforce.

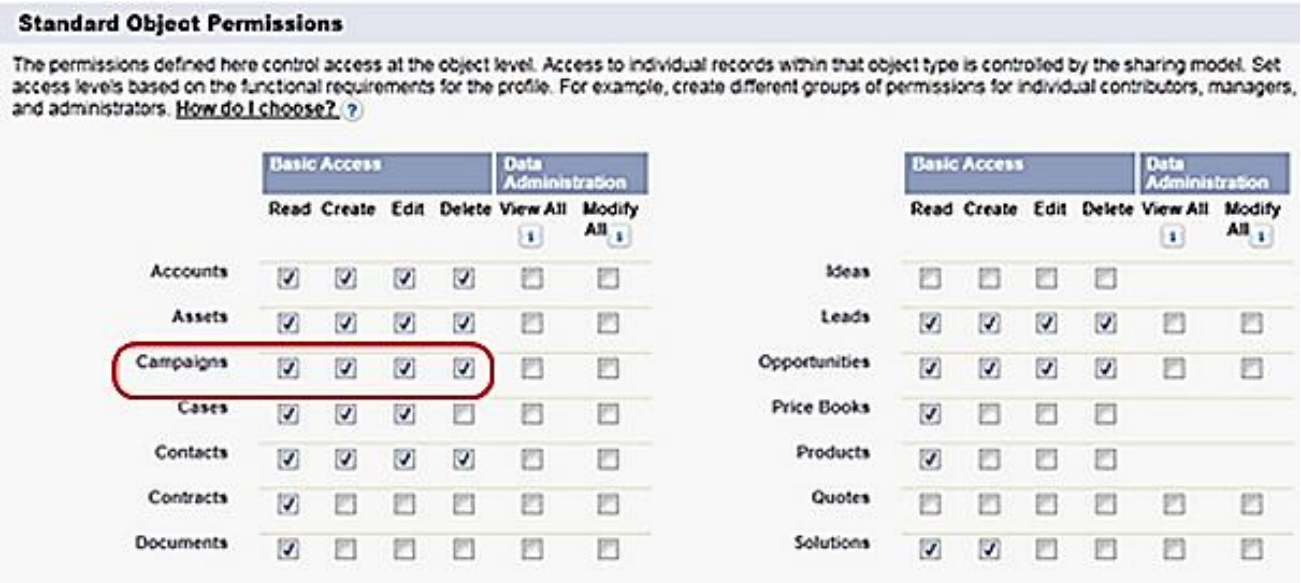
1. Lay out your entire marketing plan of projects
2. Build the basic framework and business case for a project.
3. Define statuses and success metrics for campaign responses. Success metrics measures how you determine whether the campaign was worth your company's money, time and effort.
4. Develop a detailed project plan so that important tasks get accomplished.

8.3 Configuring Campaign settings in Salesforce

In this chapter we are going to learn about Campaign settings in Salesforce. Configuring different campaign settings in Salesforce management makes us to identify who should have access to campaigns.

Campaign settings in Salesforce

In Salesforce, User must have Marketing User licence to create, edit, delete and advanced campaigning configurations. Campaign settings in Salesforce must be made as shown below.



Configuring Campaign settings in Salesforce

We can use standard fields for campaign and campaign member objects to create new custom fields in Salesforce. Different types of standard fields that are present in campaign object are:

Field	Type
Campaign Name	Text
Type	Picklist
Status	Picklist
Start Date	Date
End Date	Date
Expected Revenue	Currency
Budgeted Cost	Currency
Actual cost	Currency
Expected Response	Percentage
Num Sent	Number
Active	Check box
Description	Long Text Area
Total Leads	Number
Total Contracts	Number
Converted Leads	Number
Total Responses	Number
Total value opportunities	Currency
Total value won opportunities	Currency

Go to Setup=> Customize=>Campaigns=>Fields.

Campaign Fields

Help for this Page

This page allows you to specify the fields that can appear on the Campaign page. You can create up to 500 Campaign custom fields.

Note that deleting a custom field will delete any filters that use the custom field. It may also change the result of Assignment or Escalation Rules that rely on the custom field data.

Campaign Standard Fields

Campaign Standard Fields Help

Action	Field Label	Field Name	Data Type	Controlling Field
Edit	Active	IsActive	Checkbox	
Edit	Actual Cost	ActualCost	Currency(18, 0)	
Edit	Budgeted Cost	BudgetedCost	Currency(18, 0)	
Edit	Campaign Member Type	CampaignMemberRecordType	Lookup(Record Type)	
Edit	Campaign Name	Name	Text(80)	
Edit	Campaign Owner	Owner	Lookup(User)	
Edit	Converted Leads	NumberOfConvertedLeads	Number(9, 0)	
	Created By	CreatedBy	Lookup(User)	
Edit	Description	Description	Long Text Area(32000)	
Edit	End Date	EndDate	Date	
Edit	Expected Response (%)	ExpectedResponse	Percent(8, 2)	
Edit	Expected Revenue	ExpectedRevenue	Currency(18, 0)	
	Last Modified By	LastModifiedBy	Lookup(User)	
Edit	Num Sent	NumberSent	Number(18, 0)	
Edit	Num Total Opportunities	NumberOfOpportunities	Number(9, 0)	
Edit	Num Won Opportunities	NumberOfWonOpportunities	Number(9, 0)	
Edit	Parent Campaign	Parent	Lookup(Campaign)	
Edit	Start Date	StartDate	Date	
Replace | Edit	Status	Status	Picklist	
Edit	Total Actual Cost in Hierarchy	HierarchyActualCost	Currency(18, 0)	
Edit	Total Budgeted Cost in Hierarchy	HierarchyBudgetedCost	Currency(18, 0)	
Edit	Total Contacts	NumberOfContacts	Number(9, 0)	
Edit	Total Contacts in Hierarchy	HierarchyNumberOfContacts	Number(9, 0)	
Edit	Total Converted Leads in Hierarchy	HierarchyNumberOfConvertedLeads	Number(9, 0)	
Edit	Total Expected Revenue in Hierarchy	HierarchyExpectedRevenue	Currency(18, 0)	
Edit	Total Leads	NumberOfLeads	Number(9, 0)	
Edit	Total Leads in Hierarchy	HierarchyNumberOfLeads	Number(9, 0)	
Edit	Total Num Sent in Hierarchy	HierarchyNumberSent	Number(18, 0)	
Edit	Total Opportunities in Hierarchy	HierarchyNumberOfOpportunities	Number(9, 0)	
Edit	Total Responses	NumberOfResponses	Number(9, 0)	
Edit	Total Responses in Hierarchy	HierarchyNumberOfResponses	Number(9, 0)	
Edit	Total Value Opportunities	AmountAllOpportunities	Currency(18, 0)	
Edit	Total Value Opportunities in Hierarchy	HierarchyAmountAllOpportunities	Currency(18, 0)	
Edit	Total Value Won Opportunities	AmountWonOpportunities	Currency(18, 0)	
Edit	Total Value Won Opportunities in Hierarchy	HierarchyAmountWonOpportunities	Currency(18, 0)	
Edit	Total Won Opportunities in Hierarchy	HierarchyNumberOfWonOpportunities	Number(9, 0)	
Replace | Edit	Type	Type	Picklist	

Configuring Campaign settings in Salesforce

Standard Campaign member fields in Salesforce.

1. Campaign.
2. Contract.
3. Lead.
4. Status.

The complete set of fields will be accessed from the following path:

Setup=>Customize=>Campaigns=>Campaign Members=>Fields.

Campaign Member Standard Fields

Campaign Member Standard Fields Help

Action	Field Label	Field Name	Data Type	Controlling Field
Edit	Campaign	Campaign	Lookup(Campaign)	
Edit	City	City	Text(40)	
Edit	Company (Account)	CompanyOrAccount	Text(255)	
Edit	Contact	Contact	Lookup(Contact)	
Edit	Country	Country	Text(80)	
	Created By	CreatedBy	Lookup(User)	
	Created Date	CreatedDate	Date/Time	
Edit	Description	Description	Text(255)	
Edit	Do Not Call	DoNotCall	Checkbox	
Edit	Email	Email	Email	
Edit	Email Opt Out	HasOptedOutOfEmail	Checkbox	
Edit	Fax	Fax	Fax	
Edit	Fax Opt Out	HasOptedOutOfFax	Checkbox	
Edit	First Name	FirstName	Text(40)	
Edit	First Responded Date	FirstRespondedDate	Date	
	Last Modified By	LastModifiedBy	Lookup(User)	
	Last Modified Date	LastModifiedDate	Date/Time	
Edit	Last Name	LastName	Text(40)	
Edit	Lead	Lead	Lookup(Lead)	
Replace | Edit	Lead Source	LeadSource	Picklist	
Edit	Mobile	MobilePhone	Phone	
Edit	Phone	Phone	Phone	
Edit	Responded	HasResponded	Checkbox	
Replace | Edit	Salutation	Salutation	Picklist	
Edit	State/Province	State	Text(40)	
Replace | Edit	Status	Status	Picklist	
Edit	Street	Street	Text(255)	
Edit	Title	Title	Text(80)	
Edit	Zip/Postal Code	PostalCode	Text(20)	

These are the settings for Salesforce campaign object. In our next Salesforce tutorial we will learn about How to create Salesforce campaign.

8.4 How to create new campaigns in Salesforce CRM

Marketing professionals or marketing team uses campaigns to collect and process the leads collected from different types of campaigns in Salesforce. To track number of leads, contacts and opportunities we use campaign records. To create new campaigns in Salesforce CRM user must have Marketing User's licence and also must enable permissions on campaigns in their profiles.

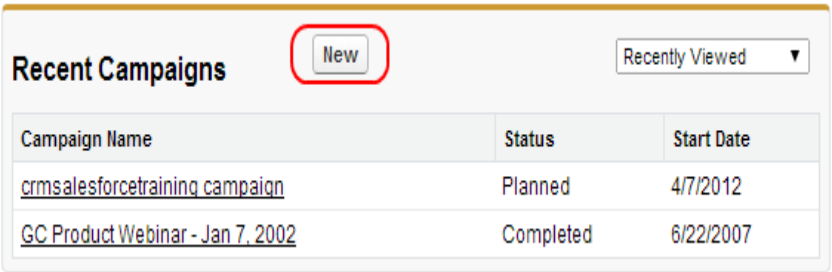
☞ Steps

How to Create new campaigns in Salesforce.com

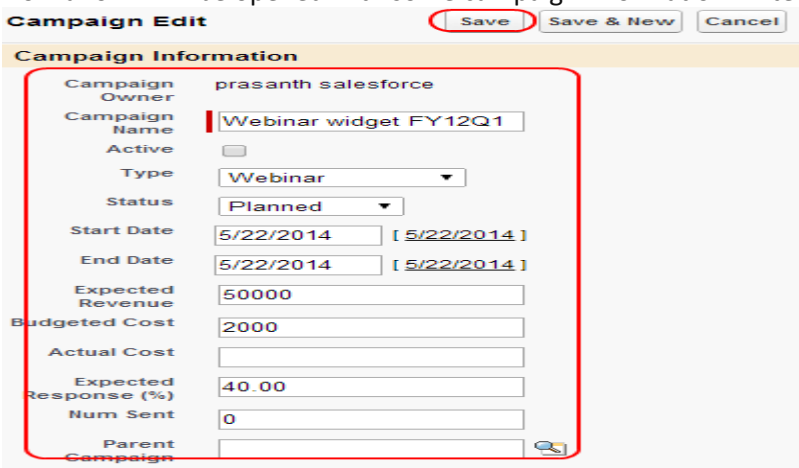
Go to Campaigns Tab.

create new campaigns in Salesforce

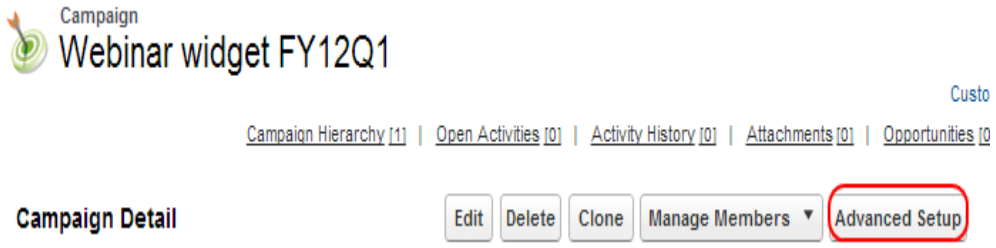
Now select new Campaign as shown below.



Now a form will be opened with some campaign information. Enter all the details.

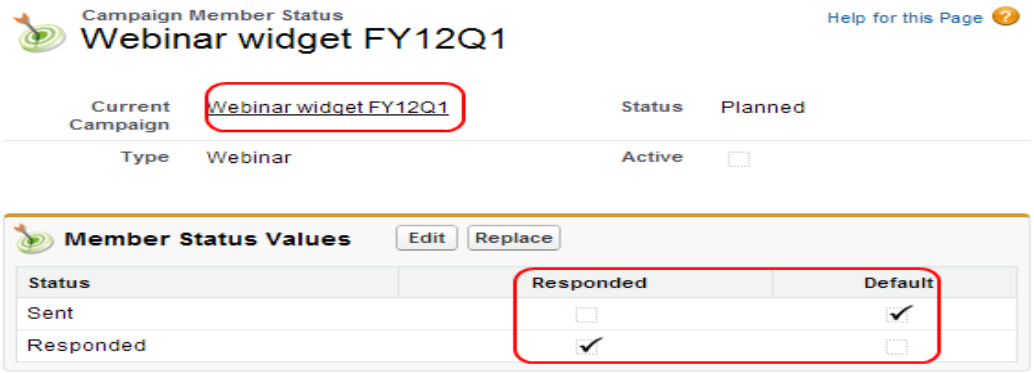


Now click Save button. Now the page will be opened in detailed view.
Campaign Member Status values.
In Salesforce CRM, new campaigns will have two default member status values they are
1. Sent 2. Responded. To set up those campaign member status values go to Advanced Tab as shown below.



Create new campaigns in Salesforce

Now the status values can be replaced with user defined values. Check boxes are checked as shown below.

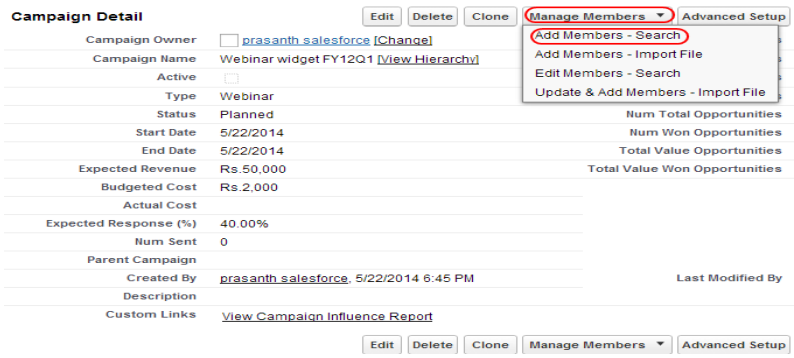


Create new campaigns in Salesforce

Multiple responded values can be created. When we have more than one responded values, they are summed together to produce the calculated field total responses.

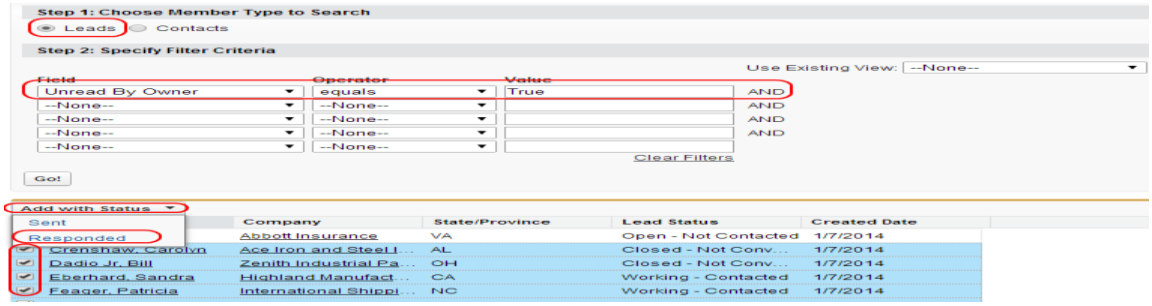
Adding Multiple Members to a campaign in Salesforce CRM.

Now we have to add multiple members to a campaign in Salesforce. Multiple members can be added from manage members in campaign page as shown below.

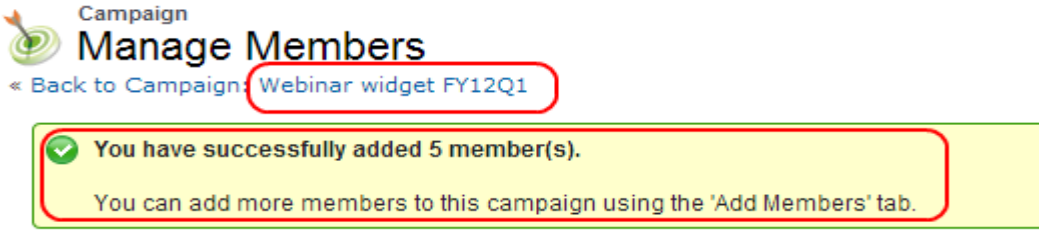


Create new campaigns in Salesforce

We can add multiple members from Manage members. Existing contacts and leads can be selected from Mange members and go to add members-Search. Use add members-Search to search members. Now a page will be opened add multiple members from lead or contact reports.

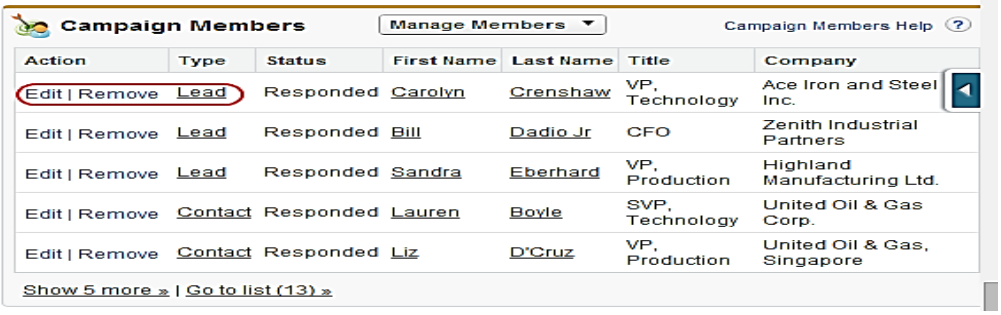


Select the required lead check boxes and select Add with status as shown. In Add with status select responded value.

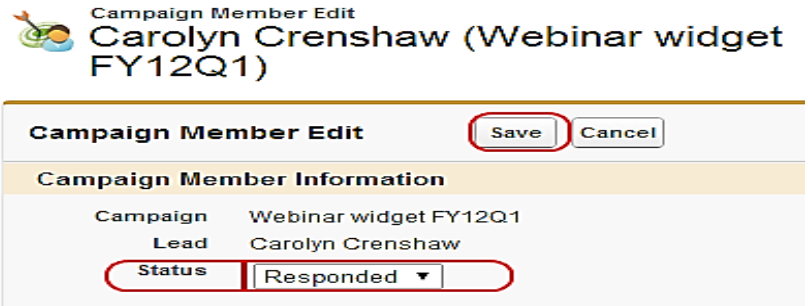


Create new campaigns in Salesforce

Confirmation message will be displayed as shown above and now go back to campaign.



In Campaign members section we can see list of members that we added to campaign as shown above. Now Edit a Lead.



Here we can change status and select Save button.

Salesforce App Builder

9.0 Organization Administration

9.1 The Salesforce setup menu

At the top of the Salesforce page, you might see the **Setup** link, as shown in the following screenshot:

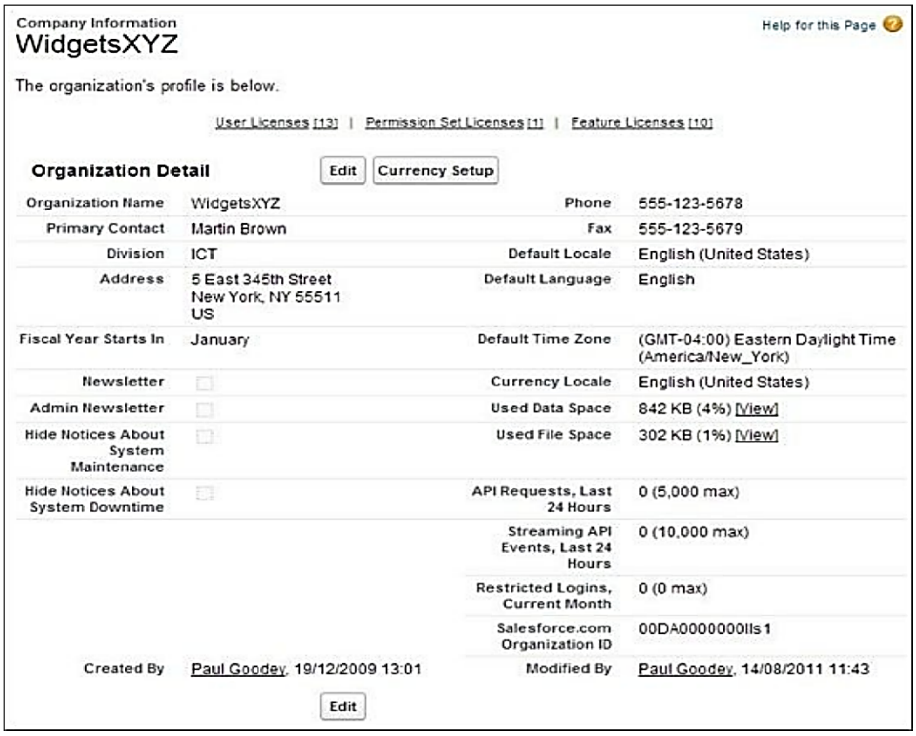
If you do not see the Setup link at the top of the Salesforce page, then click on your name and you will then see the Setup option, as shown in the following screenshot:

9.2 Establishing your company profile within Salesforce

- Company information and primary contact details
- The default language, locale, and time zone
- License information
- Fiscal year settings
- Currencies and exchange rates

Company information and primary contact details

Setup| Company Profile |Company Information.



The default language, locale, and time zone

Your Name | My Settings | Personal | Language & Time Zone

9.3 License information

There are three types of licenses:

- **User license**: A user license entitles a user to a different functionality within Salesforce and determines the profiles available to the user.
- **Feature license**: A feature license entitles a user to an additional Salesforce feature, such as **Marketing** or **Offline User.**
- **Permission set license**: A permission set licenseis used to provide a user with access to certain features that are not part of their use licenses.
- **Fiscal year settings**
  The fiscal year settings in Salesforce can be set by navigating to
  Setup | Company Profile | Fiscal Year.

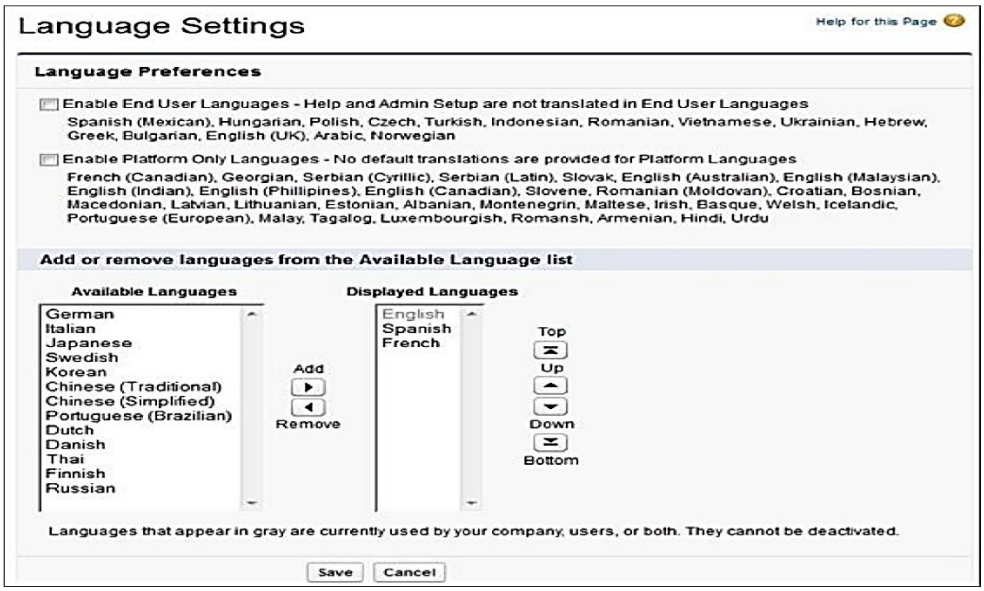
Standard fiscal years

By default, the fiscal year settings in Salesforce use the Gregorian calendar year (a 12month structure) starting from January 1 and ending on December31. If your organization follows the 12-month structure, you can use standard fiscal years. Standard fiscal years can start on the first day of any month, and you can specify whether the fiscal year is named for the starting or ending year. For example, if your fiscal year starts in April 2015 and ends in March 2016, your fiscal year setting can be either 2015 or 2016.

Custom fiscal years

If your fiscal year is more complicated than this, you can define these periods using custom fiscal years. For example, as part of a custom fiscal year, you can create a 13-week quarter represented by three periods of 4, 4, and 5 weeks instead of calendar months.

- **Language Settings**
  This feature can be set by navigating to
  Setup |Company Profile |Language Settings.



9.4 User interface

The selection of the **User Interface** option can be carried out by navigating to

Setup| Customize |User Interface

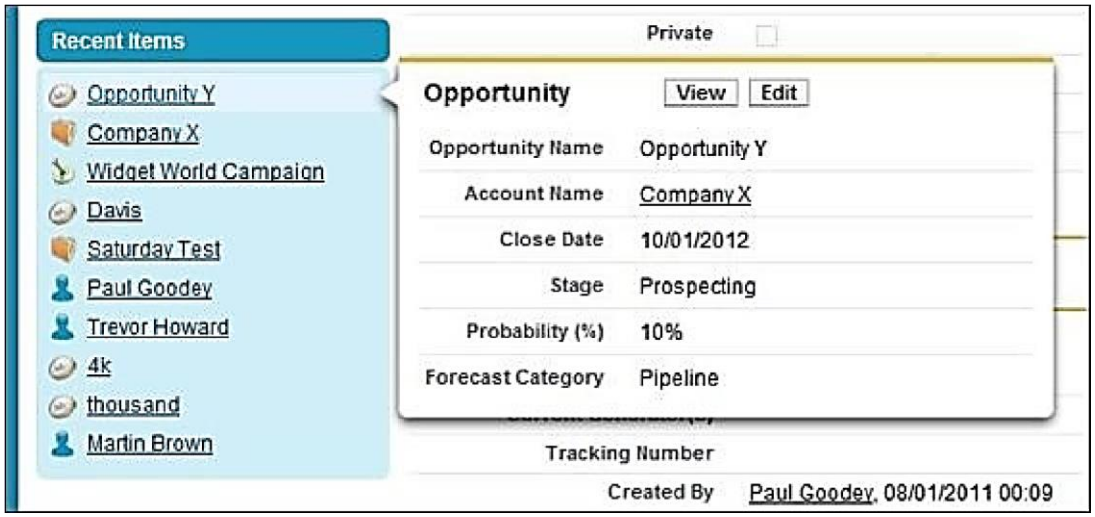
User interface settings

9.4.1 The Enable Collapsible Sections option

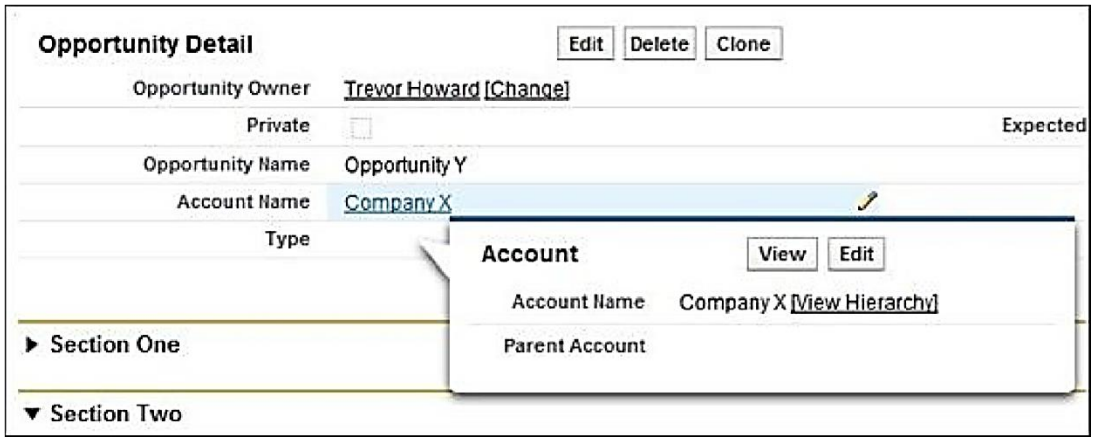
The Show Quick Create option

9.4.2 The Enable Hover Details option

In the following screenshot, we are hovering the mouse pointer over a link to a record in the **Recent Items** list on the sidebar:



In the following screenshot, we are hovering the mouse pointer over a lookup field on the record detail page:

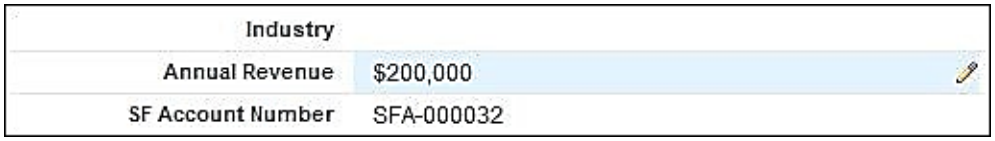


9.4.3 The Enable Related List Hover Links option

9.4.4 The Enable Inline Editing option

The **Enable Inline Editing** option allows users to change field values directly within the record Detail page, avoiding the need to load the record edit page first. By double-clicking on the

Field to be edited within the detail page, the field changes and becomes editable.



9.4.5 The Enable Enhanced Lists option

With enhanced lists enabled, users can perform the following actions:

- Create a new view or edit, delete, or refresh the existingview.
- Navigate through the list results by clicking on the first, previous, next, and last page links the bottom of the list. You can jump directly to a specific results page by entering a number in the textbox in the lower-right corner. You can also change the width of a column by dragging the right-hand side of the column heading with the mouse. Changes made to column widths
  apply to that specific list only and are recalled whenever that list is viewed. Please note that, when columns are added or removed from a list, any column width setting for that listed discarded.
- Change the order in which a column is displayed by dragging the entire column heading with your mouse to the desired position. For users who have the permission to edit the list definition, the changes are saved for all who seethe list. For users who do not have permission to edit the list definition, their changes are discarded after leaving the page.
- If inline editing is enabled for your organization, values can be entered directly into the list by double-clicking on individual field values. Users who have been granted the **Mass Edit from Lists** option on their profile can edit up to 200 records at a time.
- Enable Collapsible Sidebar
- Show Custom Sidebar Components on All Pages

We will now cover the Sidebar options in detail.

9.4.6 The Enable Collapsible Sidebar option

The collapsible sidebar gives users the ability to show or hide the sidebar on every Salesforce page where the sidebar is included. When the **Enable Collapsible Sidebar** option is selected, the collapsible sidebar becomes available to all users in your organization. However, each user can set own preference for displaying the sidebar. Users can set the sidebar to be permanently displayed, or they can collapse the sidebar and show it only when needed.

10.0 Configuration in Salesforce

10.1 Objects

Objects are a key element in Salesforce CRM, as they provide a structure for storing data and are incorporated into the interface, allowing users to interact with the data.

Similar in nature to a database table, objects have properties such as:

- Fields, which are similar in concept to a database column
- Records, which are similar in concept to a database row
- Relationships with other objects
- Optional tabs, which are user-interface components that display object data

The Objects are of two types:

- Standard Object
- Custom Object

Standard objects

Salesforce provides standard objects in the application when you sign up;

These include **Account**, **Contact**, **Opportunity**, and so on. These are the tables that contain the data records in any standard tab, such as **Accounts**, **Contacts** and **Opportunities**.

Custom objects

Custom objects are the tables you create to store your data. You can create a custom object to store data specific to your organization. Once you have custom objects and have created records for these objects, you can create reports and dashboards based on the record data in your custom object.

10.2 Fields

Fields in Salesforce are similar in concept to a database column; they store the data for the object records. An object record is analogous to a row in a database table.

The fields are also of two types:

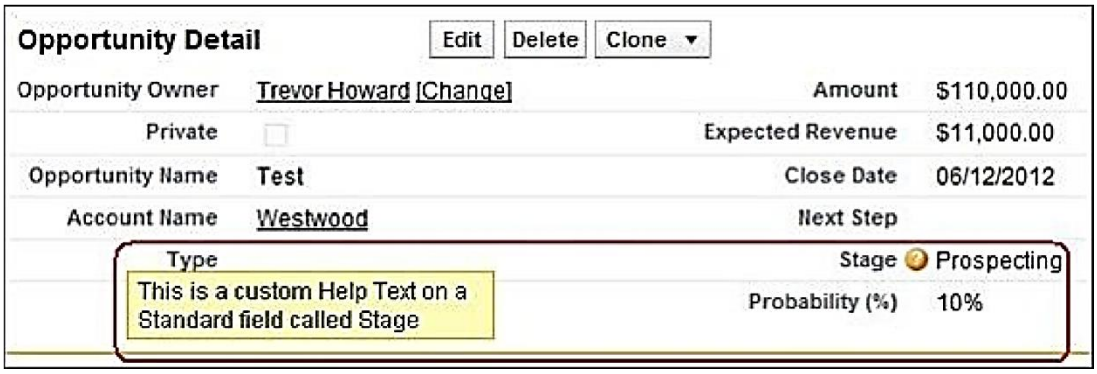
- Standard Field
- Custom field

Standard fields

Standard fields are predefined fields that are included as standard within the Salesforce CRM application. Standard fields cannot be deleted but non-required. Standard fields can be removed from page layouts, whenever necessary.

Custom fields

Custom fields are unique to your business needs and can not only be added and amended, but also deleted. Creating custom fields allows you to store the information that is necessary for your organization. Both standard and custom fields can be customized to include custom help text that helps users understand how to use the field.

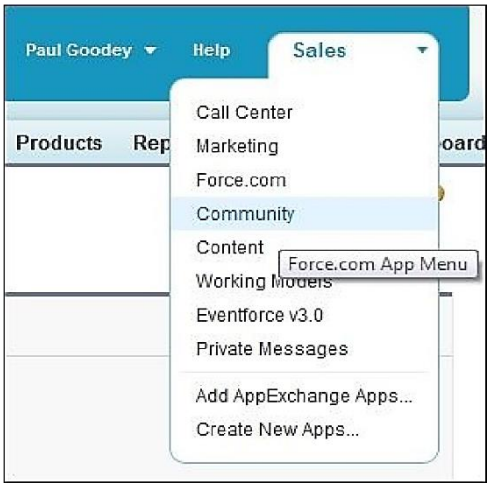


10.3 Object relationships

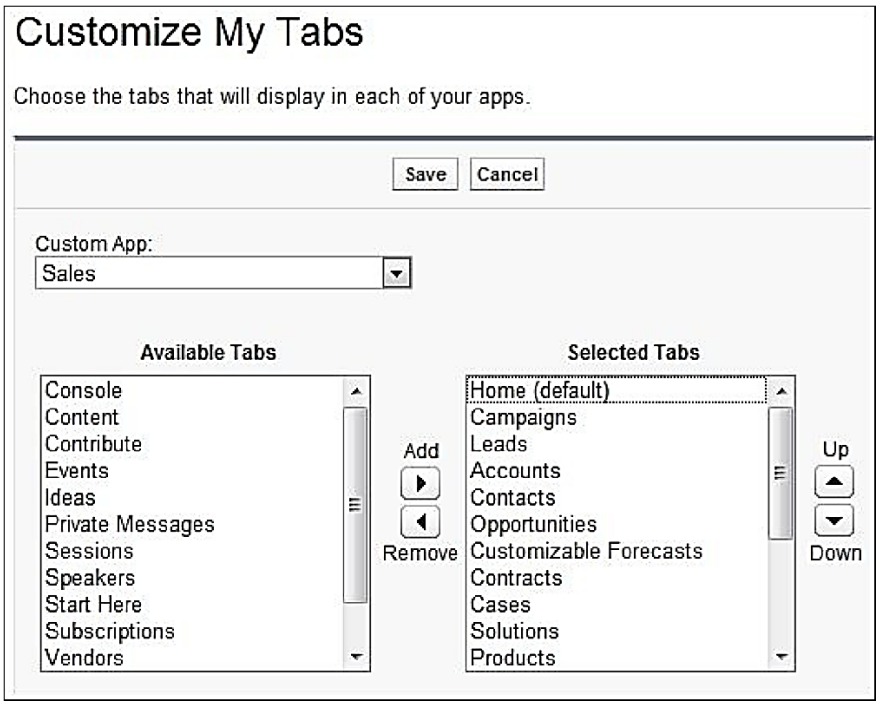
Object relationships can be set on both standard and custom objects and are used to define how records in one object relate to records in another object.
Accounts, for example, can have a one-to-many relationship with opportunities; these relationships Are presented in the application as related lists.

10.4 Apps

An app in Salesforce is a container for all the objects, tabs, processes, and services associated With a business function.
There are standard and custom apps that are accessed using the **App** menu located in the top-right corner of the Salesforce page, as shown in the following screenshot:



Standard apps
Salesforce provides standard apps such as Sales, Call Centre, and Marketing.
Custom apps
Optionally, a custom app can include a custom logo. Both standard and custom apps consist of name, a description, and an ordered list of tabs.
Tabs
A tab is a user-interface element that, when clicked on, displays the record data on a page specific to that object.
Hiding and showing tabs
To customize your personal tab settings, Your Name |My Settings| Display & Layout| Customize My Tabs.



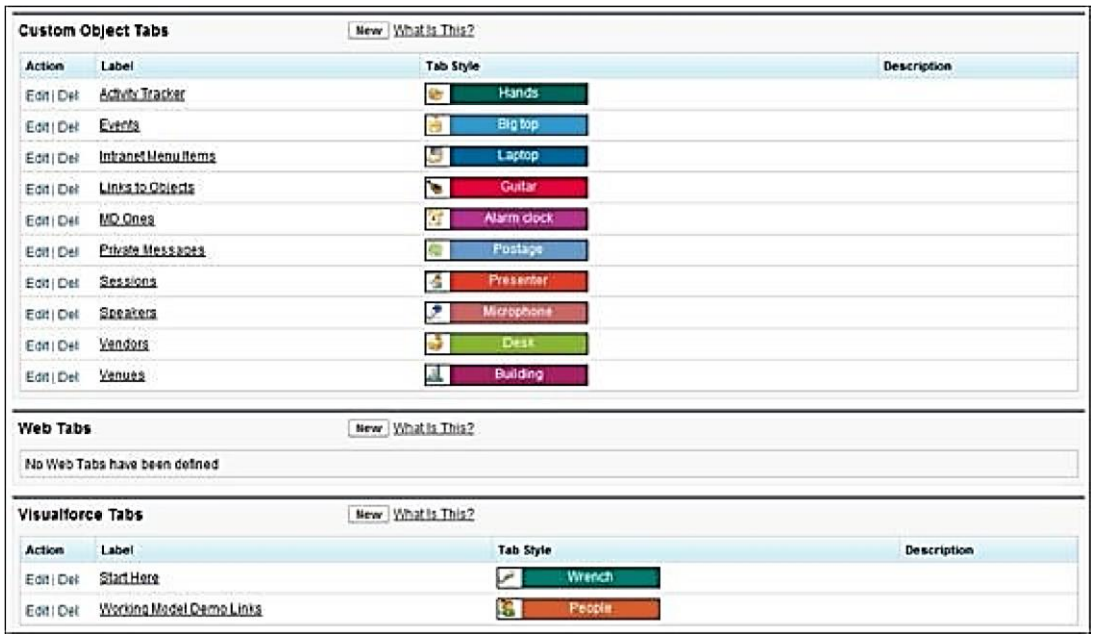
Standard tabs
Salesforce provides tabs for each of the standard objects that are provided in the application when you sign up. For example, there are standard tabs for **Accounts**, **Contacts**, **Opportunities**, and so on.

Custom tabs
You can create three different types of custom tabs: **Custom Object Tabs**, Web **Tabs and Visual force Tabs**.

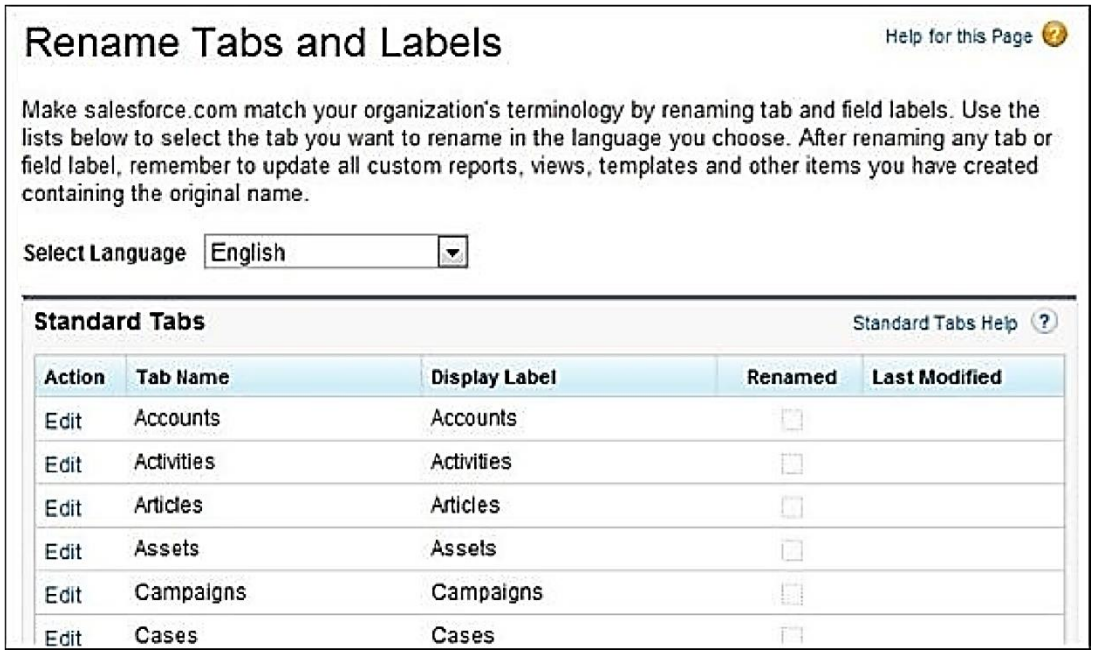
Custom Object Tabs allow you to create, read, update, and delete the data records in custom objects.

Web Tabs display any web URL in a tab within your Salesforce application.

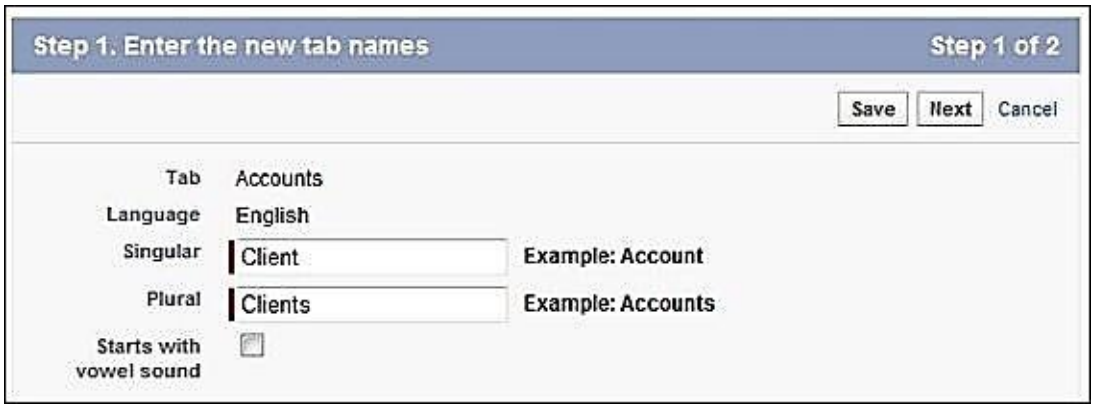
Visualforce Tabs display custom user-interface pages created using Visualforce. You can create up to 25 custom tabs in the Enterprise Edition and as many as you require in Unlimited/Performance Editions.
To create custom tabs for a custom object, navigate to **Setup|Create|Tabs**.



Renaming labels for standard tabs, standard objects, and standard fields to change standard tab, objects, and field labels, navigate to **Setup|Customize|Tabs Names and LabelsRename Tabs and Labels**. Now, select a language, and then click on **Edit** to modify the tab names and standard field labels:



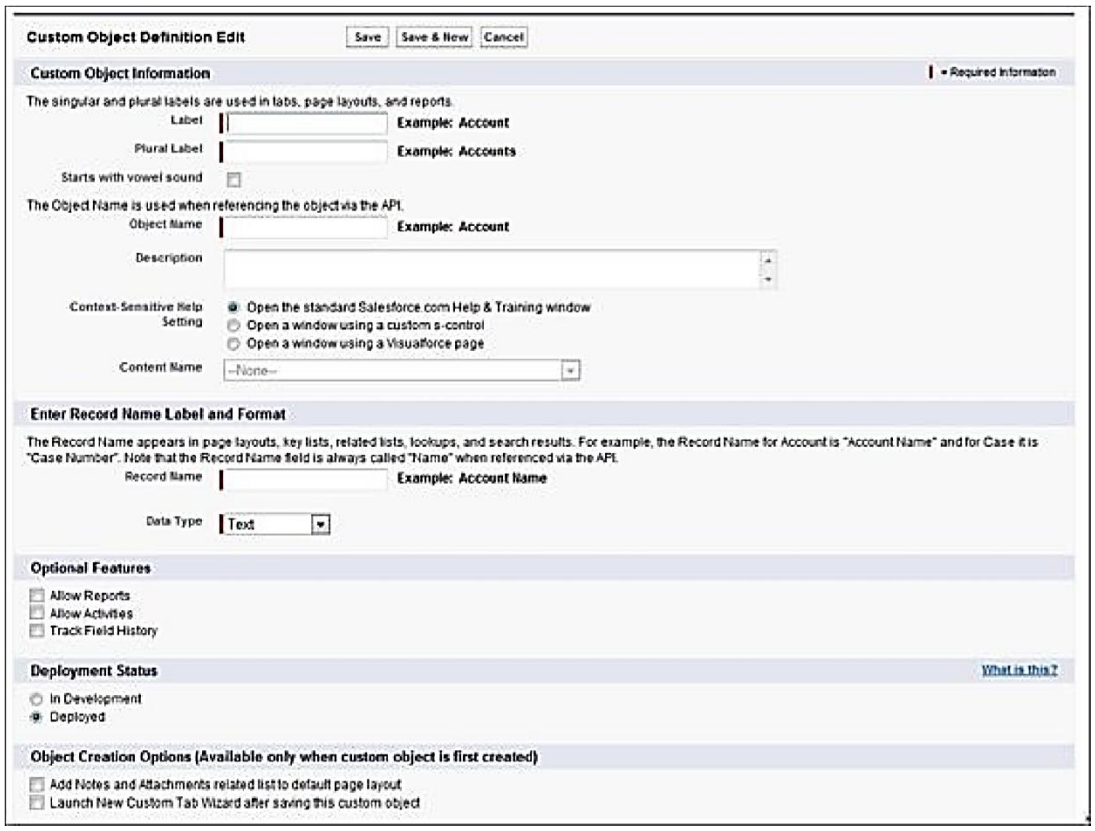
Click on **Edit** to select the tab that you wish to rename.



10.5 Creating custom objects

- You can create up to 200 custom objects in the Enterprise Edition and 2000 in the Unlimited Edition.
- Once you have created a custom object, you can create a custom tab, custom-related lists, reports, and dashboards for users to interact with the custom object data.

To create a custom object, navigate to **Setup| Create Objects**.
Click on New **Custom Object**, or click on **Edit** to modify an existing custom object. The following screenshot shows you the resulting screen:



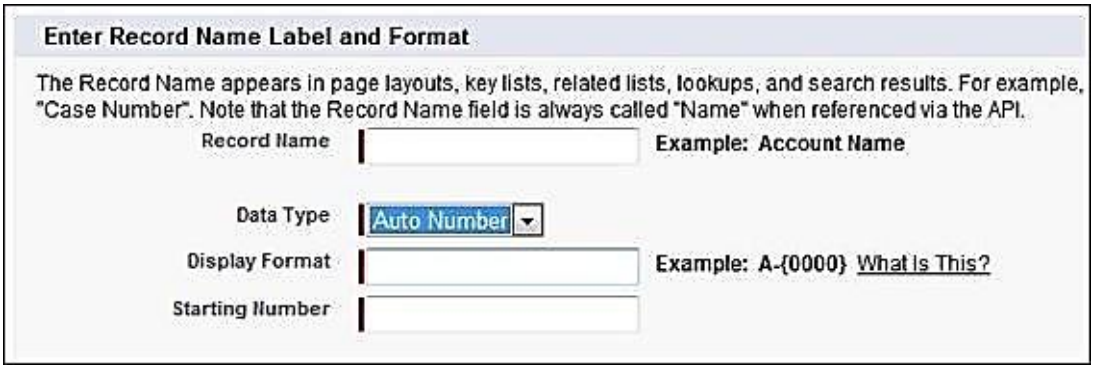
On the **Custom Object Definition Edit** page, you can enter the following:

- **Label**: This is the visible name that is displayed for the object within the Salesforce CRM user interface and is shown on pages, views, and reports, for example.
- **Plural Label**: This is the plural name specified for the object, which is used within the application in places such as reports and on tabs (if you create a tab for the object).

- **Starts with a vowel sound**: The use of this setting depends on your organization's default language and is a linguistic check that allows you to specify whether your label is to be preceded by *an* instead of *a*, for example, resulting in references to the object as *an Order* instead of *a Order*.
- **Object Name**: This is a unique name that is used to refer to the object. Here, the **Object Name** field must be unique and can only contain underscores and alphanumeric characters. It must also begin with a letter, not contain spaces, not contain two consecutive underscores, and not end with an underscore.

Description: This is an optional description of the object. A meaningful description will help explain the purpose of your custom objects when you are viewing them in a list.

- **Context-Sensitive Help Setting**: This defines what information is displayed when your users click on the **Help for this Page** context-sensitive help link from the custom object record home (overview), edit and detail pages, as well as list views and related lists. The **Help &Training** link at the top of any page is not affected by this setting; it always opens the Salesforce **Help &Training** window.
- **Record Name**: This is the name that is used in areas such as page layouts, search results, key lists, and related lists, as shown next.
- **Data Type**: This sets the type of field for the record name. Here, the data type can be either text or **Auto Number**. If the data type is set to be **Text**, then, when a record is created, users must enter a text value, which does not need to be unique. If the data type is set to **Auto Number**, it becomes a read-only field whereby new records are automatically assigned a unique number.



- **Allow Activities**: This allows users to include tasks and events related to the custom object records that appear as a related list on the custom object page.
- **Track Field History**: This enables the tracking of data-field changes on the custom object records, such as who changed the value of a field and when it was changed. Field history tracking also stores the value of the field before and after the edit. You can set field history tracking for a maximum of 20 fields for Enterprise, Unlimited, and Performance Editions.
- **Deployment Status**: This indicates whether the custom object is now visible and available for use by other users.
- **Add Notes & Attachments**: This setting allows your users to record notes and attach files to the custom object records. When this is specified, a related list with the **New Note** and **Attach File** buttons automatically appears on the custom object record page where your users can enter notes and attach.
- **Launch the New Custom Tab Wizard**: This starts the custom tab wizard after you save the custom object. The **New Custom Tab Wizard** option is only available when you create a new object.

10.6 Object Limits

Standard objects

To access the standard **Object Limits** page, navigate to **Setup** | **Customize**. Click on the name of the desired standard object, and then click on **Limits**, as shown in the following screenshot (for the **Account** object):

Account Limits

Help for this Page

These limits apply to the setup of this object in your organization. Some limits may vary by object.

Object Limits

Item	Usage	Limit	% used	Message
Custom Fields	12	500	2%	
Rollup Summary Fields	1	25	4%	
Custom Relationship Fields	0	40	0%	
Active Workflow Rules	1	50	2%	
Total Workflow Rules	1	500	0%	
Approval Processes	1	500	0%	
Active Lookup Filters	0	5	0%	
Active Validation Rules	0	100	0%	
VLOOKUP Functions	0	10	0%	
Sharing Rules (Both Owner- and Criteria-based)	0	300	0%	
Sharing Rules (Criteria-based Only)	0	50	0%	

Here, you can see usage details for these: **Custom Fields**, **Rollup Summary Fields**, **Custom Relationship Fields**, **Active Workflow Rules**, **Total Workflow Rules**, **Approval Processes**, **Active Lookup Filters**, **Active Validation Rules**, **VLOOKUP Functions**, **Sharing Rules (Both Owner- and Criteria-based)**, and **Sharing Rules (Criteria-based Only)**.

Custom objects

To view information about the usage of various fields and rules that have been created on a custom object, you can access the **Object Limits** window displayed on a custom-object definitions-related list at the bottom of a custom object definition page.

Creating custom object relationships

The considerations to be observed when creating object relationships are as follows:

- Create the object relationships as a first step before you start to build the custom fields, page layouts, and any related list
- The **Related To** entry cannot be modified after you have saved the object relationship

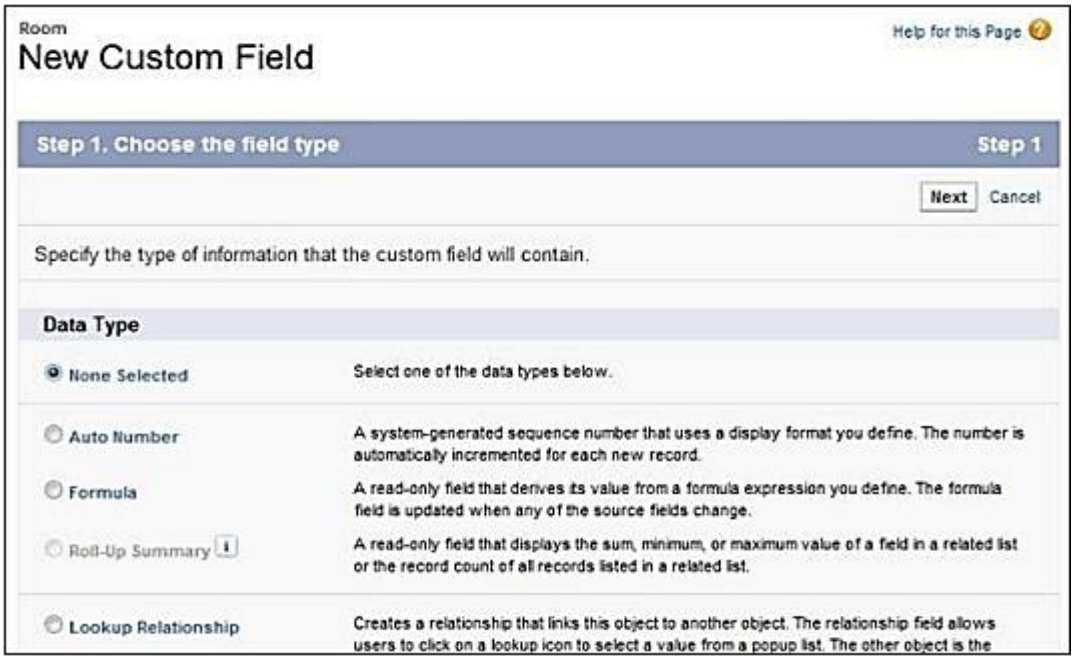
Note

- Each custom object can have up to two master-detail relationships and up to 25 total relationships.
- When you're planning to create a master-detail relationship on an object, be aware that it can only be created before the object contains record data.
- Clicking on **Edit List Layout** allows you to choose columns for the key views and lookups
- The **Standard Name** field is required for all custom object-related lists and also for any page layouts.

Creating custom fields

Adding custom fields can be carried out by navigating to the field area of the appropriate object:

- For standard objects, navigate to **Setup** | **Customize**. Now, select the appropriate object from the **Customize** menu, click on **Fields**, and then click on **New** in the **Custom Fields & Relationships** section of the object page.
- For custom task and event fields, navigate to **Setup** | **Customize** | **Activities** | **Activity Custom Fields**. Now, click on the **New** button.
- For custom objects, navigate to **Setup** | **Create** | **Objects**. Now, select one of the custom objects in the list. Next, click on **New** in the **Custom Fields & Relationships** dependencies and field history tracking.



For relationship fields, choose the object that you want to associate with it.

Room
New Relationship
Help for this Page

Step 2. Choose the related object — Step 2

Previous | Next | Cancel

Select the other object to which this object is related.

Related To: --None--

Step 3. Establish field-level security — Step 3 of 4

Previous | Next | Cancel

Field Label: Air Conditioned
Data Type: Checkbox
Field Name: Air_Conditioned
Description:

Select the profiles to which you want to grant edit access to this field via field-level security. The field will be hidden from all profiles if you do not add it to field-level security.

Field-Level Security for Profile	Visible	Read-Only
Authenticated Website		
Contract Manager		
Custom: Marketing Profile		
Custom: Sales Profile		
Custom: Support Profile		
Customer Portal Manager		

Click on **Next** and choose the page layouts to which you would want to add the new field, as shown in the following screenshot.

Step 4. Add to page layouts — Step 4 of 4

Previous | Save & New | Save | Cancel

Field Label: Air Conditioned
Data Type: Checkbox
Field Name: Air_Conditioned
Description:

Select the page layouts that should include this field. The field will be added as the last field in the first 2-column section of these page layouts. The field will not appear on any pages if you do not select a layout.

To change the location of this field on the page, you will need to customize the page layout.

Add Field	Page Layout Name
☑	Room Layout

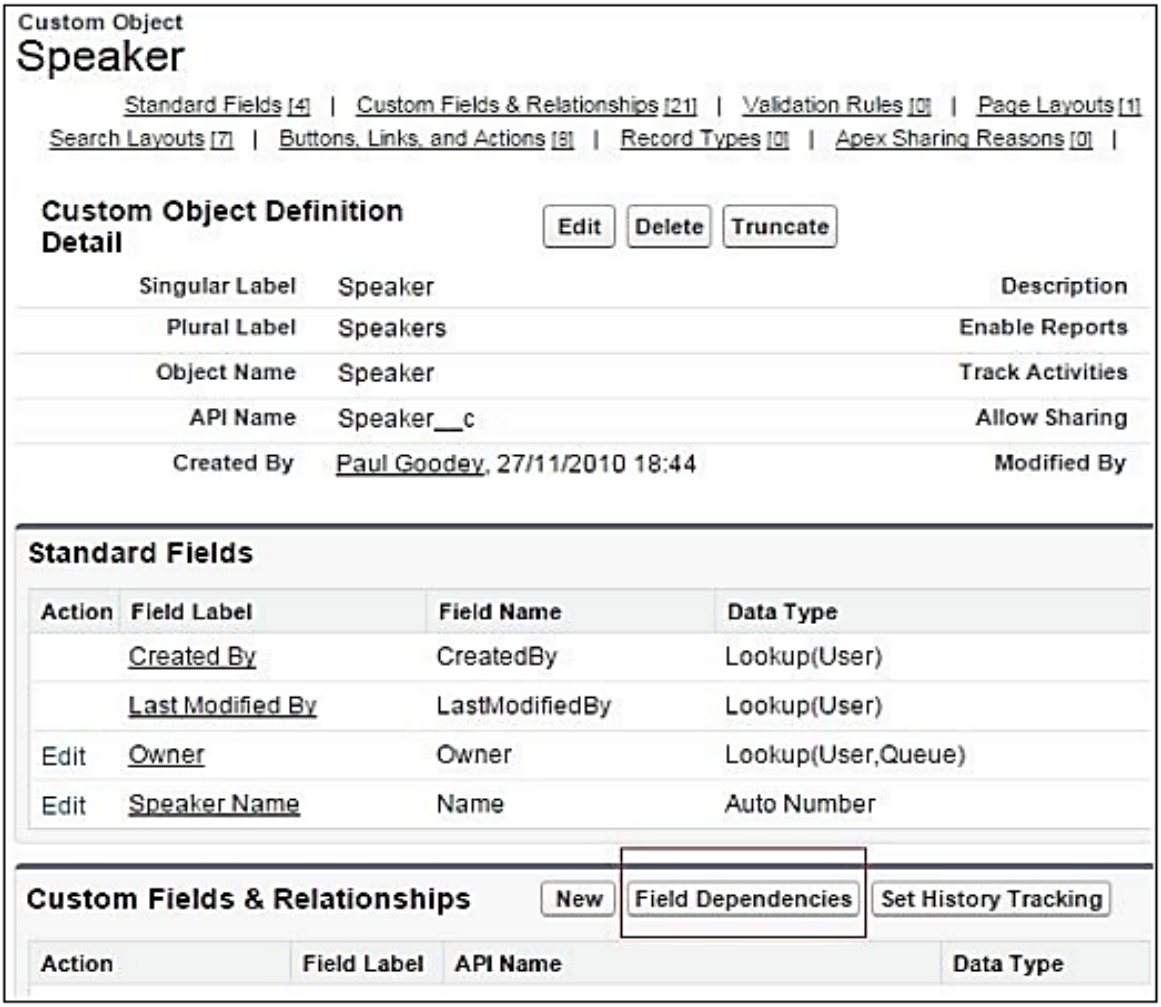
Dependent picklists

Dependent picklists are picklists (including multi-select picklists) in which the values available in the picklist depend on the value of another field, which is called the controlling field.

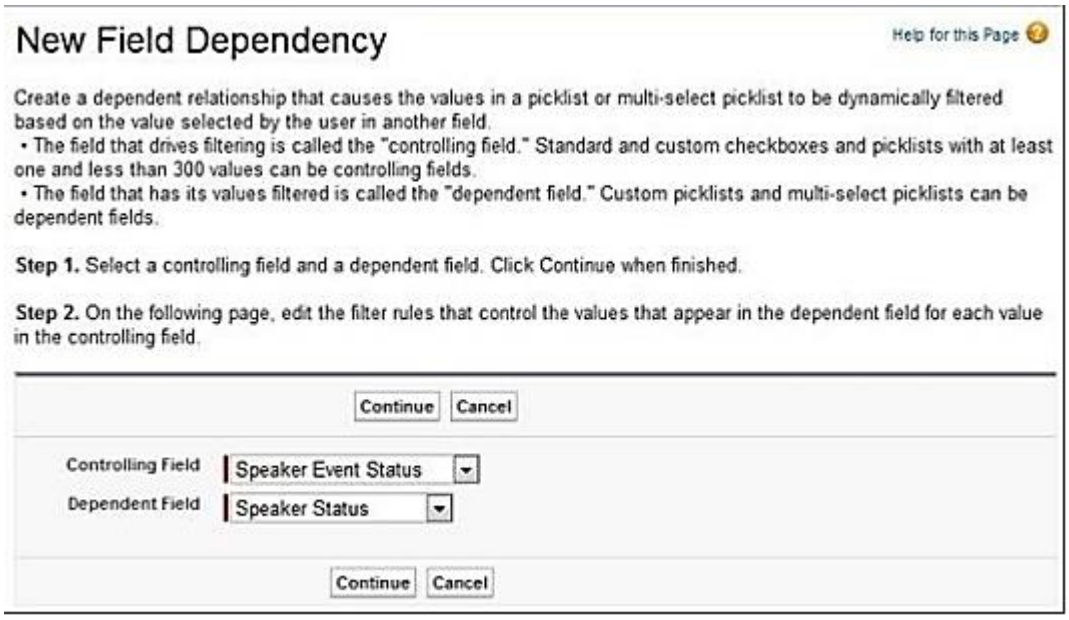
- Controlling fields can be any picklist or checkbox field within the same record
- Controlling fields that are picklists are fields with at least one or fewer than 300 value.

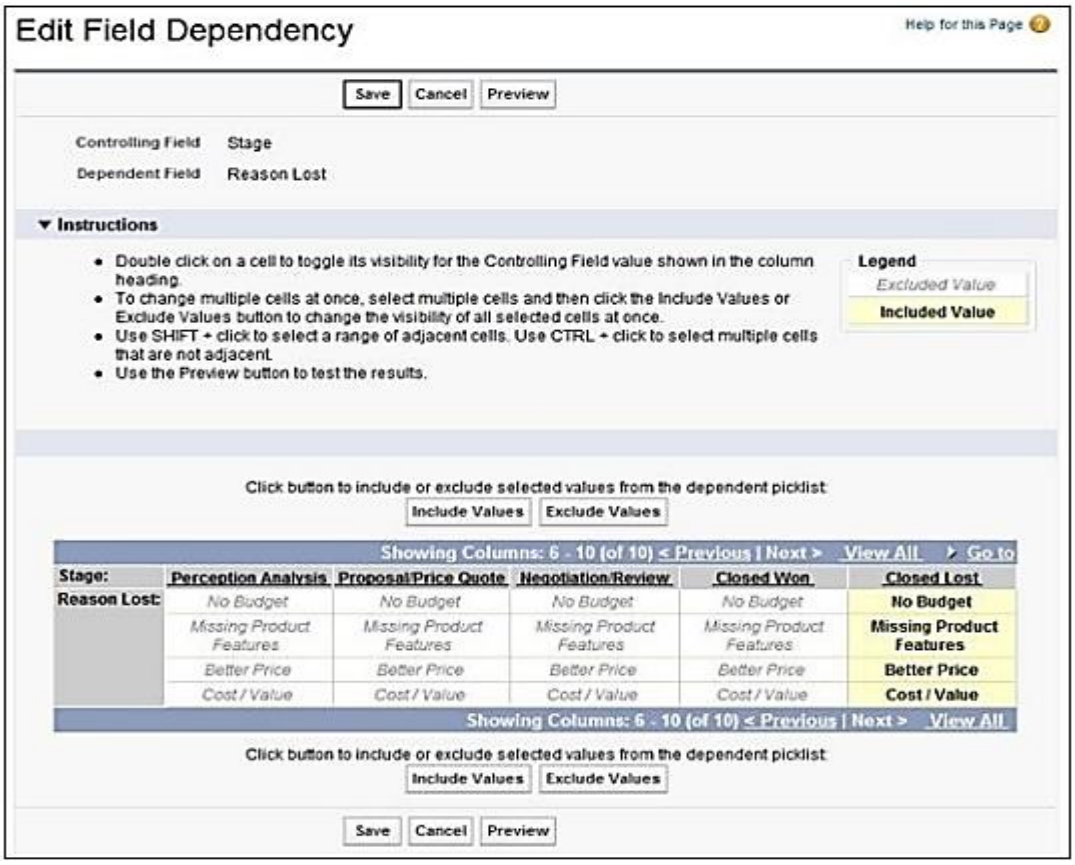
To define a dependent picklist, navigate to the field's area of the appropriate object.

- For standard objects, this is carried out by navigating to **Setup** | **Customize** | (select the appropriate standard object) | **Fields**. Then, click on **Field Dependencies**.
- For custom objects, navigate to **Setup** | **Create** | **Objects** | (select the appropriate custom object). Then, click on **Field Dependencies**.
- Now, click on **New**, choose a controlling field and dependent field, and then click on **Continue**.



Now, click on **New** to navigate to the **New Field Dependency** screen and then choose **Controlling Field** and **Dependent Field**, as shown in the following screenshot:





Note

Please note the following points:

- Checkbox fields can be controlling fields but not dependent fields.
- You can set default values for controlling fields but not for dependent picklists
- Multiselect picklists can be dependent picklists but not controlling fields.
- Standard picklist fields can be controlling fields but not dependent fields.
- Custom picklist fields can be either controlling or dependent fields.
- The maximum number of values allowed in a controlling field is 300.

10.7 Building relationship fields

Lookup relationship options
when you create a lookup field on an object, you can choose whether the lookup field is required or is optional. If it is set as optional, you can choose one of the following three actions if the lookup record is deleted:

- Clear the value of this field
- Don't allow the deletion of the lookup record that's part of a lookup relationship
- Delete this record as well

Master-detail relationship options
When you create a **Master-detail** field on an object, you can choose **Allow Reparenting Option**.

Allow Reparenting Option
By default, records in master-detail relationships cannot be reparented. However, you can allow child records in a master-detail relationship to be reparented to a different parent by selecting **Allow Reparenting Option** in the master-detail relationship definition.

10.8 Building formulas

Custom formula fields require additional settings as specified by the Salesforce CRM application. These are carried out using the following actions and steps:
1. Create the **Formula** field.
2. Choose the data type for the field based on the output of the calculation.
3. Enter the number of decimal places for currency, number, or percent data types.

Basic formula
To create a basic formula that passes specific Salesforce data, select the **Simple Formula** tab, choose the field type in the **Select Field Type** drop-down list, and choose one of the fields listed in the **Insert Field** drop-down list.

To insert an operator, choose the appropriate operator icon from the **Insert Operator** drop-down list. Here, you can choose between these operators: **+ 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**, and **|| Or**.

Advanced formula
The basic formula feature is quite restricted and you will most likely seek to create more complicated formulas that can be performed by selecting the **Advanced Formula** tab. Within this tab, click on **Insert Field**, choose a field, and then click on **Insert**.
You can now include merge fields along with advanced operators as well as functions, which are prebuilt Salesforce CRM formulas that you can invoke and pass your input values to.

Tip

Function description and example usage
Select a function and click on **Help** to view a description and examples of formulas using that function.

The following are the steps to create a new custom field:

1. Click on **Check Syntax** to check your formula for errors.
2. Enter a description of the formula in the **Description** box.
3. If your formula references any number, currencies, or percent fields, choose an option to handle blank fields. To give any blank fields a zero value, choose **Treat blank fields as zeros**. To leave these fields blank, choose **Treat blank fields as blanks**.
4. Click on **Next**.
5. Set the field-level security to determine whether the field should be visible for specific profiles or not, and click on **Next**.
6. Choose the page layouts that should display the field. The field is added as the last field in the first two-column section on the page layout. For user custom fields, the field is automatically added to the bottom of the user detail page.

Note
Formula fields are automatically calculated. Therefore, they are not visible on edit pages and are read-only on record detail pages. Formula fields do not update last modified date fields

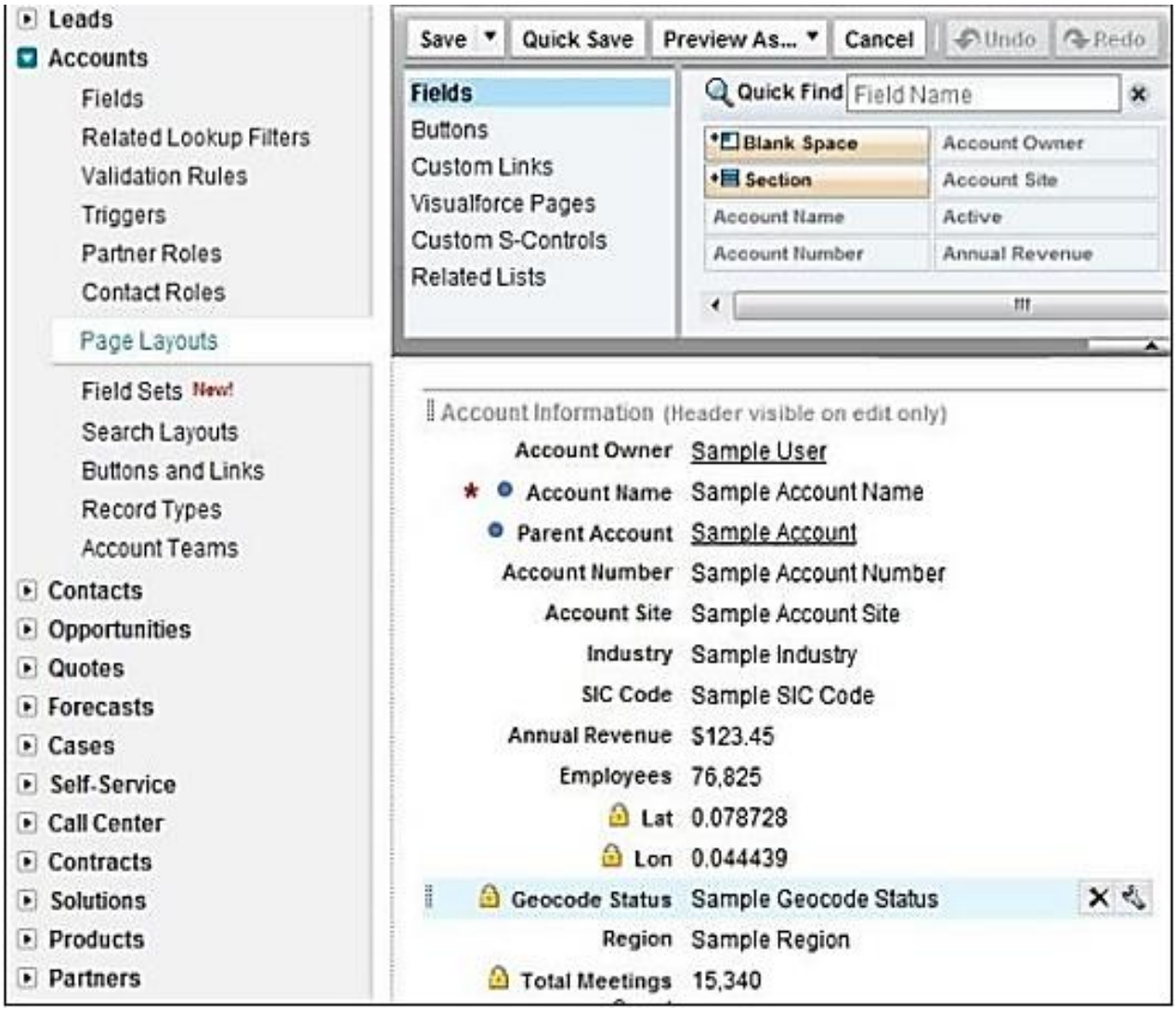
7. Click on **Save** to finish or on **Save & New** to create more custom fields.

Note
Formula fields have character and byte size limits and cannot contain more than 3,900 characters

10.9 Page layouts

Page layouts are used to organize the display of fields, buttons, custom links, inline Visualforce pages, Report Charts, and related lists on an object detail or edit page. They are used to establish unique layouts for different business scenarios. The displayed fields within a related list are controlled by the page layout; the name of the related list is determined by the lookup/master-detail relationship on the related object.

Page layouts are comprised of sections that contain buttons, fields, related lists, and customer links that can be edited using the enhanced page layout editor, as shown in the following screenshot. Here, we are showing how we can edit the properties of the **Account Site** field.



Within the field sections, the user interface can be used to make a field required or read only, as shown in the following screenshot:

Related lists

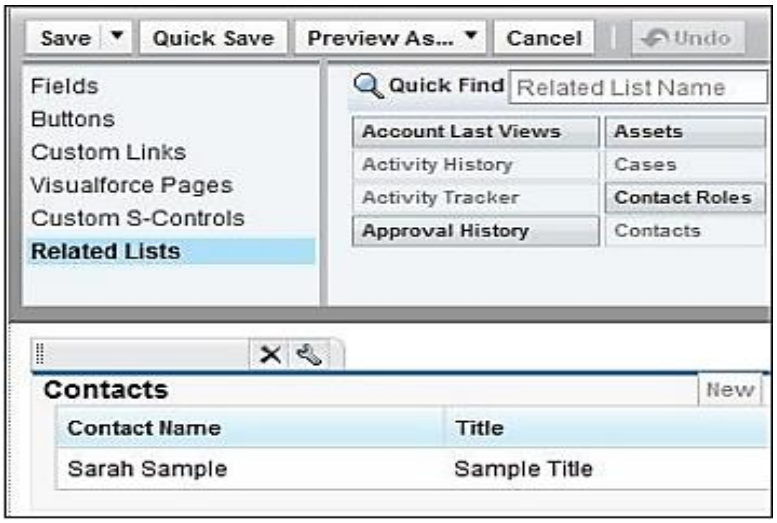
Related lists are displayed on the lower portion of the object detail page to display the related record details. Related lists show you the object records that are associated with that record.

From a related list, you can:

- Click on the object record name to view detailed information
- Click on **Edit** or **Del** to edit or delete the object record
- Click on **New** to create a new object record that is associated with the record you are viewing

To define whether an object can be related to another type of record, you would use either a master-detail or a lookup relationship.

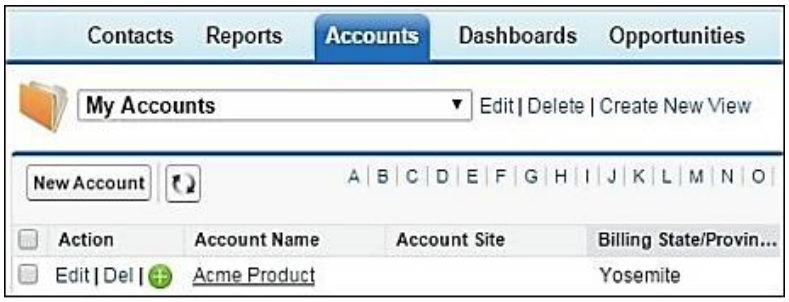
Here, we show you how editing a page layout for the account object enables the arrangement and configuration of any related list:



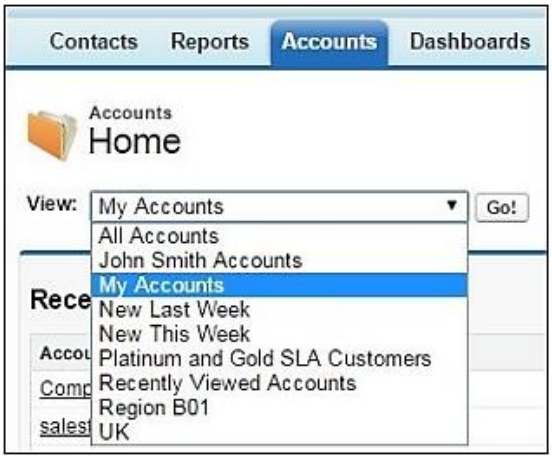
The following screenshot shows you the results of changing the related lists in the page layout editor screen when navigating to the **Account** detail page:

10.10 List views

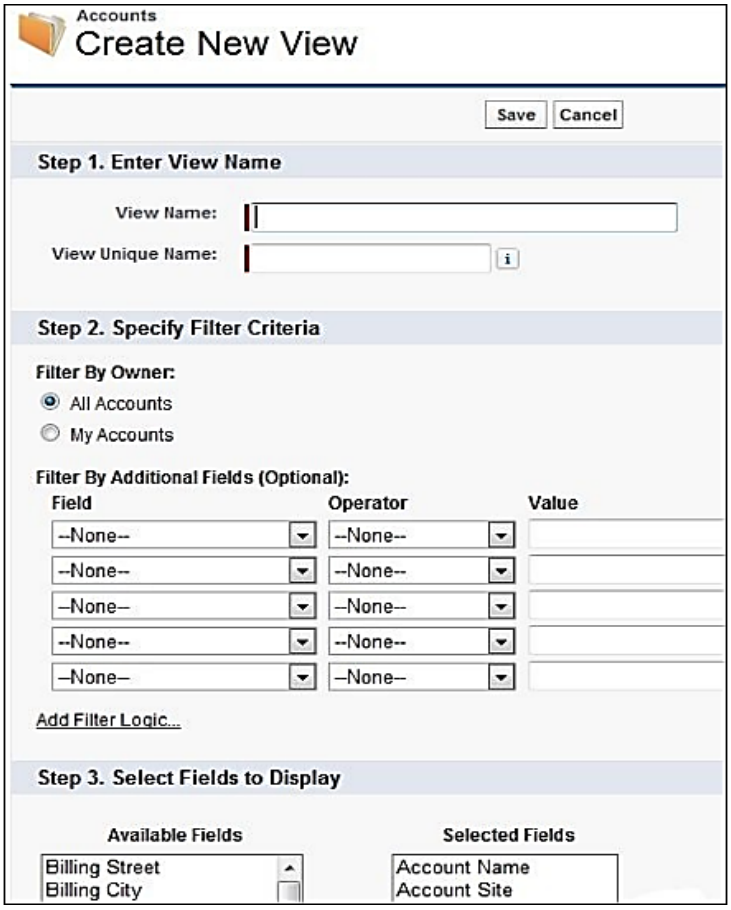
When you click on a tab, the **Accounts** tab for instance, you will be shown the **My Accounts** field in that view. This is termed as a list view and can be seen as shown in the following screenshot:



Other list views can be selected from the picklist.

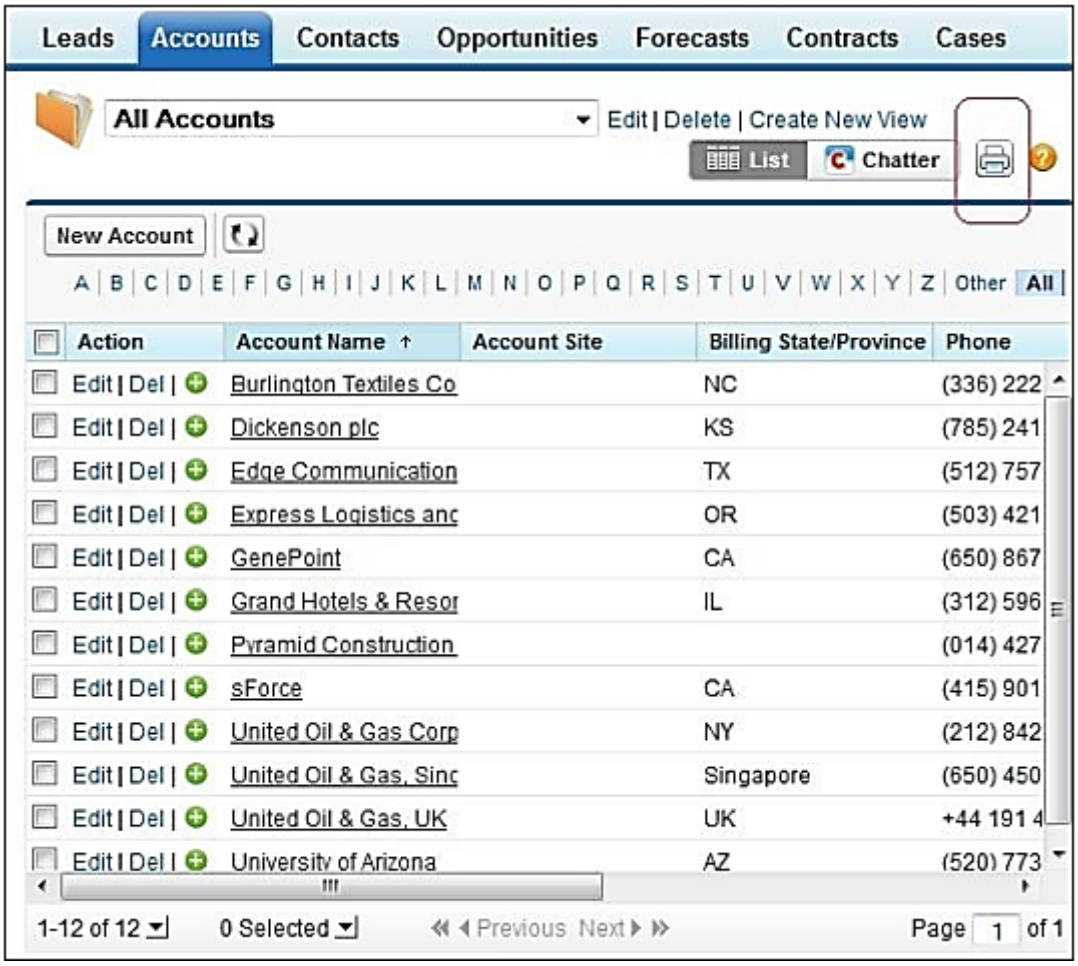


You can click on **New** to create new views.



The following points apply to list views:

- Every object in Salesforce CRM that is associated with a tab automatically has at least one list view. If there is no tab set up for the object, then there will be no corresponding list view.
- List views can be modified by assigning filter criteria to control which records are returned for the affected object.
- List views can be set up to be seen and accessed only by you, or you can set them to be accessed by certain roles and groups of individuals.
- List views have a print feature that can be used by you and your users. To print from a list view, click on the **printable view** button located in the top-right corner of the page, as shown:



User Management & Security in Salesforce CRM

11.0 User Management

11.1 Manage Users

Every Salesforce user is identified by a username, a password, and a single profile. Together with other settings, the profile determines what tasks users can perform, what data they see, and what they can do with the data. To view and manage the users in your org, use the Quick Find box in Setup to find **Users**. The user list shows all the users in your org.

Create a User

You can create users—even multiple users—in just a few clicks. It's as simple as entering a username, alias, and email, and selecting a role, license, and profile. Many more options are available, of course, but that's all you need to get started. Salesforce auto-generates a password and notifies new users immediately. Users can change or add to their own personal information after they log in.

1. Use the Quick Find box to find **Users** | **Users** in Setup.
2. Click **New User**.
3. Or you can click **Add Multiple Users** to add up to ten users at a time.
4. Enter the user's name, email address, and a unique username in the form of an email address. By default, the username is the same as the email address.
5. Select the user license this user will have.

The license determines which profiles are available for each user.

6. Select a profile, which specifies the user's minimum permissions and access settings.
7. Select the option to generate a new password and notify the user, then save.

Deactivate a User

You can't delete a user, but you can deactivate an account so a user can't log in. Deactivated users lose access to all records. (That includes records that are shared with them individually and records shared with them as team members.) However, you can still transfer this data to other users and view the names on the Users page.

1. In Setup, use the Quick Find box to go to **Users**.
2. Click **Edit** next to the name of the user you want to deactivate.
3. Clear the **Active** checkbox and click **Save**.
   If you can't immediately deactivate an account (for example, when the user is selected in a custom hierarchy field), you can freeze their account. That prevents the user from logging in to your organization while you're working on deactivating them.
   a. On the Users page in Setup, click the username of the user whose account you want to freeze.
   b. Click **Freeze**.

12.0 Overview of Data Security

12.1 Introduction

Choosing the data set each user or group of users can see is one of the key decisions that affects the security of your Salesforce org or app. Once you've designed and implemented your data model, give some thought to the kinds of things your users are doing and the data they need to do it. Let's say you're building a recruiting app to help manage open positions, candidates, and job applications. You'll have to store confidential data, such as social security numbers, salary amounts, and applicant reviews that only some types of users should see. You'll want to secure the sensitive data without making life harder for recruiters, hiring managers, and interviewers.

With the Salesforce platform's flexible, layered sharing model, it's easy to assign different data sets to different sets of users. You can balance security and convenience, reduce the risk of stolen or misused data, and still make sure all users can easily get the data they need. The platform makes it easy to specify which users can view, create, edit, or delete any record or field in the app. You can control access to your whole org, a specific object, a specific field, or even an individual record. By combining security controls at different levels, you can provide just the right level of data access to thousands of users without having to specify permissions for each user individually.

12.2 Levels of Data Access

You can control which users have access to which data in your whole org, a specific object, a specific field, or an individual record.

Organization

For your whole org, you can maintain a list of authorized users, set password policies, and limit logins to certain hours and locations.

Objects

Access to object-level data is the simplest thing to control. By setting permissions on a particular type of object, you can prevent a group of users from creating, viewing, editing, or deleting any records of that object. For example, you can use object permissions to ensure that interviewers can view positions and job applications but not edit or delete them.

Fields

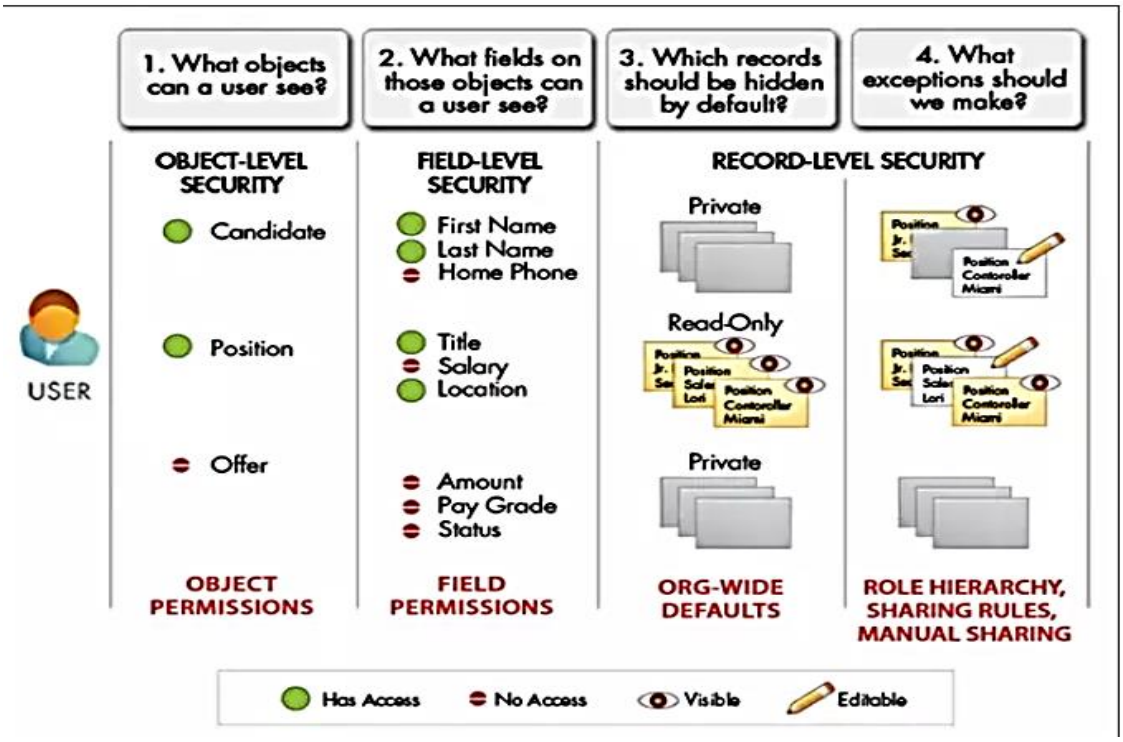
You can restrict access to certain fields, even if a user has access to the object. For example, you can make the salary field in a position object invisible to interviewers but visible to hiring managers and recruiters.

Records

You can allow particular users to view an object, but then restrict the individual object records they're allowed to see. For example, an interviewer can see and edit her own reviews, but not the reviews of other interviewers. You can manage record-level access in these four ways.

- **Organization-wide defaults** specify the default level of access users have to each other's' records. You use org-wide sharing settings to lock down your data to the most restrictive level, and then use the other record-level security and sharing tools to selectively give access to other users.
- **Role hierarchies** give access for users higher in the hierarchy to all records owned by users below them in the hierarchy. Role hierarchies don't have to match your organization chart exactly. Instead, each role in the hierarchy should represent a level of data access that a user or group of users' needs.
- **Sharing rules** are automatic exceptions to organization-wide defaults for particular groups of users, so they can get to records they don't own or can't normally see. Sharing rules, like role hierarchies, are only used to give additional users access to records. They can't be stricter than your organization-wide default settings.
- **Manual sharing** allows owners of particular records to share them with other users. Although manual sharing isn't automated like org-wide sharing settings, role hierarchies, or sharing rules, it can be useful in some situations, such as when a recruiter going on vacation needs to temporarily assign ownership of a job application to someone else.

  Controlling Data Access with the Salesforce Platform



12.3 Control Access to the Organization

When you ensure that only employees who meet certain criteria can log in to Salesforce, you're protecting your data at the broadest level. You do this by managing authorized users, setting password policies, and limiting when and where users can log in.

12.3.1 Set Password Policy

You can configure several settings to ensure that your users' passwords are strong and secure.

Password policies

Set password and login policies, such as specifying an amount of time before all users' passwords expire and the level of complexity required for passwords.

User password expiration

Expire the passwords for all the users in your org, except for users with "Password Never Expires" permission.

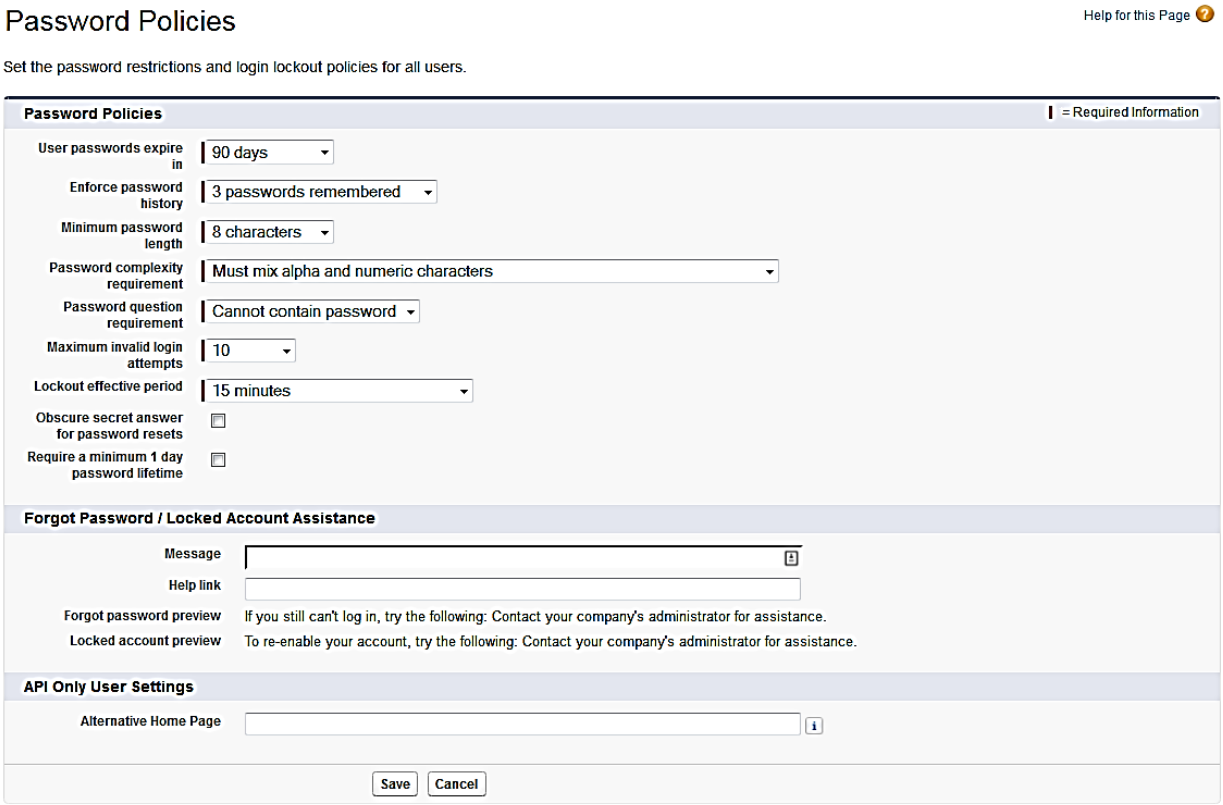
User password resets

Reset the password for specified users.

Login attempts and lockout periods

If a user is locked out due to too many failed login attempts, you can unlock the person's access.

1. Use the Quick Find box to find **Password Policies** in Setup.



2. Customize the password settings.
   a. How long should passwords be?
      Longer is usually better, within reason.
   b. How complex do you want your passwords?
      You can require alphabetical, numeric, uppercase, lowercase, or special characters.
   c. How many days is a password valid?
   d. How many times can someone try to log in with invalid credentials before being locked out?
3. Choose what to do about forgotten passwords and locked accounts.
4. Click **Save**.

12.3.2 Whitelist Trusted IP Ranges for the Org

The first time you log in to Salesforce, the IP address is cached in your browser. Anytime you log in from a different IP address, you will be asked to verify your identity, typically by entering a verification code. You can bypass this step for trusted IP ranges. For example, suppose that your users should be able to log in without entering a verification code whenever they are in the office.

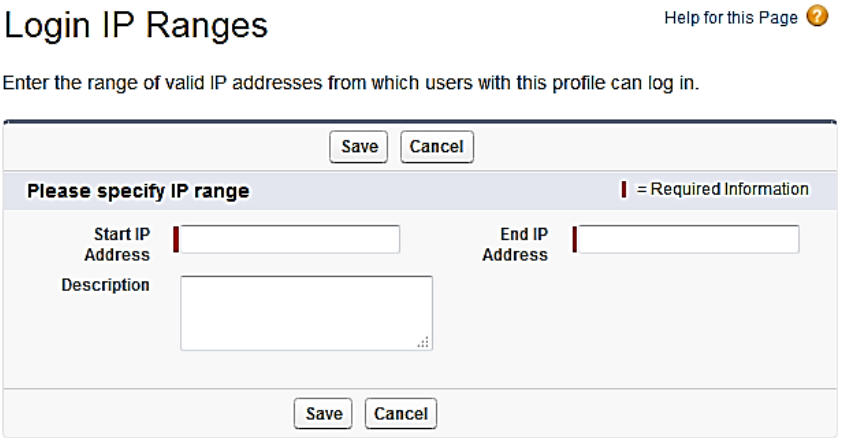
1. From Setup, in the Quick Find box, enter **Network Access**, then select **Network Access**.
2. Click **New**.
3. Enter the start and end point of the range of trusted IP addresses, and click **Save**.

If you have an address outside this range, you aren't excluded from logging in. You simply have to verify your identity by entering a verification code.

12.3.3 Restrict Login Access by IP Address Using Profiles

By default, Salesforce doesn't restrict locations for login access. If you do nothing, users can log in from any IP address. You can restrict where users can log in from using profiles. For example, suppose that certain users shouldn't be able to log in if they're using an IP address outside of the office.

1. From Setup, in the Quick Find box, enter **Profiles**, then select **Profiles**.
2. Select a profile and click its name.
3. Click **IP Ranges**. If you don't have Enhanced Profile Interface enabled, scroll down to the Login IP Range related list.
4. Click **New**.



5. Enter the start and end point of the range of trusted IP addresses, and click **Save**.

Now all users with this profile who are outside the trusted range can't log in. When using profile IP ranges, there are no verification codes to worry about - a user is either in or out.

12.3.4 Restrict Login Access by Time

For each profile, you can specify the hours when users can log in. For example, if you decide your call center employees really only need to look at customer data while they're taking phone calls nine to five, you can make it so they can't log in during evenings and weekends.

1. In Setup, use the Quick Find box to find **Profiles**.
2. Click the profile you want to change.
3. Under **Login Hours**, click **Edit**.
4. Set the days and hours when users with this profile can log in to the organization.
   - To allow users to log in at any time, click **Clear all times**.
   - To prohibit users from using the system on a specific day, set the start and end times to the same value.

12.4 Control Access to Objects

12.4.1 Manage Object Permissions

The simplest way to control data access is to set permissions on a particular type of object. (An object is a collection of records, like leads or contacts.) You can control whether a group of users can create, view, edit, or delete any records of that object.

You can set object permissions with profiles or permission sets. A user can have one profile and many permission sets.

- A user's profile determines the objects they can access and the things they can do with any object record (such as create, read, edit, or delete).
- Permission sets grant *additional* permissions and access settings to a user.

Use profiles to grant the minimum permissions and settings that all users of a particular type need. Then use permission sets to grant more permissions as needed. The combination of profiles and permission sets gives you a great deal of flexibility in specifying object-level access.

Object Permissions for the Recruiting App

As an example, let's explore how you might configure object-level access in the Recruiting app. The app has four main types of users: hiring managers, recruiters, interviewers, and standard employees. What kinds of access to objects does each type of user need?

Hiring Managers

Ben, a hiring manager, should be able to access the recruiting records related to his open positions, but shouldn't have access to other recruiting records (unless they're owned by other hiring managers who report to him). Also, there are certain sensitive fields that he has no need to see, like the social security number field. Let's consider the permissions Ben needs for each of the key custom objects in the app.

- **Position**—Ben should be able to post new positions, as well as update and view all fields for positions for which he's the hiring manager, but he should only be able to view other managers' positions.
- **Candidate**—Ben should only be able to view those candidates who have applied for a position on which he's the hiring manager. Also, since Ben has no reason to see a candidate's social security number, this field should be restricted from his view.
- **Job Application**— Ben needs to be able to update the status of job applications to specify which candidates should be selected or rejected. However, he should not be able to change the candidate listed on the job application, nor the position to which the candidate is applying, so we'll have to find a way of preventing Ben from updating the lookup fields on job applications.
- **Review**—To make a decision about the candidates who are applying, Ben needs to see the reviews posted by the interviewers, as well as make comments on them if he thinks the interviewer was being too biased in his or her review. Likewise, Ben needs to be able to create reviews so that he can remember his own impressions of the candidates he interviews.

Recruiters

Mario, a recruiter, needs to be able to create, view, and modify any position, candidate, job application, or review that's in the system. He also needs to view and modify the recruiting records that all of the other recruiters own, since all the recruiters work together to fill each position, regardless of who created it.

We need to make sure a recruiter will never accidentally delete a record with information about a candidate. That's because state and federal laws require recruitment-related records be saved for a number of years, so that if a hiring decision is questioned, it can be defended in court.

Interviewers

Melissa is an engineer who interviews candidates for highly technical positions. She should be able to view only the candidates and job applications to which she's assigned as an interviewer. Also, she shouldn't be able to view the minimum and maximum salary values for any of the positions or the social security number of any candidate, as that's sensitive information that has nothing to do with her job.

Standard Employees

Employees, such as Harry, are often the best resources for recruiting new hires, even if they are not active hiring managers or interviewers. For this reason, we need to make sure that employees can view open positions, but that they can't see the values for the positions' minimum and maximum salary fields—otherwise they might tip off friends to negotiate for a

position's maximum salary value! Harry also shouldn't be able to view any other records in the Recruiting app.

Here are the required permissions for each of the four types of users.

Custom Object	Recruiter	Hiring Manager	Interviewer	Standard Employee
Position	Read Create Edit	Read Create Edit*	Read (No min/max pay)	Read (No min/max pay)
Candidate	Read Create Edit	Read* (No SSN)	Read * (No SSN)	
Job Application	Read Create Edit	Read Edit (No lookup fields)	Read *	
Review	Read Create Edit	Read Create Edit	Read ** Create Edit **	

* Only for those records that are associated with a position to which the hiring manager or interviewer has been assigned.
** Only for those records that the interviewer owns.
In the rest of this module, you'll learn how you can use the platform to implement these rules in the Recruiting app. As you'll see, this will require configuring security controls at all three levels: objects, fields, and records.

12.4.2 Use Profiles to Restrict Access

Each user has a single profile that controls which data and features that user has access to. A profile is a collection of settings and permissions. Profile settings determine which data the user can see, and permissions determine what the user can do with that data.

- The settings in a user's profile determine whether she can see a particular app, tab, field, or record type.
- The permissions in a user's profile determine whether she can create or edit records of a given type, run reports, and customize the app.

Profiles usually match up with a user's job function (for example, system administrator, recruiter, or hiring manager), but you can have profiles for anything that makes sense for your Salesforce org. A profile can be assigned to many users, but a user can have only one profile at a time.

12.4.3 Standard Profiles

The platform includes a set of standard profiles. Some examples are:

- Read Only
- Standard User
- Marketing User
- Contract Manager
- System Administrator

Each standard profile includes a default set of permissions for all standard objects available on the platform. For example, a Standard User can create and edit records while a Read Only user can view records, but not create or edit them. The System Administrator profile has the widest access to data and the greatest ability to configure and customize Salesforce. The System Administrator profile also includes two special permissions:

- View All Data
- Modify All Data

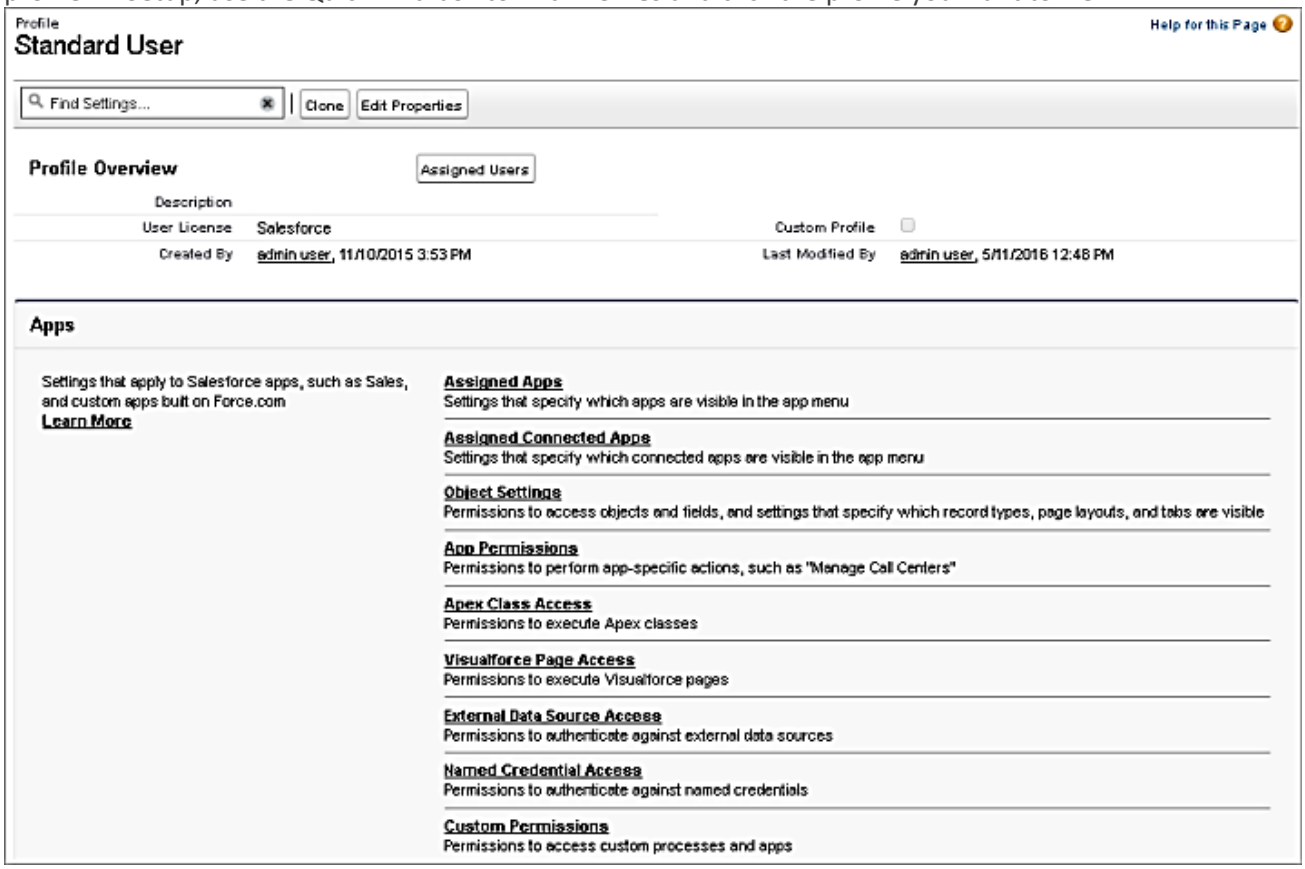
These permissions override all other sharing settings, so use caution when assigning them to any profile other than System Administrator. You can view a list of all standard and custom profiles in Setup.
You can't edit the object permissions on a standard profile. However, you can clone any existing profile, and use that as the basis for a new profile, adjusting the apps and system settings as needed. For example, in the Recruiting app, you might create three new profiles, one each for recruiters, interviewers, and hiring managers. Each profile can then be configured to provide the specific type of data access required for a particular role. You can then use permission sets to grant additional permissions, as required.

Note

The profiles functionality in an org depends on the user license type.

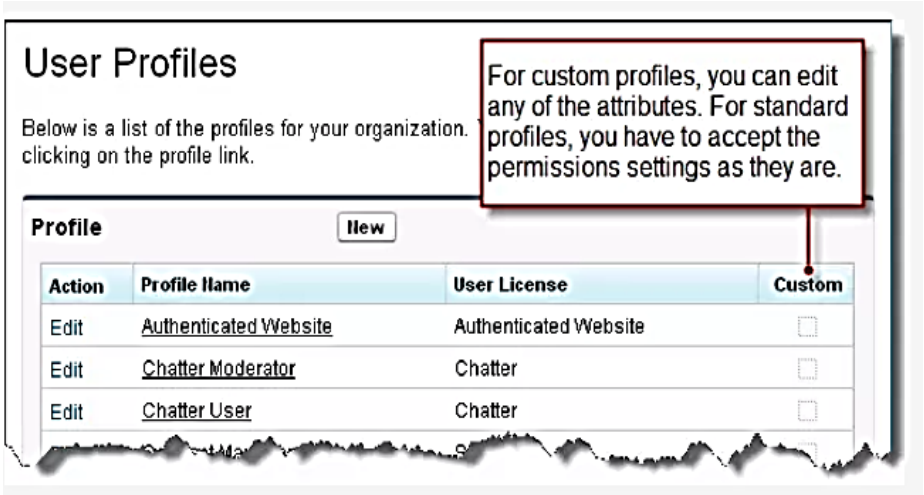
12.4.4 Managing Profiles

The profile overview page provides an entry point for all of the settings and permissions for a single profile. In Setup, use the Quick Find box to find **Profiles** and click the profile you want to view.



12.4.5 Create a Profile

The easiest way to create a profile is to clone an existing profile that's similar to the one you want to create, and then modify it.
Salesforce has an enhanced profile user interface that makes it easy to find and modify profile settings. That's what we'll use for this exercise. To do that, find `User Management Settings` in the **Quick Find** box in Setup, then enable **Enhanced Profile User Interface** and click **Save**.



1. In Setup, use the Quick Find box to find **Profiles**.
2. Click **Clone** next to a profile similar to the one you want to create.
3. Give your new profile a name, and save.

Assign a Profile

Once you've created a profile, you'll want to customize it to match the needs of a set of users, and then assign the profile to those users.

1. Find Profiles in Setup.
2. Select a profile and then click **Object Settings**. Click **Edit** to see its settings.
3. Set the most restrictive settings and permissions you can for this user type, and save. (Don't worry about blocking the user from doing things they need to do. We'll open up more possibilities for them later, when we give them permission sets.)
4. Find Users in Setup and click **Edit** next to one of them.
5. From the **Profile** dropdown select the profile you just set up, and save.

12.4.6 Use Permission Sets to Grant Access

A permission set is a collection of settings and permissions that give users access to various tools and functions. The settings and permissions in permission sets are also found in profiles, but permission sets extend users' functional access without changing their profiles.

Permission sets make it easy to grant access to the various apps and custom objects in your org, and to take away access when it's no longer needed.

Users can have only one profile, but they can have multiple permission sets.

You'll be using permission sets for two general purposes: to grant access to custom objects or apps, and to grant permissions—temporarily or long term—to specific fields.

Grant access to custom objects or apps.

Let's say you have many users in your org with the same fundamental job functions. You can assign them all one profile that grants them all the access they need to do their job. But a few of those users are working on a special project and they need access to an app no one else uses. And a few other users need access to that app, as well as another app that the first group doesn't need. If we only had profiles, you'd have to create more profiles customized to those few users' needs, or take your chances and add more access to the original profile, making the apps available to users that don't need them. Neither of these options is ideal, especially if your org is growing and your users' needs change regularly.

Grant permissions to specific fields.

Let's say you have a user, Tom, who needs temporary edit access to a field while his co-worker is on vacation. You can create a permission set that grants access to the field and

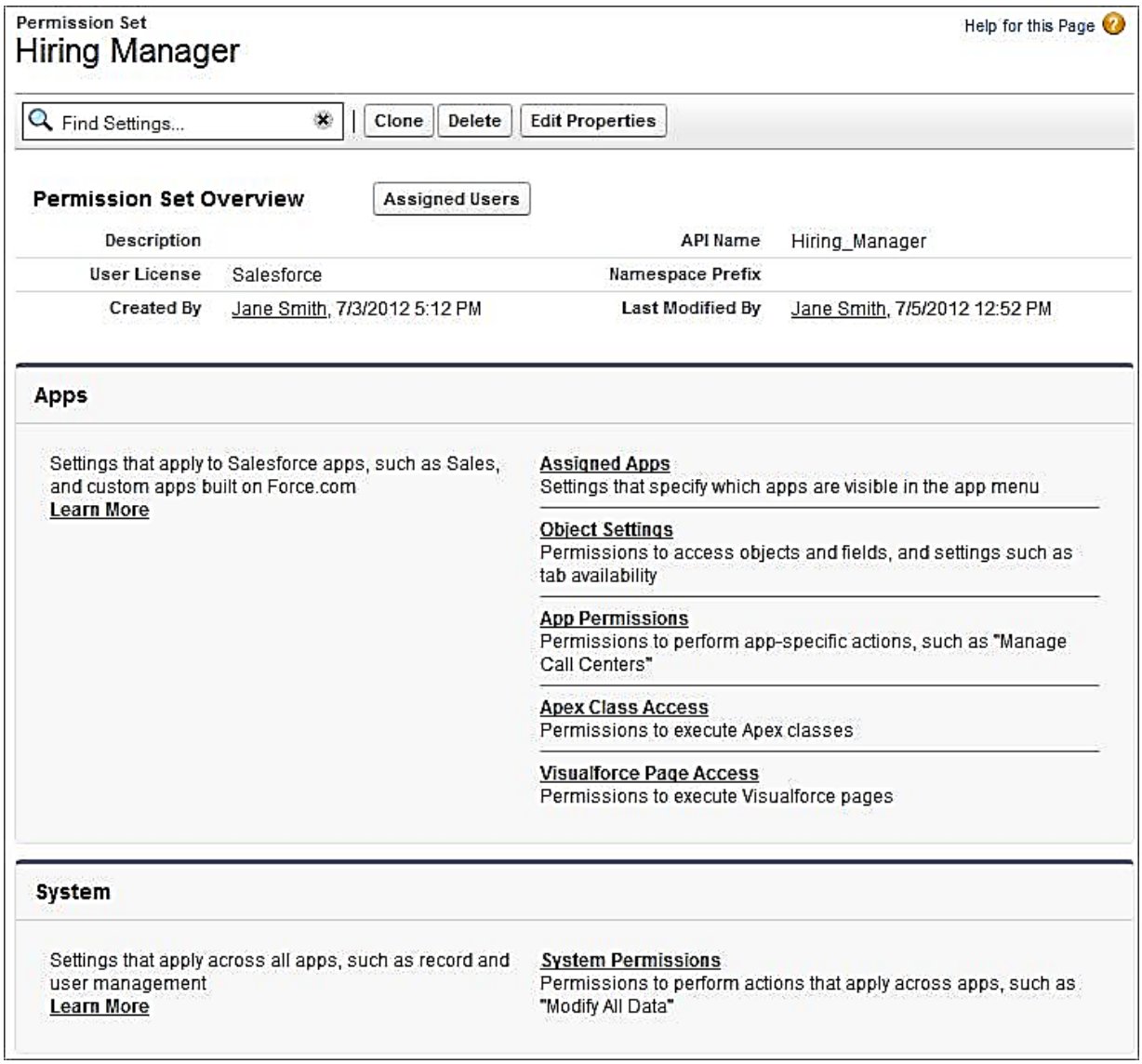
assign the permission set to Tom. When Tom's co-worker returns from vacation and Tom no longer needs access to the field, you just remove the permission set assignment from Tom's user record.

Note

If a user has a permission in their base profile, you can't remove it by assigning a permission set to that user. A permission can only *add* permissions. To take away a permission, you have to remove it from the user's base profile and from any permission sets the user may have.

12.4.7 Managing Permission Sets

A permission set's overview page is the entry point for all of the permissions in a permission set. To open a permission set overview page, find Permission Sets in Setup, then select the permission set you want to view. In each permission set, permissions and settings are organized into app settings, system settings, object permissions, and field permissions.



12.4.8 Create a Permission Set

Create a permission set to grant additional permissions to specific users, on top of their existing profile permissions, without having to modify existing profiles, create new profiles, or grant an administrator profile.

1. Use the Quick Find box to find **Permission Sets** in Setup.
2. Click **Clone** next to the set you want to copy.

Note

A cloned permission set has the same user license as the original. To create a set with a different license, click **New** instead.

3. Enter a label and a description.
   The API name is a unique name used by the API and managed packages. It automatically replicates the label, but you can modify it.
4. If this is a new permission set, select a user license option.
   - If you plan to assign this permission set to multiple users with different licenses, select --None--.
   - If only users with one type of license will use this permission set, select that user license.
5. Click **Save** to go back to the permission set overview page.
6. In the permission set toolbar, click **Manage Assignments**, then click **Add Assignments**.
7. Select the users to assign to this permission set and click **Assign**.
   Review the messages on the Assignment Summary page. If any users weren't assigned, the Message column tells you why.
8. Click **Done** to return to a list of the users assigned to the permission set.

12.4.9 Profiles and Permission Sets for the Recruiting App

Now that you've seen how to create and modify profiles and permission sets, let's set up the appropriate object-level access for our Recruiting app. The app has four main types of users: recruiters, hiring managers, interviewers, and standard employees.

Here are the key considerations for deciding whether to create a profile or permission set for each type of user.

Recruiters

These represent a clearly defined job function, and they need access to different types of data than other users. Hence, it makes sense to create a profile for recruiters.

Hiring Managers

For most orgs, a hiring manager in Sales will need access to a different type of data than a hiring manager in Engineering. However, all hiring managers still need the same types of access to recruiting data—reviews, candidates, positions, and job applications. Hence, it's convenient to create a hiring manager permission set that can be assigned to various types of users.

Interviewers

An employee from any department and in any job function might be called upon to perform an interview and requires access to recruiting information only for a limited amount of time. It makes sense to define a permission set for interviewers, since permissions can be easily assigned and revoked as needed.

Standard Employees

This is a generic group that doesn't reflect a particular job function. For most employees, you can create a base profile that provides access to a small set of data, and then depending on what their specialties are, create and assign permission sets to give them more access as needed.

So from what we've seen, the optimal way to configure object permissions for the Recruiting app is like this:

1. Create two profiles: Recruiters and Standard Employees.
2. Create two permission sets: Hiring Managers and Interviewers.
3. Assign the Standard Employee profile to hiring managers and interviewers, and then grant the appropriate permission set for their function.

12.5 Control Access to Fields

12.5.1 Modify Field-Level Security

Defining field-level security for sensitive fields is the second piece of the security and sharing puzzle, after controlling object-level access.

In some cases, you want users to have access to an object, but limit their access to individual fields in that object. Field-level security settings—or field permissions—control whether a user can see, edit, and delete the value for a particular field on an object. These are the settings that allow us to protect sensitive fields such as a candidate's social security number without having to hide the candidate object.

Unlike page layouts, which only control the visibility of fields on detail and edit pages, field-level security controls the visibility of fields in any part of the app, including related lists, list views, reports, and search results. In fact, to make absolutely sure that a user can't access a particular field, it's important to use the field-level security page for a given object to restrict access to the field. There are simply no other shortcuts that provide the same level of protection for a particular field.

For example, here are some field-level security settings you can set for the Recruiting app.

- Position object—hide minimum and maximum pay from standard employees and interviewers.
- Candidate object—hide social security numbers from hiring managers and interviewers.
- Job Application object—make the Position and Candidate lookup fields read-only for hiring managers.

Field settings can be applied either by modifying profiles or permission sets or from the Field Accessibility menu in Setup.

After setting field-level security for users, you can:

- Create page layouts to organize the fields on detail and edit pages.
- Verify users' access to fields by checking the field accessibility.
- Customize search layouts to set the fields that display in search results, in lookup dialog search results, and in the key lists on tab home pages.

12.5.2 Restrict Field Access with a Profile

You apply field settings by modifying profiles or permission sets. Let's try *restricting* a user's general access with a profile. Then we'll *expand* it as needed with a permission set.

Tip: If you haven't already done it, before you start, enable the enhanced profile user interface. Type `User Management Settings` in the **Quick Find** box in Setup, then enable **Enhanced Profile User Interface** and click **Save**.

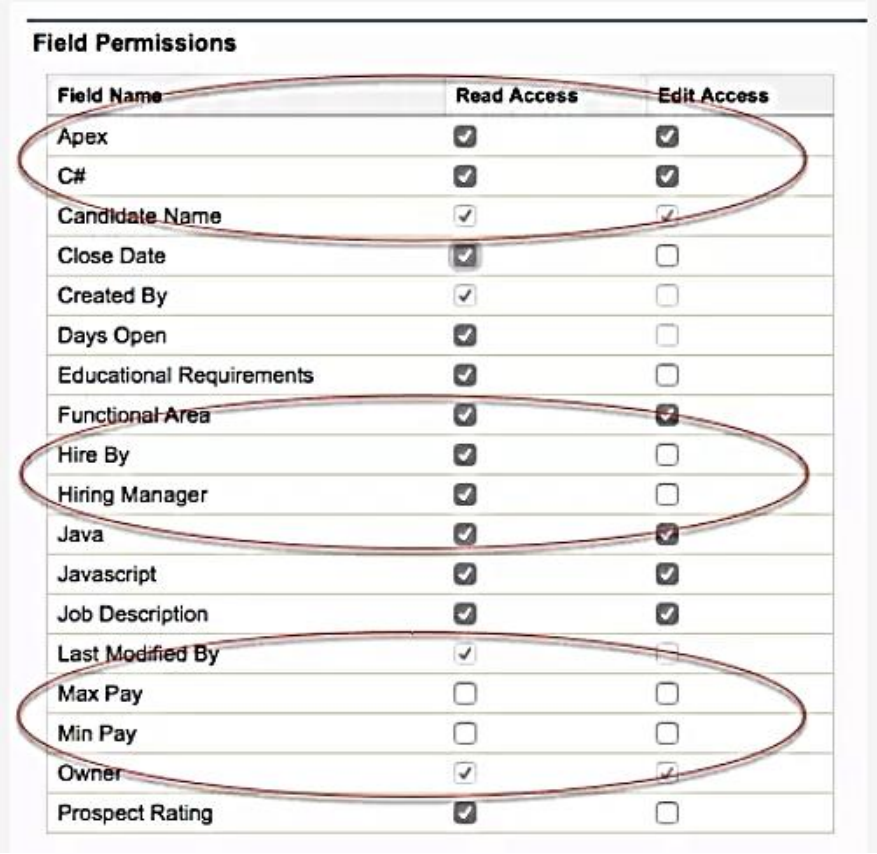
1. Use the Quick Find box to find **Profiles** in Setup.
2. Select the profile you want to change. "Standard User" will do nicely.
3. Click **Object Settings** and select the object for which you want to update the field settings.
4. Click **Edit**.
5. For each field, specify the kind of access you want for users with this profile, and save your settings.

Now that you've set field-level security for sensitive data, you can create page layouts to organize the fields for users' convenience, and customize how the fields display in search results and lists. For the final piece of the puzzle, specify the individual records to which each user needs access. By combining security controls at all three levels, you can set up a highly secure data access model which is flexible enough to meet the needs of many different types of users.

Add Field Access with a Permission Set

Let's look at how field settings can be applied by modifying permission sets. Remember, a permission set is for *expanding* a user's access to fields that are restricted in their profile.
Let's set up our interviewers to update the candidate record when they've interviewed a candidate. We'll assume our interviewers have the Standard User profile.
We worked with Permission Sets when we set up our custom objects. Now we'll go back to that Setup page to make sure the right fields in one or our objects are available to the users who need them.

1. In Setup, use the Quick Find box to find **Permission Sets**.
2. Select a permission set and click **Object Settings**.
3. Click the object you're working with, then click **Edit**.
   In this example, we're modifying the Candidate object.
4. Under **Field Permissions**, specify the kinds of access your interviewer's need, and then save this permission set.



Field Permissions

Field Name	Read Access	Edit Access
Apex	☑	☑
C#	☑	☑
Candidate Name	☑	☑
Close Date	☑	☐
Created By	☑	☐
Days Open	☑	☐
Educational Requirements	☑	☐
Functional Area	☑	☑
Hire By	☑	☐
Hiring Manager	☑	☐
Java	☑	☑
Javascript	☑	☑
Job Description	☑	☑
Last Modified By	☑	☐
Max Pay	☐	☐
Min Pay	☐	☐
Owner	☑	☑
Prospect Rating	☑	☐

See how we've enabled our interviewers to both read and change the values of the Apex and C# checkboxes? Now they can check or uncheck those boxes when they've determined the candidate's command of those skills. We've prevented them from changing the Hire By date or the name of the hiring manager, but they can see that information. And they don't need to know the pay rate for the position, so we've removed both their Read and Edit access for those fields.

5. Click **Manage Assignments** and select the users who you expect to need the permissions you've just specified. Click **Add Assignments** and **Done**, and you're done!

Now you've defined field-level security for sensitive data. For the final piece of the puzzle, specify the individual records each user needs access to. By combining security controls at all three levels, you can set up a highly secure data access model that's flexible enough to meet the needs of many different types of users.

12.6 Control Access to Records

12.6.1 Record-Level Security

To control data access precisely, you can allow particular users to view specific fields in a specific object, but then restrict the individual records they're allowed to see.
Record access determines which individual records users can view and edit in each object they have access to in their profile. First ask yourself these questions:

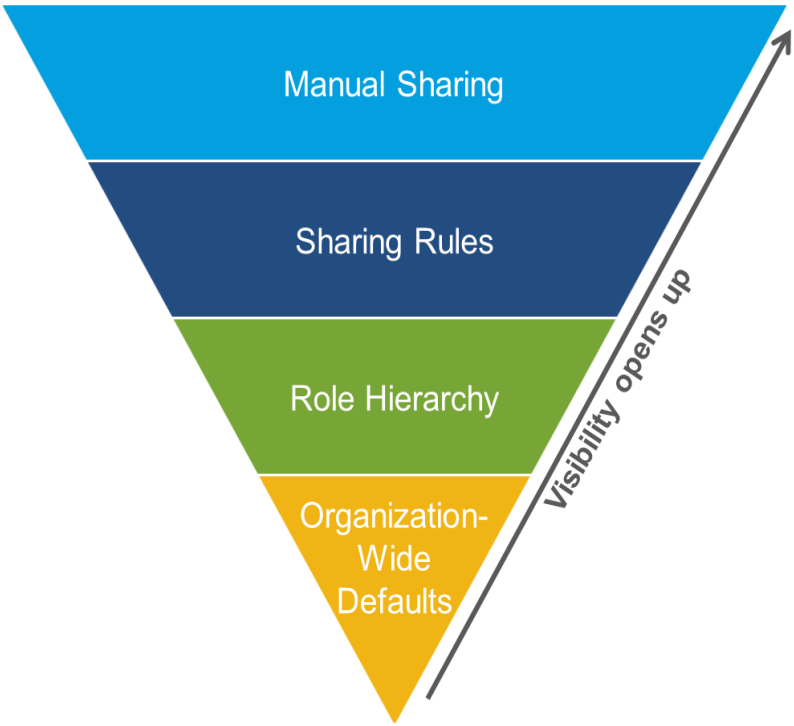
- Should your users have open access to every record, or just a subset?
- If it's a subset, what rules should determine whether the user can access them?

Let's say you create a new profile called Recruiter to give recruiters the object-level permissions they need. You restrict the power to delete recruiting-related objects, so recruiters will never be able to delete these objects. However, granting recruiters permission to create, read, or edit recruiting objects does not necessarily mean recruiters can read or edit every record in the recruiting object. This is a consequence of two important concepts:

- The permissions on a record are always evaluated according to a combination of object-level, field-level, and record-level permissions.
- When object-level permissions conflict with record-level permissions, the most restrictive settings win.

That means even if you grant a profile create, read, and edit permissions on the recruiting objects, if the record-level permissions for an individual recruiting record are more restrictive, those are the rules that define what a recruiter can access.
You control record-level access in four ways. They're listed in order of increasing access. You use org-wide defaults to lock down your data to the most restrictive level, and then use the other record-level security tools to grant access to selected users, as required.

- **Org-wide defaults** specify the default level of access users have to each other's records.
- **Role hierarchies** ensure managers have access to the same records as their subordinates. Each role in the hierarchy represents a level of data access that a user or group of users needs.
- **Sharing rules** are automatic exceptions to org-wide defaults for particular groups of users, to give them access to records they don't own or can't normally see.
- **Manual sharing** lets record owners give read and edit permissions to users who might not have access to the record any other way.



The visibility and access for any type of data is determined by the interaction of the above security controls, based on these key principles.

- A user's baseline permissions on any object are determined by their profile.
- If the user has any permission sets assigned, these also set the baseline permissions in conjunction with the profile.
- Access to records a user does not own are set first by the org-wide defaults.
- If the org-wide defaults are anything less than **Public Read/Write**, you can open access back up for certain roles using the role hierarchy.
- You can use sharing rules to expand access to additional groups of users.
- Each record owner can manually share individual records with other users by using the Share button on the record.

You've already seen how to configure object-level and field-level access using profiles and permission sets. Now we'll look at details of the various record-level security controls.

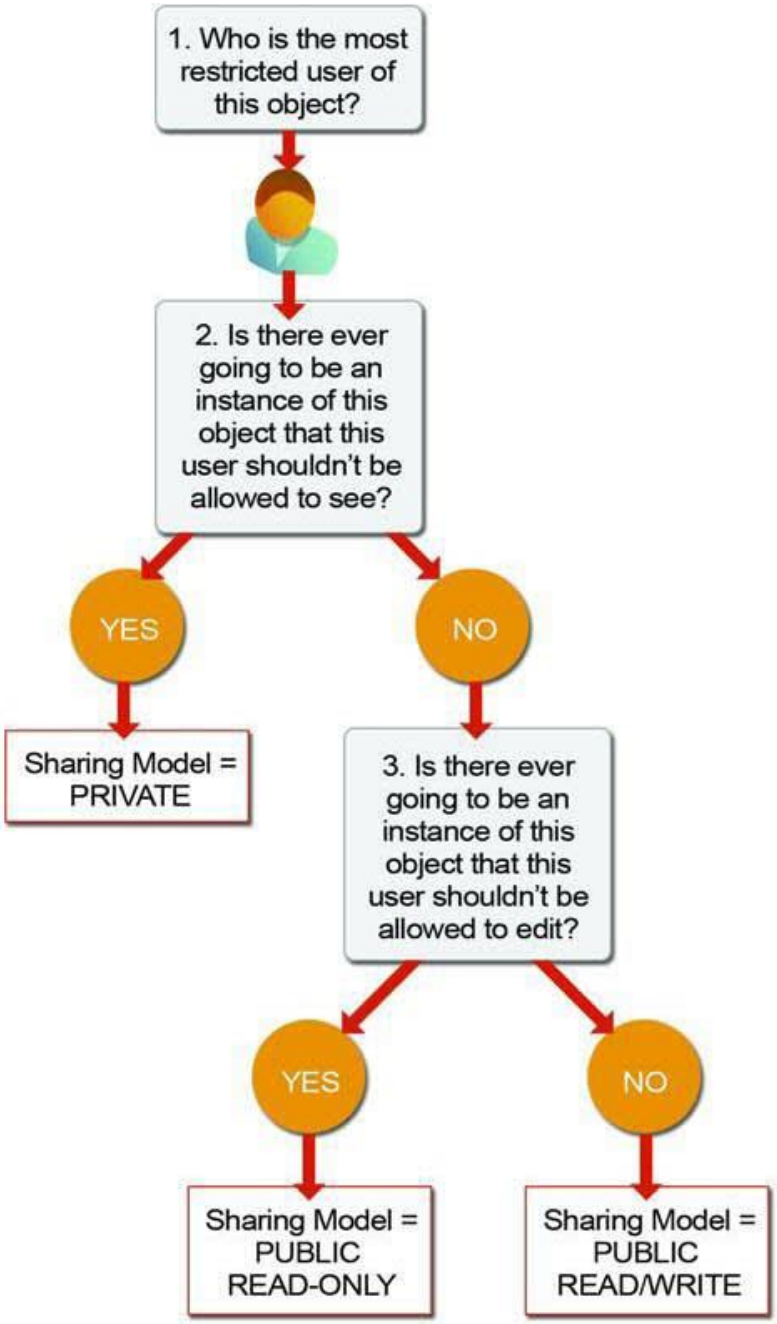
12.6.2 Org-Wide Sharing

Org-wide defaults specify the baseline level of access that the most restricted user should have. Use org-wide defaults to lock down your data, and then use the other record-level security and sharing tools (role hierarchies, sharing rules, and manual sharing) to open up the data to users who need it. Object permissions determine the baseline level of access for all the records in an object. Org-wide defaults modify those permissions for records a user doesn't own. Org-wide sharing settings can be set separately for each type of object.

Org-wide defaults can never grant users more access than they have through their object permission.

To determine the org-wide defaults you need for your app, ask yourself these questions about each object:

1. Who is the most restricted user of this object?
2. Is there ever going to be an instance of this object that this user shouldn't be allowed to see?
3. Is there ever going to be an instance of this object that this user shouldn't be allowed to edit?



Based on your answers, you can set the sharing model for that object to one of these settings.

Private

Only the record owner, and users above that role in the hierarchy, can view, edit, and report on those records.

Public Read Only

All users can view and report on records, but only the owner, and users above that role in the hierarchy, can edit them.

Public Read/Write

All users can view, edit, and report on all records.

Controlled by Parent

A user can view, edit, or delete a record if she can perform that same action on the record it belongs to.

When the org-wide sharing setting for an object is **Private** or **Public Read Only**, an admin can grant users additional access to records by setting up a role hierarchy or defining sharing rules. Sharing rules can only be used to grant additional access. They cannot be used to restrict access to records beyond what was originally specified with the org-wide sharing defaults.

As an example, let's go through and answer the above list of questions for the Position object in the Recruiting app.

Who is the most restricted user of this object?

A member of the Standard Employee profile. All that they're allowed to do is view a position.

Is there ever going to be an instance of this object that this user shouldn't be allowed to see?

No. Although the values for the minimum and maximum pay fields are hidden from standard employees, they're still allowed to view all position records.

Is there ever going to be an instance of this object that this user shouldn't be allowed to edit?

Yes. Standard employees aren't allowed to edit any position record.

Since we answered "Yes" to the third question, the sharing model for the Position object should be set to Public Read Only. By repeating the same exercise with the other recruiting objects, you can easily figure out the appropriate org-wide default settings for them. The Standard Employee profile is the most restricted user for each object, and there are going to be candidate, job application, and review records that particular employees won't be able to view. Consequently, the sharing model for the Candidate, Job Application, and Review objects should all be set to **Private**.

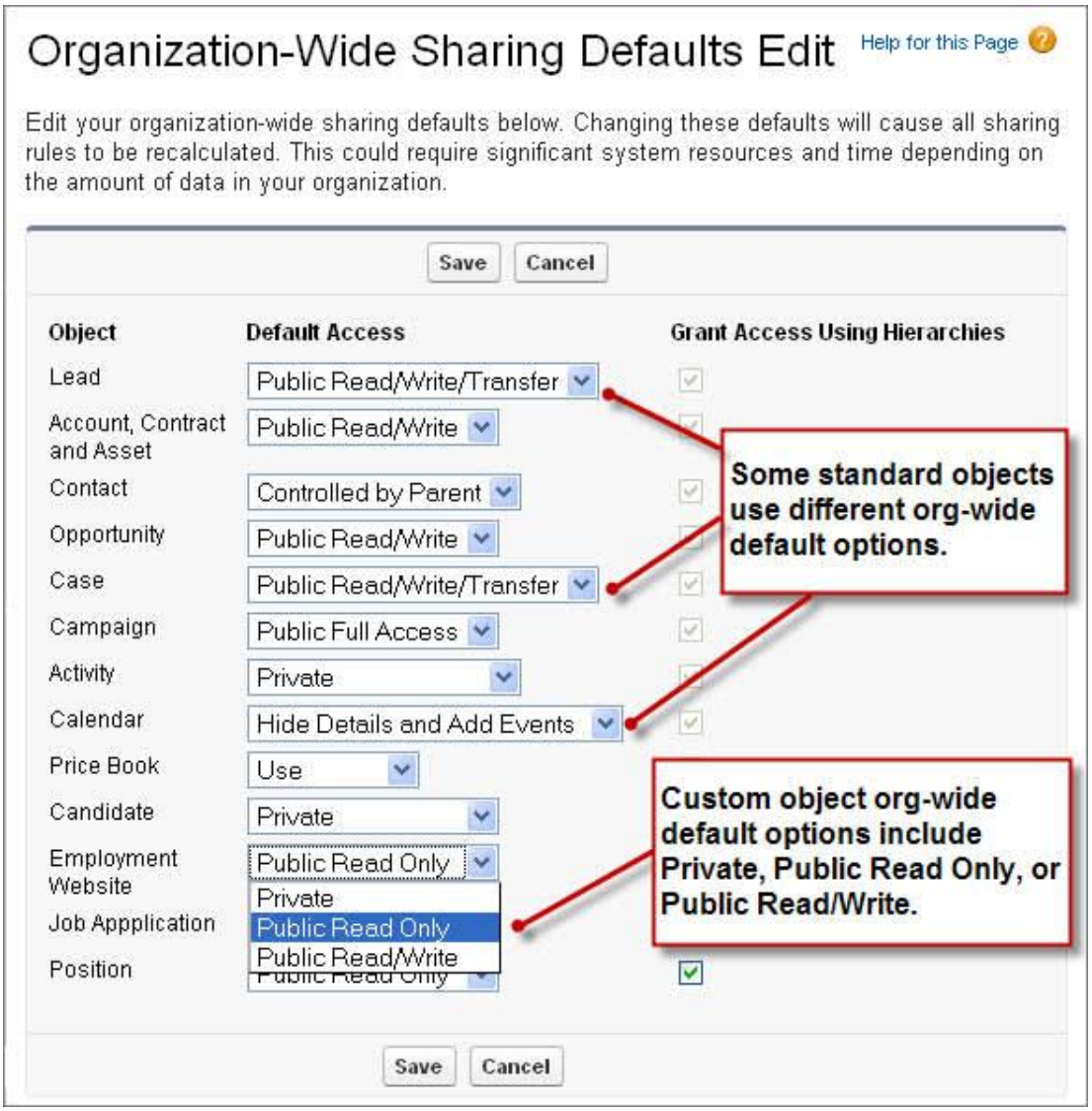
Note

You can't set the org-wide defaults for the Review object, because that object is on the detail side of a master-detail relationship, and a detail record automatically inherits the sharing setting of its parent. So, in our app, the Review object is automatically set to **Private.**

Set Your Org-Wide Sharing Defaults

Use org-wide defaults to specify the baseline level of access that the most restricted user should have.

1. In Setup, use the Quick Find box to find **Sharing Settings**.
2. Click **Edit** in the Organization-Wide Defaults area.



3. For each object, select the default access you want to give everyone.
4. To disable automatic access using your hierarchies, deselect **Grant Access Using Hierarchies** for any custom object that does not have a default access of Controlled by Parent.

By default, a role hierarchy automatically grants access to records for users above the record owner in the hierarchy. Setting an object to **Private** makes those records visible *only* to record owners and those above them in the role hierarchy. Use the **Grant Access Using Hierarchies** checkbox to disable access to records to users above the record owner in the hierarchy for custom objects. If you deselect this checkbox for a custom object, only the record owner and users granted access by the org-wide defaults receive access to the records.

Even if **Grant Access Using Hierarchies** is deselected, some users—such as those with the "View All" and "Modify All" object permissions and the "View All Data" and "Modify All Data" system permissions—can still access records they don't own.

Note

Updating the org-wide defaults automatically runs sharing recalculation to apply any access changes to your records. You receive a notification email when the recalculation completes and you can refresh the Sharing Settings page to see your changes. To view the update status, from Setup, enter `View Setup Audit Trail` in the Quick Find box, then select **View Setup Audit Trail**.

Once you've locked down your data with org-wide defaults, the resulting settings might be too restrictive for some users. You can then use the remaining record-level security controls (role hierarchies, sharing rules, and manual sharing) to open up record access selectively to specific employees who need it.

12.6.3 Create a Role Hierarchy

Create and Edit Roles

A role hierarchy works together with sharing settings to determine the levels of access users have to your Salesforce data. Users can access the data of all the users directly below them in the hierarchy. Users who need to see a lot of data (such as the CEO, executives, or other management) often appear near the top of the hierarchy. But role hierarchies don't have to match your org chart. Each role in the hierarchy just represents a level of data access that a user or group of users needs.

- A manager always has access to the same data as his or her employees, regardless of the org-wide default settings.
- Users who tend to need access to the same types of records can be grouped together. We'll use these groups later when we talk about sharing rules.

Depending on your sharing settings, roles can control the level of visibility that users have into your Salesforce data. Users at any given role level can view, edit, and report on all data owned by or shared with users below them in the role hierarchy, unless your sharing model for an object specifies otherwise. Specifically, in the Organization-Wide Defaults related list, if the **Grant Access Using Hierarchies** option is disabled for a custom object, only the record owner and users granted access by the org-wide defaults receive access to the object's records.

Beyond setting the org-wide sharing defaults for each object, you can specify whether users have access to the data owned by or shared with their subordinates in the hierarchy. For example, the role hierarchy automatically grants record access to users above the record owner in the hierarchy. By default, the **Grant Access Using Hierarchies** option is enabled for all objects. It can only be changed for custom objects.

To control sharing access using hierarchies for any custom object, enter Sharing Settings in the Quick Find box and select **Sharing Settings**. In the Organization Wide Defaults section, click **Edit**.

Deselect **Grant Access Using Hierarchies** if you want to prevent users from gaining automatic access to data owned by or shared with their subordinates in the hierarchies.

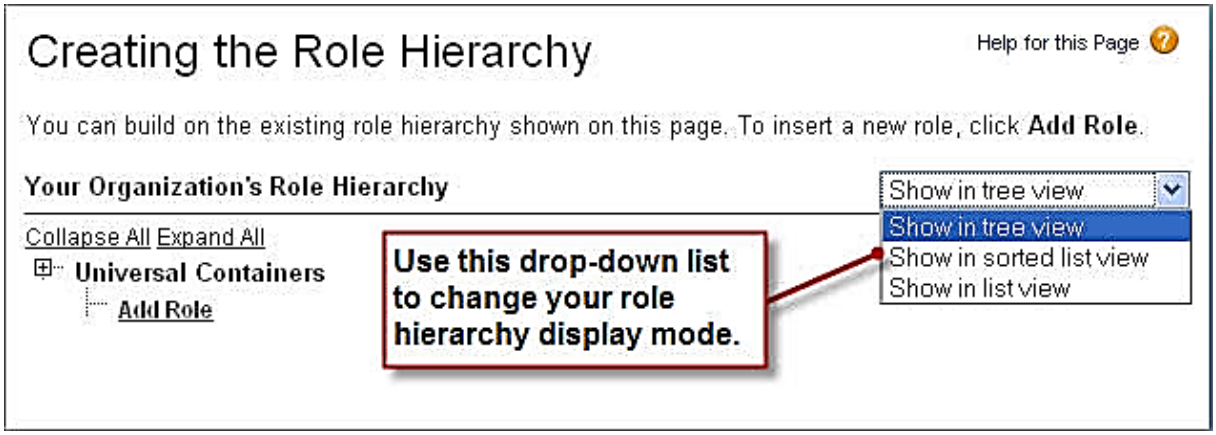
Define a Role Hierarchy

Implementing a role hierarchy in the platform is easy once you have an idea of what the hierarchy should look like. It's best to start with your company's org chart and then consolidate different job titles into single roles wherever possible.

For example, if a software development group has a staff software engineer and a junior software engineer, these positions can be consolidated into a single Software Engineer role in the hierarchy. Once that's done, you can get started defining the role hierarchy itself.

1. From Setup, use the Quick Find box to find **Roles**.

Tip: If you see an introductory splash page called Understanding Roles, click **Set Up Roles** at the bottom of the page to skip to the actual tool.



The default view for this page is the tree view, as indicated in the drop-down list on the far right side of the Role Hierarchy title bar. When creating a role hierarchy, it's probably easiest to stick with this or the list view, because they both make it easy to see how the roles all fit together in the hierarchy. The sorted list view is best if you know the name of a role that you want to find but aren't sure where it fits in the hierarchy, or if you don't want to click open all the tree nodes. For our purposes, we'll stick with the tree view.
When you first start defining a role hierarchy, the tree view displays a single placeholder node with the name of your org. From this point, we need to add the name of the role that is highest up in the hierarchy—in our case, the CEO.

Note

If you're building your app with a free Developer Edition or Trailhead Playground org, you may have a role hierarchy predefined as a sample. That's all right. You can still follow along and create some more roles.

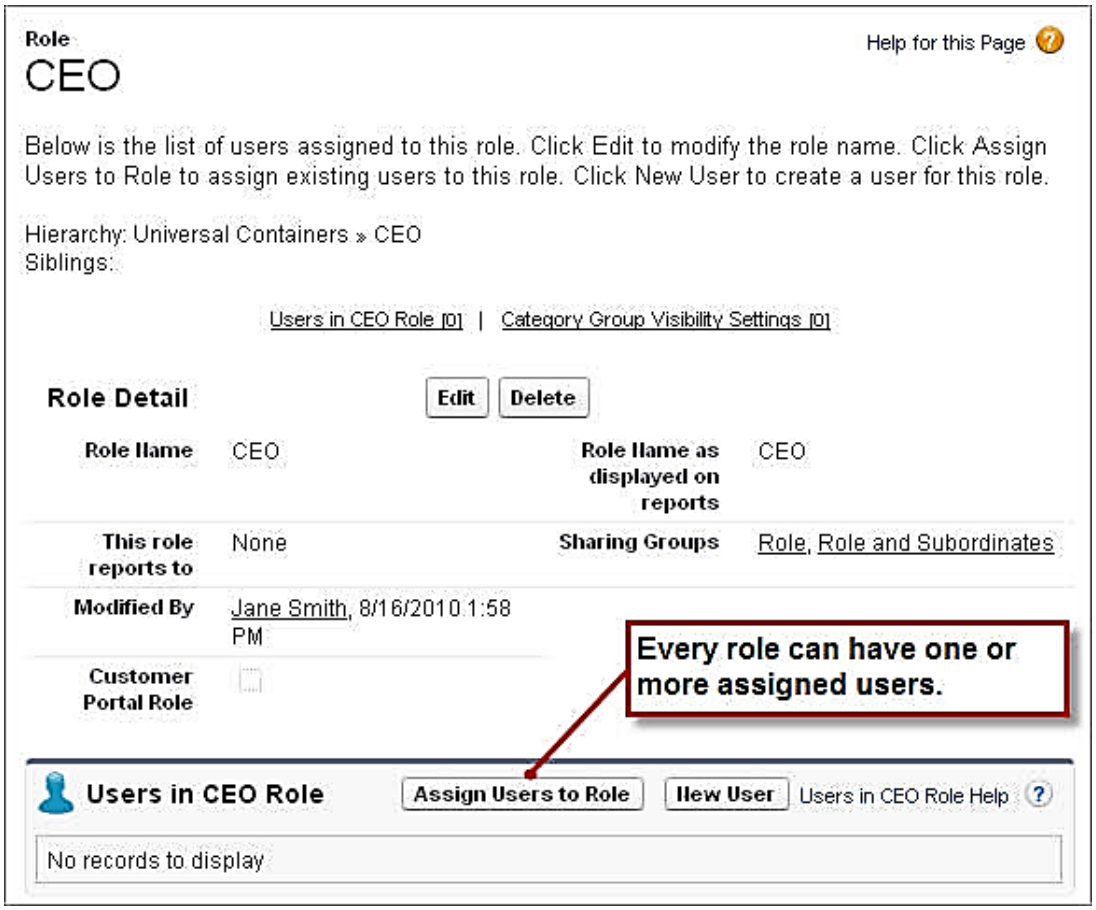
2. Just under the company name, click **Add Role**.

Note

If the CEO role already exists, click **Edit**.

3. In the **Label** text box, enter CEO.
The **Role Name** text box auto populates with CEO.

4. In the **This role reports to** text box, click the lookup icon and click **Select** next to the name of your org.
   By choosing the name of the org in the **This role reports to** text box, we're indicating that the CEO role is a top-level position in our role hierarchy and doesn't report to anyone.
5. In the **Role Name as displayed on reports** text box, enter CEO.
   This text is used in reports to indicate the name of a role. Since a long role name, like Vice President of Product Development, takes extra space in your report columns, we recommend using a short but readable abbreviation.
6. Leave any other options, such as **Opportunity Access**, set to their defaults, and save.
   These access options don't have anything to do with our Recruiting app, and only appear if you have the org-wide defaults for a standard object set to a level more restrictive than **Public Read/Write**.
7. Now that you've created your first role, you can assign the appropriate user to it. Click **CEO**, and on the CEO role detail page, click **Assign Users to Role**.



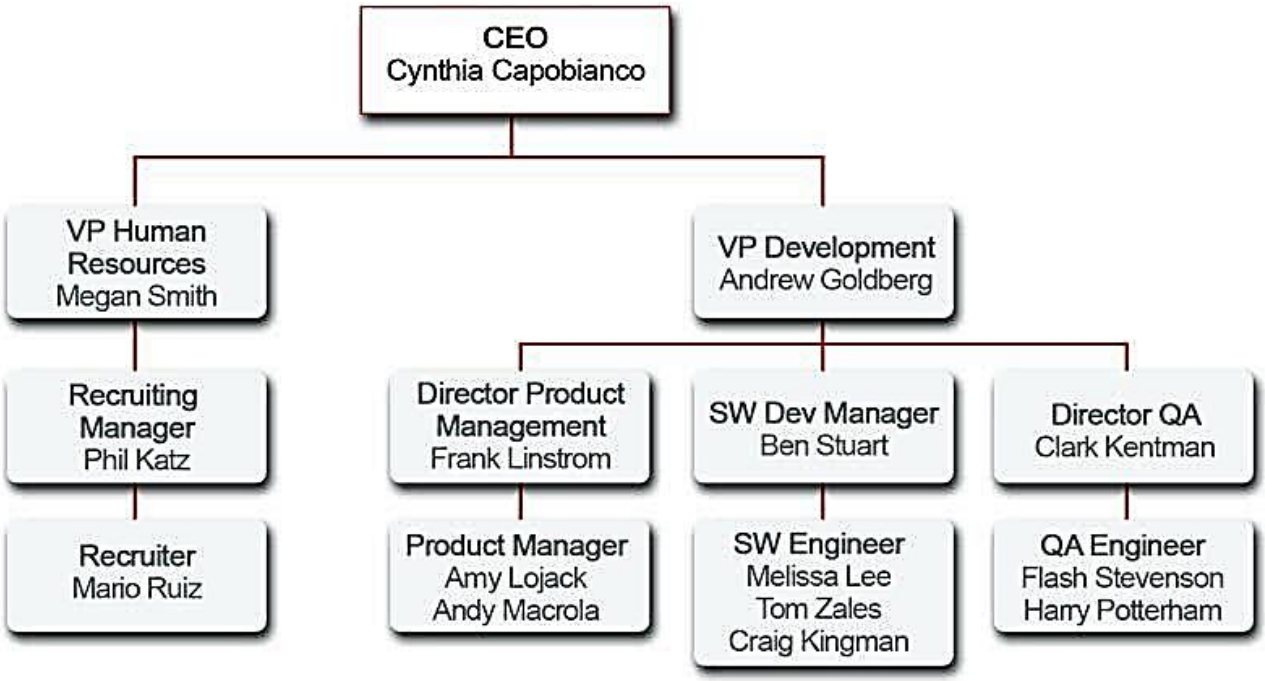
8. In the **Available Users** drop-down list, select All Unassigned.
9. Choose a user from the list, and click **Add** to move her to the **Selected Users for CEO** list, then save.

If you return to the main Roles page from Setup, you can now see the new CEO role in the hierarchy. You can define the rest of the roles according to your role hierarchy diagram. There's no need to assign users to every role right away—you can do that later as you create the rest of your users and test out your app.

Tip
To speed up the process of adding a new role, click **Add Role** directly under the name of the role to which the new role should report. When you do this, the **This role reports to** text box is automatically filled in with the name of the appropriate role.

Role Hierarchy for the Recruiting App

Let's take a look at a branch of the role hierarchy for a fictional company that's using our Recruiting App. Remember, with the org-wide defaults we defined, hiring managers are allowed to view (but not create or update) all position, job posting, and employment website records, and to view and update other recruiting records they own. That doesn't make our app all that useful. However, once our role hierarchy is in place, our users can get to the data they need, and our app is off and running.



This role hierarchy automatically grants these kinds of record-level permissions:

- The CEO, Cynthia, can view and update every record that anyone else in the organization can view and update.
- The VP of Development, Andrew, can view and update any record that his managers or his managers' employees can view or update.
- The VP of Human Resources, Megan, can view and update any record that Phil, her recruiting manager, or Mario, Phil's recruiter, can view and update.
- The Recruiting Manager, Phil, can view and update any record that is owned by Mario, his recruiter.
- The Software Development manager, Ben, can view and update any record that is owned by Melissa, Tom, or Craig, his software engineers.
- The director of QA, Clark, can view and update any record that is owned by Flash or Harry, his QA engineers.

As you can see, the role hierarchy is a powerful way to open up data for people who need to see a lot of it! With org-wide defaults and a role hierarchy in place, you're almost done with the record-level access permissions for the Recruiting app. All that's left is to share recruiting-related records between groups that appear in separate branches of the role hierarchy, and between peers in a single group. You can do both those things with a combination of sharing rules and manual sharing.

12.6.4 Define Sharing Rules

Extend Access with Sharing Rules

Your org-wide default sharing settings give you a (relatively restrictive) baseline level of access for each object. If you have org-wide sharing defaults of Public Read Only or Private, you can open access back up for some users with sharing rules. This enables you to make automatic exceptions to your org-wide sharing settings for selected sets of users.

Sharing rules can be based on who owns the record or on the values of fields in the record. For example, use sharing rules to extend sharing access to users in public groups or roles. As with role hierarchies, sharing rules can never be stricter than your org-wide default settings. They just allow greater access for particular users.

Each sharing rule has three components.

Share which records?

You can share records owned by certain users or meeting certain criteria. Criteria-based sharing rules determine what records to share based on field values other than ownership.

With which users?

You can define groups of users by role, by territory, or by defining a public group. A public group is an admin-defined grouping of users that can be used to simplify the creation of sharing rules. Each public group can be a combination of:

- individual users
- roles
- roles and subordinates
- territories
- territories and subordinates
- other public groups
-

What kind of access?

You can assign either Read-Only or Read/Write access.

Sharing rules work best when they're defined for a particular group of users that you can determine or predict in advance, rather than a set of users that frequently changes. For example, in the Recruiting app, it’s important to share every position, candidate, job application, and review with every recruiter. Since recruiters all belong to either the Recruiting Manager or Recruiter roles in the role hierarchy, we can easily use a sharing rule to share those objects with the Recruiting Manager role and its subordinates.

Alternatively, consider another use case from the Recruiting app: interviewers need read access on the candidates and job applications for people they're interviewing. In this case, the set of interviewers is a lot harder to predict in advance—hiring managers might use different sets of interviewers depending on the position for which they're hiring, and the interviewers might come from different groups in the role hierarchy. So, this use case probably shouldn't be handled with sharing rules—the team of interviewers for any given manager is just too hard to predict.

Should we use a sharing rule?

Let's go through the set of required permissions we still want to implement and pick out the ones that work best with sharing rules.

Recruiters need read and update access on every position, candidate, job application, and review record that exists in the app.

Yes. As we discussed previously, it's easy to pick out the group of recruiters in our role hierarchy.

Hiring managers need read and update access on position and job posting records on which they're the hiring manager.

No. It's too hard to predict which positions will be assigned to which hiring manager. We'll need to handle this use case some other way.

Hiring managers need read access on candidate records on which they're the hiring manager.
No. Again, it's too hard to predict which positions will be assigned to which hiring manager.

Hiring managers need read and update access on every job application and review record.
Yes. Since we're not restricting which job applications and reviews a hiring manager gets to read and update, we can easily pick out all of the hiring managers from our role hierarchy and define a sharing rule for them.

Interviewers need read access on the candidate and job application records for people they're interviewing.
No. As we discussed previously, it's hard to predict who will be a member of an interview team for a particular position.
To summarize, here are the key sharing rules we define for the Recruiting app.

Object	Rule Label	Owned by...	Share with...	Access Level
Review	Edit Reviews	Entire Organization	Recruiters and Hiring Managers	Read/Write
Candidate	Edit Candidates	Entire Organization	The role and subordinates of the Recruiting Manager	Read/Write
Position	Edit Positions	The role and subordinates of the Recruiting Manager	The role and subordinates of the Recruiting Manager	Read/Write

12.6.5 Define a Public Group

Before creating a sharing rule, it’s important to set up the appropriate public group. A public group is a collection of individual users, other groups, individual roles or territories, and/or roles or territories with their subordinates that all have a function in common. For example, users with the Recruiter profile as well as users in the SW Dev Manager role both review job applications.
Using a public group when defining a sharing rule makes the rule easier to create and, more important, easier to understand later, especially if it's one of many sharing rules that you're trying to maintain in a large organization. Create a public group if you want to define a sharing rule that encompasses more than one or two groups or roles, or any individual.
Looking at the required permissions that we want to implement, there are just two objects that need a public group for their sharing rules: Job Application and Review. The good news is that we can cover these objects in a single group because the Review object is on the detail side of a master-detail relationship, so it inherits the sharing settings we apply to the Job Application object. Since both recruiters and hiring managers need read and update access to job applications and reviews, we can define a public group called Reviewers that includes both recruiters and hiring managers.

1. In Setup, use the Quick Find box to find **Public Groups**.
2. Click **New**.
   The New Public Group page allows you to choose other public groups, individual roles, individual roles including the roles’ subordinates, or individual users.
3. Give your rule the label Reviewers.

The **Group Name** text box populates automatically when you click it. This is the unique name used by the API and managed packages.

4. In the **Search** drop-down list, choose Roles.
5. In the **Available Members** list, select SW Dev Manager, Director Product Management, and Director QA, then click **Add**.
6. Go back up to the **Search** drop-down list, and this time choose Role and Subordinates.
7. In the **Available Members** list, select Recruiting Manager, click **Add**, and save.

Once you've defined your group, you can use it to define sharing rules.

Define a Sharing Rule

Let's use the public group we've created to define a sharing rule for review records. You can define a sharing rule for a single public group, role, or territory. You can also define a sharing rule for a role plus its subordinates or for a territory plus its subordinates.

There is already one default public group that encompasses every user in your organization.

1. In Setup, use the Quick Find box to find **Sharing Settings**. This is the same page used to define org-wide defaults.
2. In the **Manage sharing settings for** drop-down list, choose Job Application.
   Choosing an object in this drop-down list allows you to focus in on the org-wide defaults and sharing rules for a single object at a time rather than looking at all of them in a long page—a useful thing if you've got a large org with multiple custom objects. Let's use this Sharing Rules related list to create a sharing rule that applies to both the Job Application and the Review objects.
3. In the Job Application Sharing Rules area, click **New** and give your rule the label Review Records.
   The **Rule Name** text box populates automatically when you click it.
4. For the rule type, make sure **Based on record owner** is selected.
5. For **Select which records to be shared**, select Public Groups, then choose Entire Organization.
6. For **Select users to share with**, select Public Groups, then choose Reviewers.
7. For **Select the level of access for the users**, select Read/Write, then save.

We just created a rule that shares reviews written and owned by any member of the org with all recruiters and hiring managers. Since reviewers and hiring managers all need to read and update reviews, we handled everyone with a single sharing rule and a public group.

13.0 Implementing Business: Processes in Salesforce CRM

This chapter will focus on how you can configure actions for workflow rules and approval processes to automate, improve quality, and generate high-value processes within your organization. The topics covered are as follows:

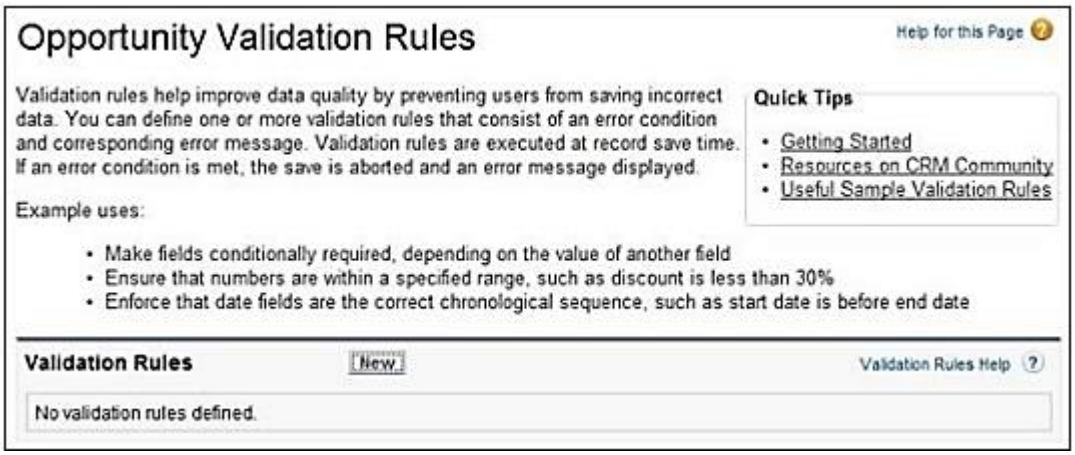
- Validation Rules
- Workflow Rules
- The Approval Process
- The Process Builder
- Flow Builder

13.1 Data validation rules

Data validation rules can be applied to both custom and standard fields and are used to verify that the data entered in a record meets the criteria you have specified before the record can be saved.

Validation rules contain a formula or expression that evaluates the data in one or more fields and returns a value of either true or false.

When validation rules are defined for a field or set of fields, the following actions are fired when the user creates a new record or edits an existing record and then clicks on the **Save** button:
1. Salesforce executes the validation rules, and only if all data is valid is the record then saved.
2. For any invalid data, Salesforce displays the associated error message without saving the record.
You can specify the error message to be displayed when a validation rule gets fired with the option to show the error message inline next to a field firing the validation rule.
To begin using validation rules, navigate to **Setup** | **Customize**.
For standard objects, go to the appropriate activity, standard object, or the user's link from the menu, and click on **Validation Rules**.
For custom objects, navigate to **Setup** | **Create** | **Objects**. Now, go to the custom object.
Validation rules are listed in the **Validation Rules** related list.
As shown, to begin adding a new validation rule, click on the **New**button in the **Validation Rules** section, as shown in the following screenshot:



The field description section
Add a **Rule Name**, which is a unique identifier of up to 80 characters with no spaces or special characters such as extended characters. The **Active** checkbox that is used to set that the rule is enabled. Fill in the **Description** field, which is an optional 255 character or less textbox that you can set to describe the purpose of the validation rule.
The Error Condition Formula section
The formula that is entered here forms the expression that is used to validate the field.
The Error Message section
The **Error Message** field is the text to be displayed to the user when a record update fails the validation rule.
Error Location is used to determine where on the page the error is displayed to the user. The options that are available are:

- **Top of Page**
- **Field**

The **Top of Page** option sets the error message to be displayed at the top of the page. To display the error next to a field, choose the **Field** option and then select the appropriate field.
If the error location is a field, the validation rule is also listed on the **Detail** page of that field.
You can click on **Check Syntax** to check your formula for errors and finally, click on **Save** to finish or **Save & New** to create additional validation rules.

13.2 Workflow rules

Workflows in Salesforce are a fantastic way to automate certain business processes. You can create a rule, and based on certain criteria that you set, Salesforce can do a number of things, like send an email, create a task, or update a field.

13.2.1 Creating a workflow rule

A workflow in Salesforce is a great way to automate business processes. In order to create a workflow rule, you need 2 things:

1. Criteria that cause that rule to run (in this case, when a person's lead score hits a certain number)
2. Actions that execute when a record matches the criteria (in this case, an email alert)

Navigate to **Setup > App Setup > Create > Workflow & Approval > Workflow Rules**.
Select **"New Workflow Rule"** and then select the object you want the rule to apply to. We are interested in leads.

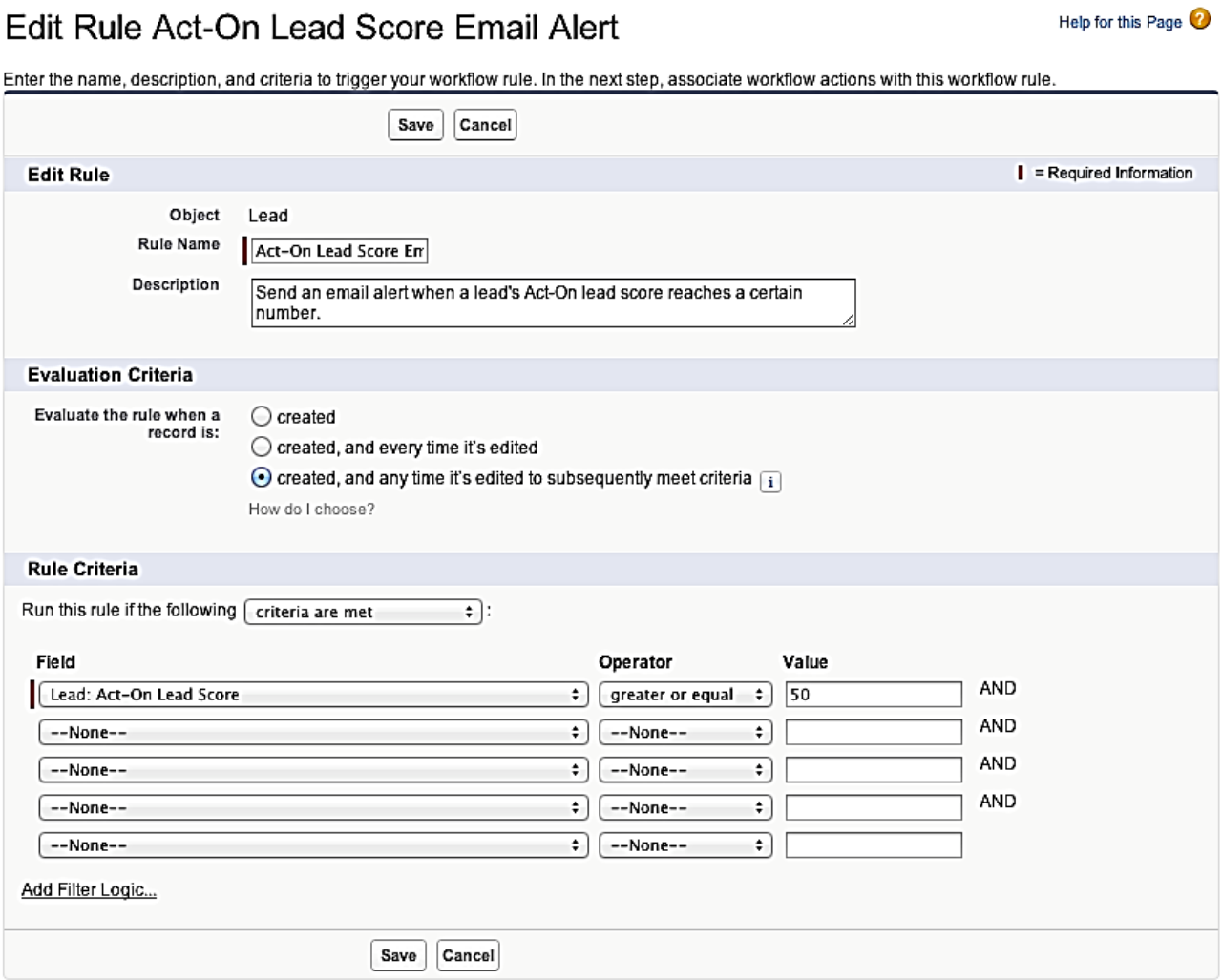
Name your rule, and give it a description. I named my rule **"Act-On Lead Score Email Alert,"** and described it thusly: **"Send an email alert when a lead's Act-On score exceeds 50."**
Next, decide on your evaluation criteria. There are 3 options here.
Salesforce will evaluate the rule when a record is:

1. **Created**
2. Created, and every time it's edited
3. Created, and every time it's edited to subsequently meet criteria.

The first option will only check if the rule applies when the lead is created, which means it will ignore updates to existing leads. This definitely doesn't apply, since we want our rule to apply to existing records.
The second option means that if the rule criteria is met, the rule will run, no matter what. Say we have a lead whose lead score is greater than 50. The rule will run when the record is edited to show that the lead score is above 50, but the rule will also run every single time that record is edited for any reason and the lead score is still above 50. This will result in an email about that lead, even after it's already been determined that they're a hot lead.
The third option means that the rule will run if a new record is created that meets the criteria, or if an existing record is changed from not meeting the criteria to meeting the criteria. This is the best option for our rule. If a hot lead is added as a new record, we'll receive an alert. If a record is subsequently updated to have a lead score over 50, we'll receive an alert only once that happens.
Next, set your rule criteria.
We've chosen:

- **Field: Act-On Lead Score**
- **Operator: greater or equal**
- **Value: 50**

You can choose to add filter logic here, if need be. Perhaps in the future we will only want an email kicked out for leads who have an email address at a company. We would add another criterion that ignores emails from Gmail, Yahoo, etc. and then we would add 'AND' filter logic so that both of those criteria have to be met for the rule to run.



13.2.2 Setting up workflow actions

Now that you've set up your rule, you need to specify the workflow actions that the rule will trigger. On the detail page for your rule, there will be a section called 'Workflow Actions'—clicking Edit, and then Add Workflow Action, will present you with a dropdown menu of the actions available. I won't go through the details of setting up each of these specific actions, but I've included the link to the Salesforce Help & Training document for each, in case you need further instructions.

1. **New Task**
2. **New Email Alert**
3. **New Field Update**
4. **New Outbound Message (this action allows you to send a SOAP message with field values to an endpoint such as an external server)**
5. **Select Existing Action (this action allows you to use workflow actions you've created for other workflow rules or approval processes to be associated with your new rule)**

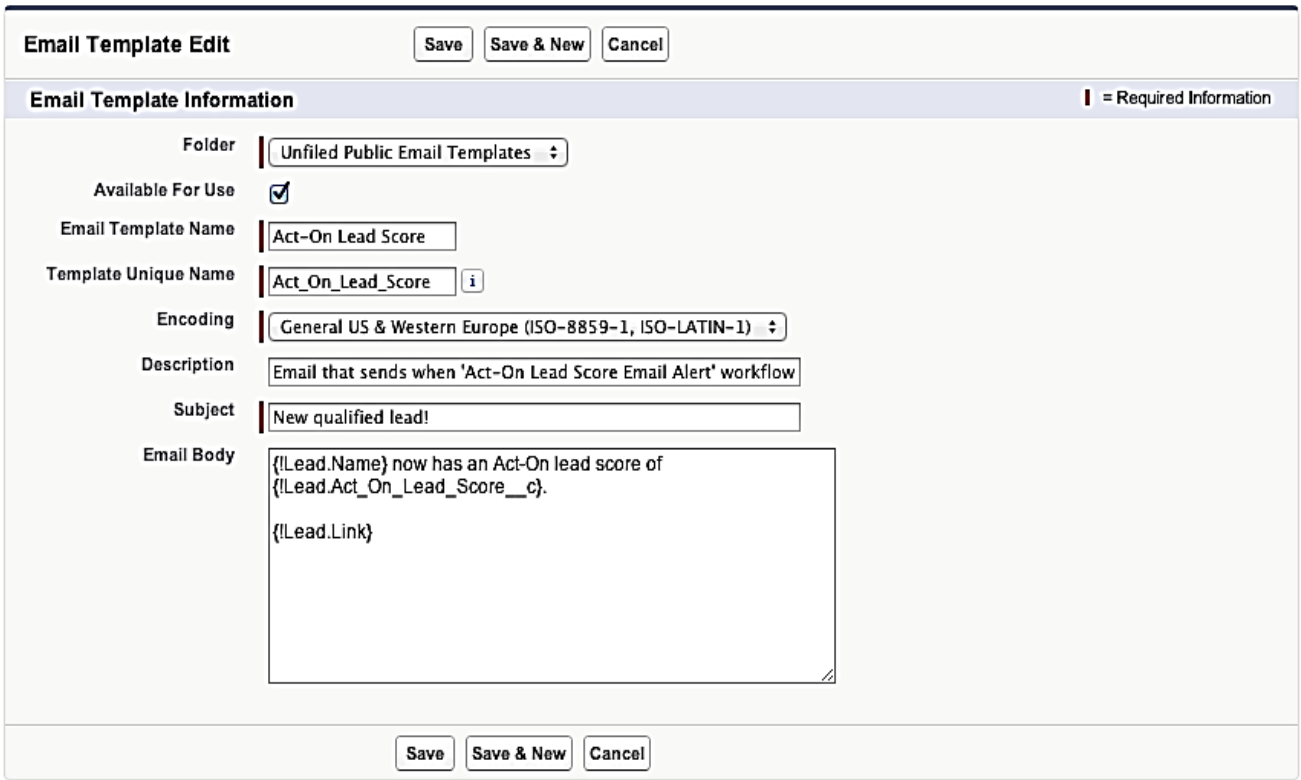
13.2.3 Setting up an email alert

I will briefly outline how to set up an email alert, as that is the action our scenario requires.
Before selecting 'New Email Alert' from the workflow action dropdown menu, you will need to set up an email template in Salesforce.
To do this, navigate **to Setup > Administration Setup > Communication Templates > Email Templates.**
Click 'New Template.'

Choose the type of template you want to create. There are four types of templates available:

1. **Text**
2. **HTML (using Letterhead)**
3. **Custom HTML (without using Letterhead)**
4. **Visualforce (Visualforce is a native Salesforce development framework, which includes its own mark-up language that is similar to HTML)**

Above the text box that says 'Step 2. Text Email Template…', note the section that says, 'Available Merge Fields.' From this section, you can select fields from any object to be included in your email template. When the email is sent, these fields will be replaced with information from the designated records.
Here is what our email template looks like:



As you can see, we're using merge fields to populate the email with the lead's name, Act-On Lead Score, and a link to the lead record so that the reader can navigate to that lead from the email.
Once your email template is set up, you can finish setting up your workflow action.
Once again, navigate to the detail page for your rule, and in the section called 'Workflow Actions'—click Edit, and then Add Workflow Action, and select 'New Email Template' from the dropdown. Enter a description for the email alert; ours is 'new qualified lead.' Choose the email template you set up previously. Select one or more of your Salesforce users to receive the email, and add any additional outside email addresses that should receive the alert to the 'Additional Emails' field. Choose your 'From Email Address' and then hit Save**.**

13.2.4 Adding time-dependent workflow actions

To add a time-dependent workflow action, click on **Add Time Trigger** in the **Time Dependent Workflow Actions** section. Then, specify the number of days or hours before or after a date relevant to the record, such as the date on which the record was created or modified or even for an opportunity close date, as shown in the following screenshot:

The **Add Time Trigger** button is unavailable if either:
The rule criteria is set to **Every time a record is created or edited**
The rule is already active (here, you must temporarily deactivate it in order to apply the action)
The rule is deactivated but has pending actions in the workflow queue
Additional, immediate, or time-dependent actions can now be configured, and then finally, click on the **Done** button in the top-right corner of the screen.

13.2.5 Activating your workflow rule

The last step in the process is to activate your rule. On the Workflow Rules landing page, find the name of your new rule. To the left, there should be three links: **'Edit | Del | Activate.'** Click **'Activate,'** and you're finished!

13.3 Approval process

The building of the approval processes also requires the setting up of the required actions to be taken after the record is either first submitted, approved, rejected, or recalled for approval. Approval processes require a good understanding of your business rules and processes in order to be successfully implemented.
Approval processes are similar to workflow rules in the sense that they can invoke the same key actions. However, there are significant differences between workflow rules and approval processes, such as:

- Workflow rules are activated when a record is saved, whereas approval processes are triggered by explicitly clicking on the **Submit for Approval** button.
- Workflow rules consist of a single step and a single action. Approval processes consist of multiple steps where a different action is taken based on whether the record is approved or rejected.
- Workflows can be modified or deleted. In approvals, some attributes cannot be modified, and approval processes must be deactivated before outstanding approvals can be deleted.
- Approval processes result in the approval history being automatically tracked, which is not applied to workflow rules.
- When an approval is initiated, the record is "locked down" and cannot be changed by someone other than the approver or system administrator until the record has completed the approval process.

13.3.1 Configuring approval processes

To create an approval process, follow these steps:

1. Launch the approval process wizard.
2. Specify **Name**, **Unique Name**, and **Description**.
3. Specify **Criteria** for **Entering Process**.
4. Specify **Approver Field** and **Record Editability**.
5. Select **Email Notification Template**.
6. Configure **Approval Request Page Layout**.
7. Specify **Initial Submitters**.
8. Activate the approval process.

Choosing an approval process wizard

When you click on the **Create New Approval Process** button to start creating an approval process, you are presented with these two options in which you can build the process:

UseJump Start Wizard or **Use Standard Setup Wizard**. The upcoming sections outline the differences between these two mechanisms.

The Jump Start Wizard option

The **Jump Start Wizard** option is provided as a quick way to create simple approval processes that have a single step. To simplify the settings, the Salesforce CRM application automatically determines some default options for you with this option.

The Standard Setup Wizard option

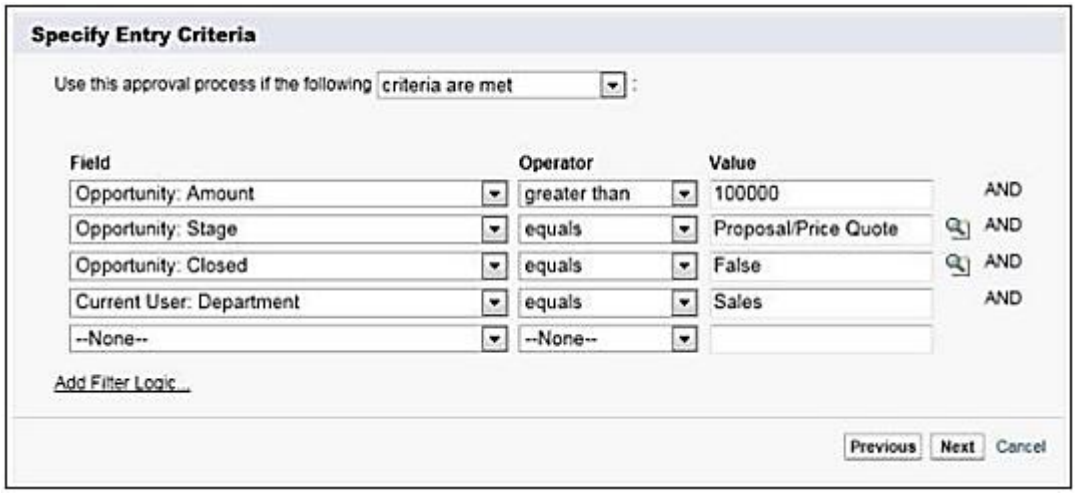
The **Standard Setup Wizard** option enables the creation of complex approval processes and is used where multiple processing steps are required. This option provides the mechanisms to define your process and then uses a setup wizard to define each step within that process.

To create an approval process, navigate to **Setup** | **Create** | **Workflow & Approvals** | **Approval Processes**.

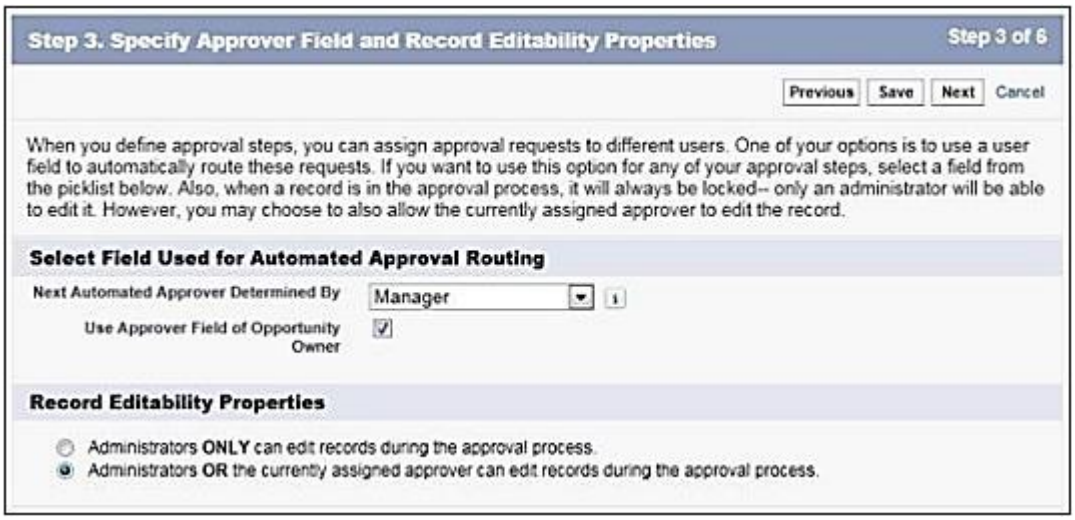
Choose the object for the new approval process, click on it, and then select **Use Standard Setup Wizard**, where we have selected **Opportunity**, as shown next:

To set the details for the approval process, do this in the **Step 1. Enter Name and Description** page (where this is step 1 of 6): enter **Process Name**, a **Unique Name**, and optionally, **Description**, and then click on **Next**

To specify the entry criteria, which is an optional step in the **Step 2. Specify Entry Criteria** page and is used to determine the records that enter the approval process, you can either choose from the formula logic or you can select certain fields, operators, and values to specify when the desired criteria are met. This is shown in the following screenshot, which is presented as step 2 of 6:



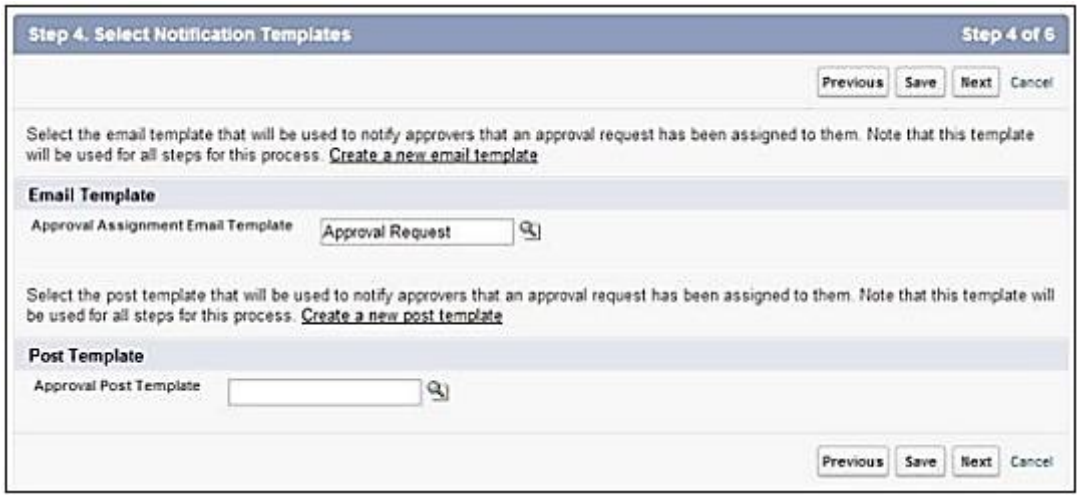
In **Step 2. Specify Entry Criteria**, either enter the filter criteria for records that are to be included by this approval process or leave all the filters blank to have all records submitted within the approval process.
In **Step 3. Specify Approver Field and Record Editability Properties**, using the **Next Automated Approver Determined By** picklist, select a user field if you want the Salesforce CRM application to automatically assign approval requests to an approver based on the value in the user field. For example, you might want to automatically send approval requests to a user's manager, as specified in the user's **Manager** field, as shown next:



To allow users to manually choose another user who will approve any approval requests, leave the **Next Automated Approver Determined By** field blank:

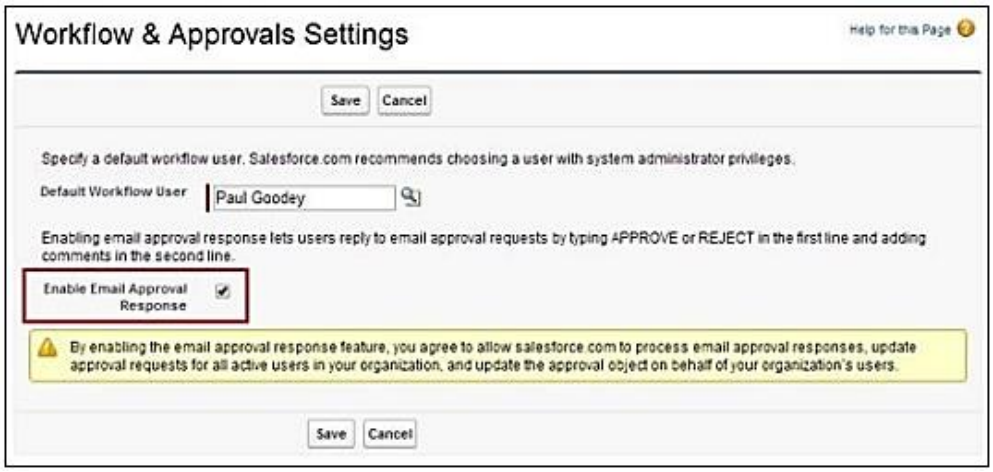
By selecting the **Use Approver Field of Record Owner** checkbox, you can set the approval process to use the standard **Manager** field or a custom field on the record owner's user record instead of the submitting user's record.
When an approval process assigns an approval request to a user, Salesforce.com automatically sends the user an approval request e-mail. This e-mail contains a web link that the user can click on to access the approval page within the Salesforce CRM application, which lets the user approve or reject the request and also enter comments.

In **Step 4. Select Notification Templates**, you can choose a custom e-mail template that can be used when notifying an approver that an approval request has been assigned to them, as shown next for the example **Approval Request** template.
Alternatively, by leaving this field blank, a simple default e-mail template is used. In addition, if **Approvals in Chatter** has been enabled, you can also choose **ApprovalPost Template**, as shown in the following screenshot:

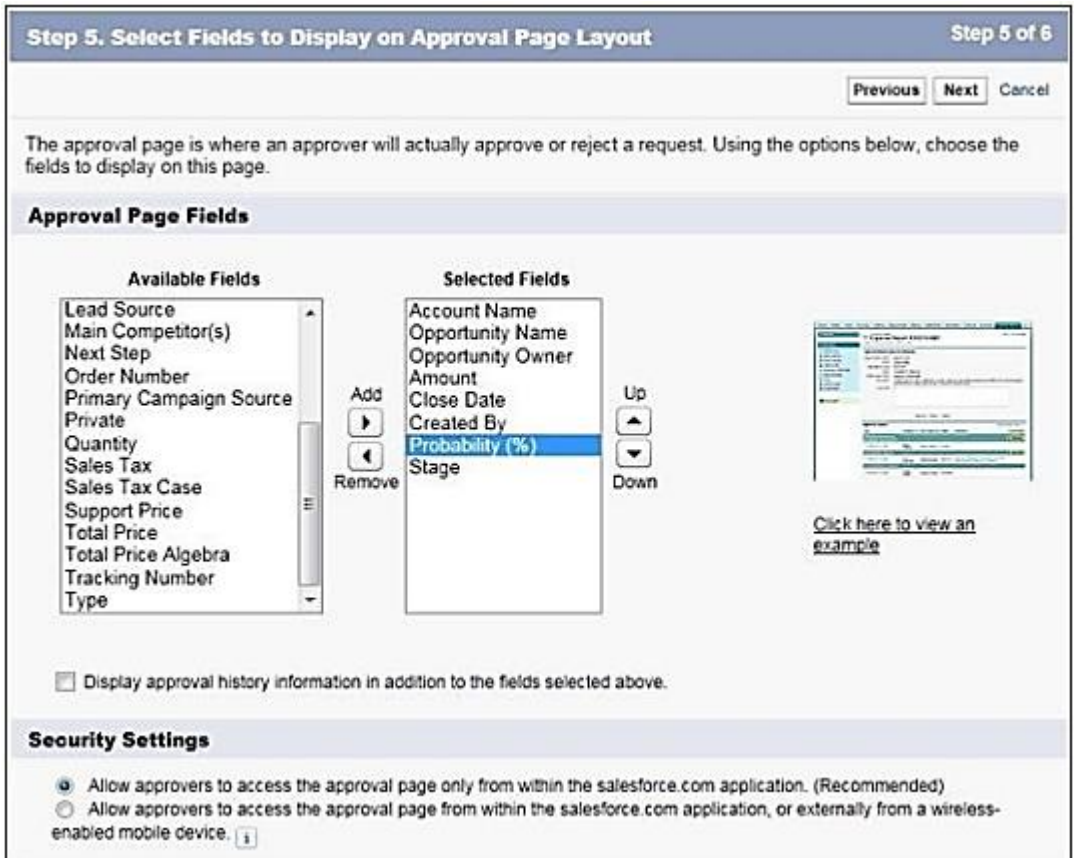


Email Approval Response

By enabling the e-mail approval response, the user can alternatively reply to the e-mail by typing approve, approved, yes, reject, rejected, or no in the first line of the e-mail body. To enable Email Approval Response, navigate to to**Setup** | **Create** | **Workflow &Approvals** | **Settings**. Now, check the **Enable Email Approval Response** checkbox, as shown in the following screenshot:

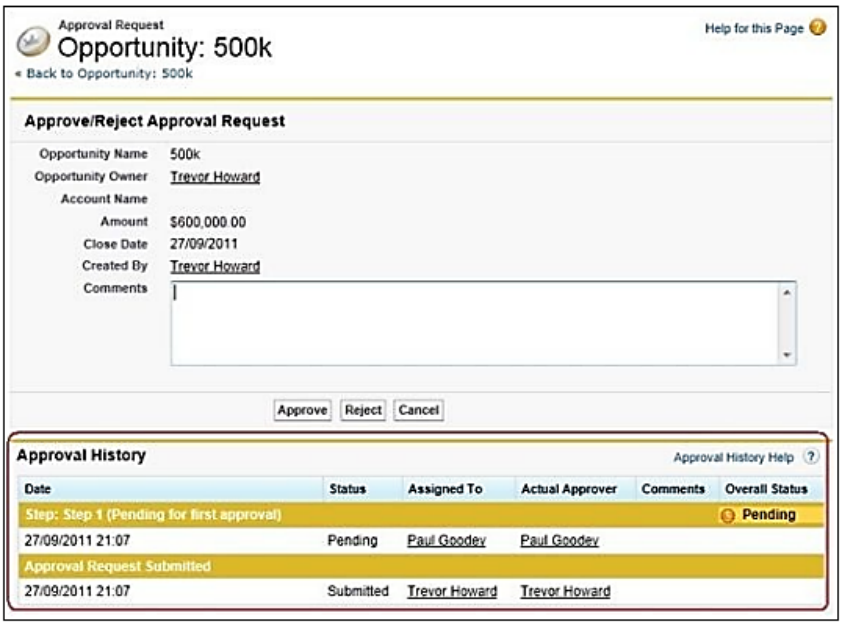


Step 5. Select Fields to Display on Approval Page Layout page, the option to configure the approval request page layout can be carried out as shown in the following example:

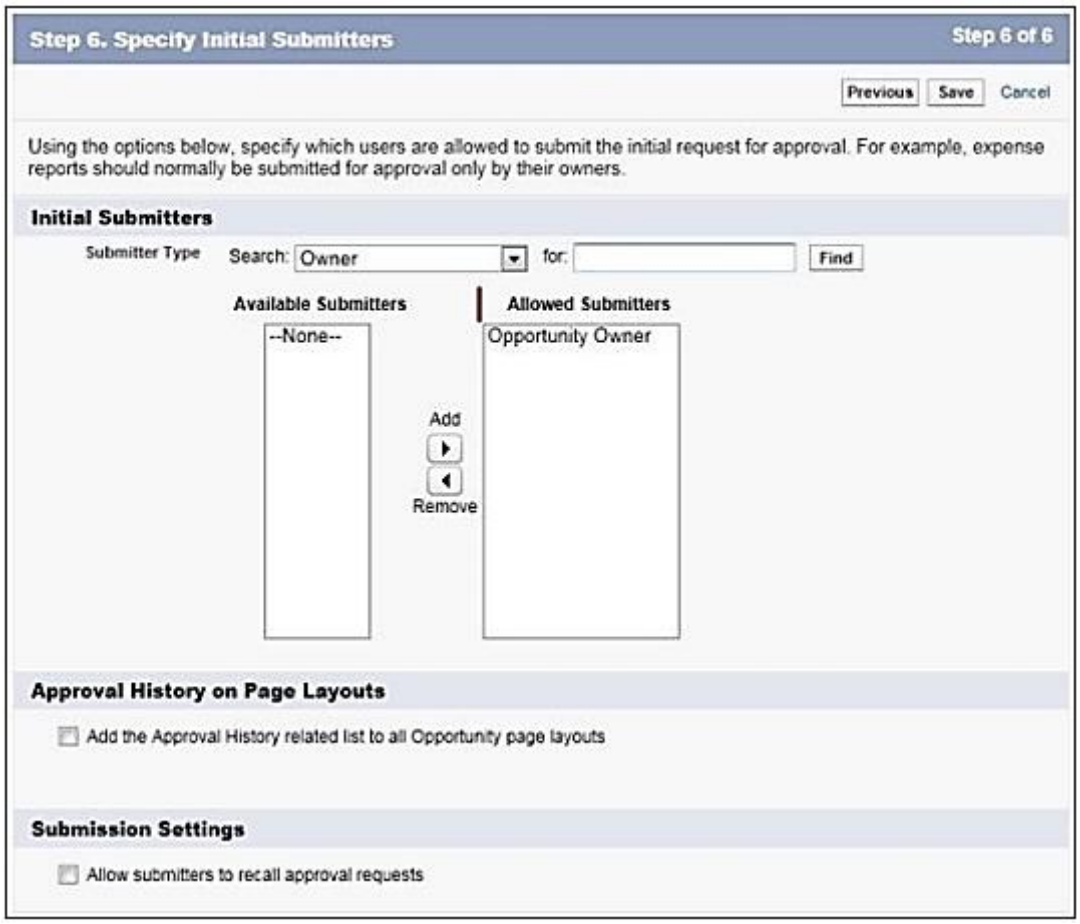


The approval page is where an approver approves or rejects a request, and it is in **Step 5. Select Fields to Display on Approval Page Layout** of configuring the approval process where you can carry out the following:

1. Select and sort the fields you want to display on the approval request page.
2. Select **Display approval history information** in addition to the field selected previously to include the **Approval History** related list that displays the **Date**, **Status**, **Assigned To**, **Actual Approver**, **Comments**, and **Overall Status** columns on the resulting approval request page, as shown in the following screenshot:



3. To specify how approvers can access an approval page, select either **Allow approvers to access the approval page only from within the application. (Recommended)** or **Allow approvers to access the approval page from within theSalesforce.com application, or externally from a wireless-enabled mobile device**, and then click on **Next** (as shown previously in the step 5 screenshot).
4. Now, specify **Initial Submitters**, as shown in the following screenshot:

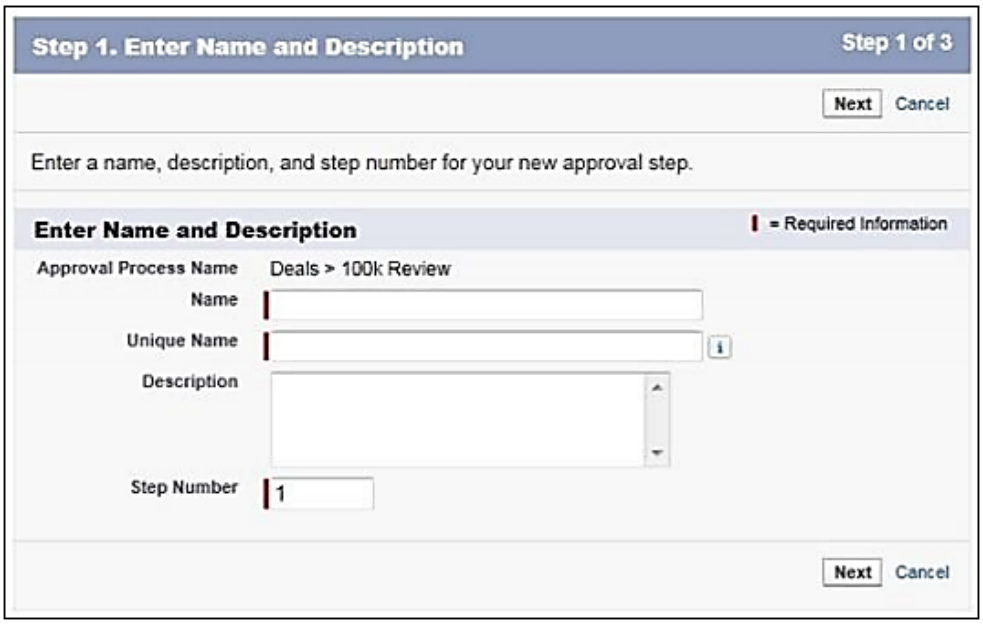


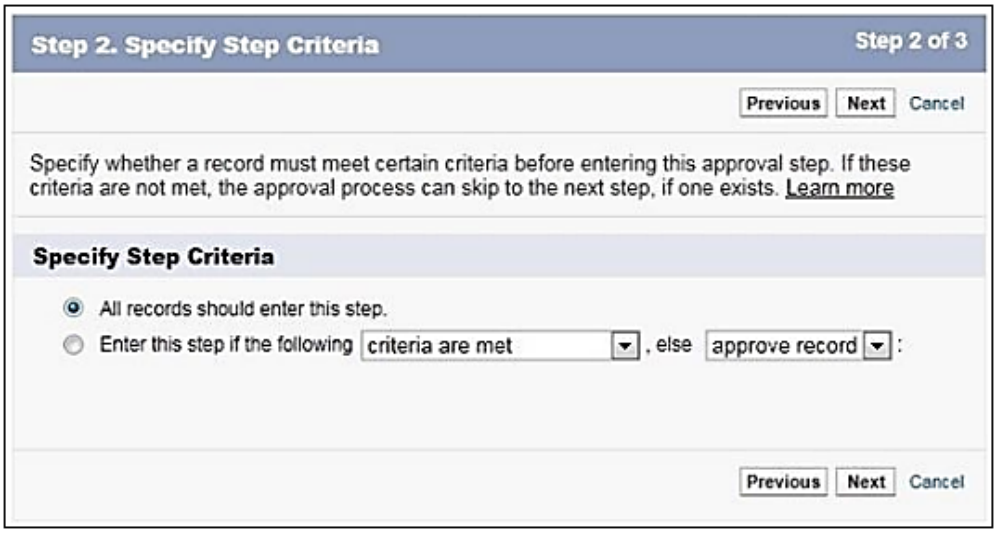
5. Specify which users are allowed to submit records for approval.
6. Optionally, select **Add the Approval History related list to all [object name] Pagelayouts** (as shown for the Opportunity object-related approval). This will automatically update all the page layouts for this object and include a related list That allows users to view and submit approval requests.
7. Optionally, select **Allow submitters to recall approval requests** to give Submitters the option to withdraw their approval requests.

13.3.2 Creating approval steps

To create an approval step, navigate to **Setup** | **Create** | **Workflow & Approvals** | **Approval Processes** and select the name of the approval process, and then carry out the following:

- Click on the **New Approval Step** button from the **Approval Steps** related list section
- Enter the **Name**, a **Unique Name**, and an optional **Description** fields for the approval step
- Enter a step number that positions the step in relation to any other step in the approval process, as shown next, and click on **Next**:

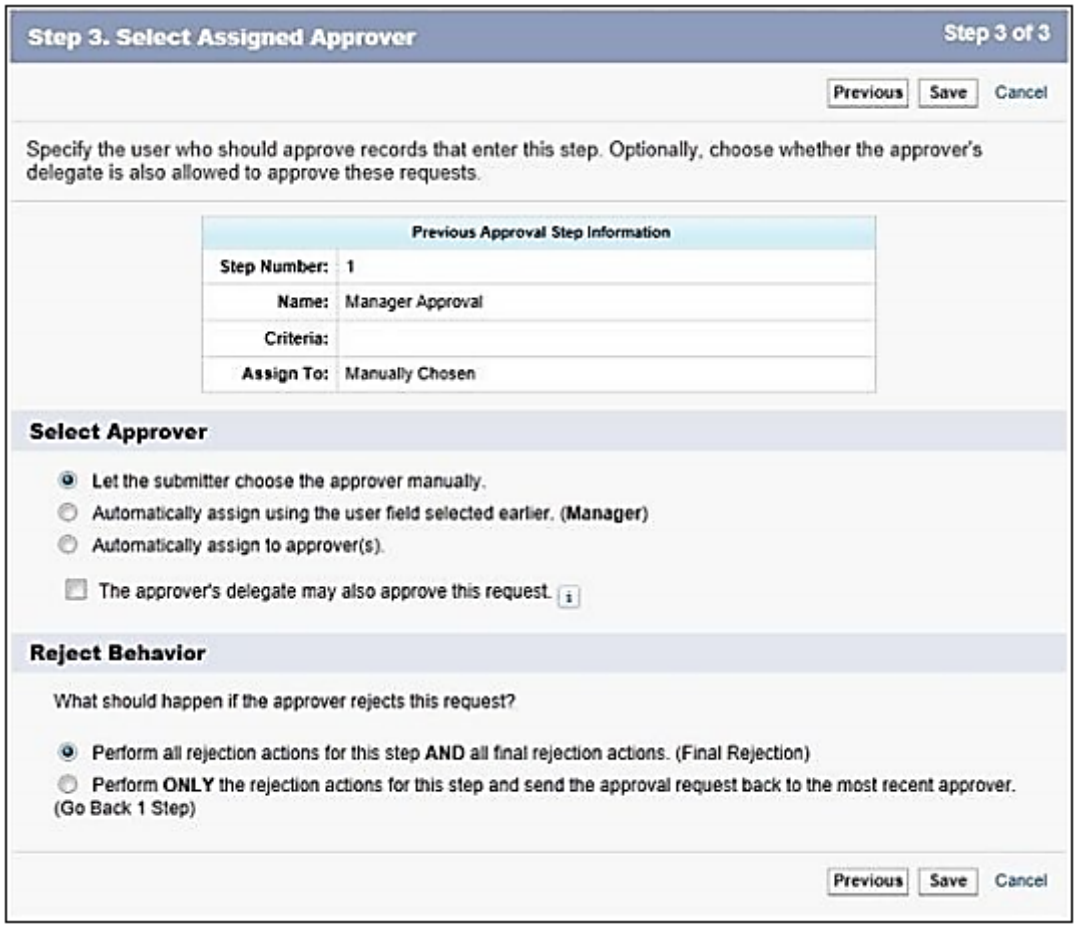




If you specified the filter criteria or entered a formula, you can choose what should happen to records that do not meet the criteria or if the formula returns False, where the options are:

- **Approve record**: This automatically approves the request and performs all final approval actions.
- **Reject record**: This automatically rejects the request and performs all final rejection actions. This option is only available for the first step in the approval process.

Now, click on **Next** to display the **Step 3. Select Assigned Approver** page, where you specify the user who should approve records that enter this step and optionally, choose whether the approver's delegate is also allowed to approve these requests, as shown next:



The options are:

- **Let the submitter choose the approver manually**, which prompts the user to manually select the next approver

- **Automatically assign using the user field selected earlier**, which assigns the approval request to the user in the custom field that is displayed next to this option (where this custom field was selected during the earlier configuration of the approval process, which, in the preceding screenshot, is **Manager**)

- **Automatically assign to approver(s)**, which allows you to assign the approval request to one or more users or related users, as shown next:

If you specify multiple approvers in the **Automatically assign to approver(s)** option, choose one of the following options:

- **Approve or reject based on the FIRST response**, whereby the first response to the approval request determines whether the record is approved or rejected.
- **Require UNANIMOUS approval from all selected approvers**, whereby the record is only approved if all of the approvers approve the request. If any of the approvers reject the request, then the approval request is rejected.

Also, on this page is the option to specify **the approver's delegate may also approve this request**, where the delegate user is set in the **Delegated Approver** field on the assigned approver's user page.

Note

Delegated approvers cannot reassign approval requests, and they are only permitted to approve or reject approval requests.

If this is not the first step in the approval process, you must specify what will happen if the approver rejects a request in this step, as shown in the following example, where the options are as follows:

- Perform all rejection actions for this step and all final rejection actions (Final Rejection).
- Perform only the rejection actions for this step and send the approval request back to the most recent approver (go back one step).

Now, click on **Save** and specify any workflow action you want to set within this step using the following options:

- Yes, I'd like to create a new approval action for this step now
- Yes, I'd like to create a new rejection action for this step now
- No, I'll do this later. Take me to the approval process detail page to review what I've just created

Finally, click on **Go!** to complete the approval process.

13.4 Process Builder

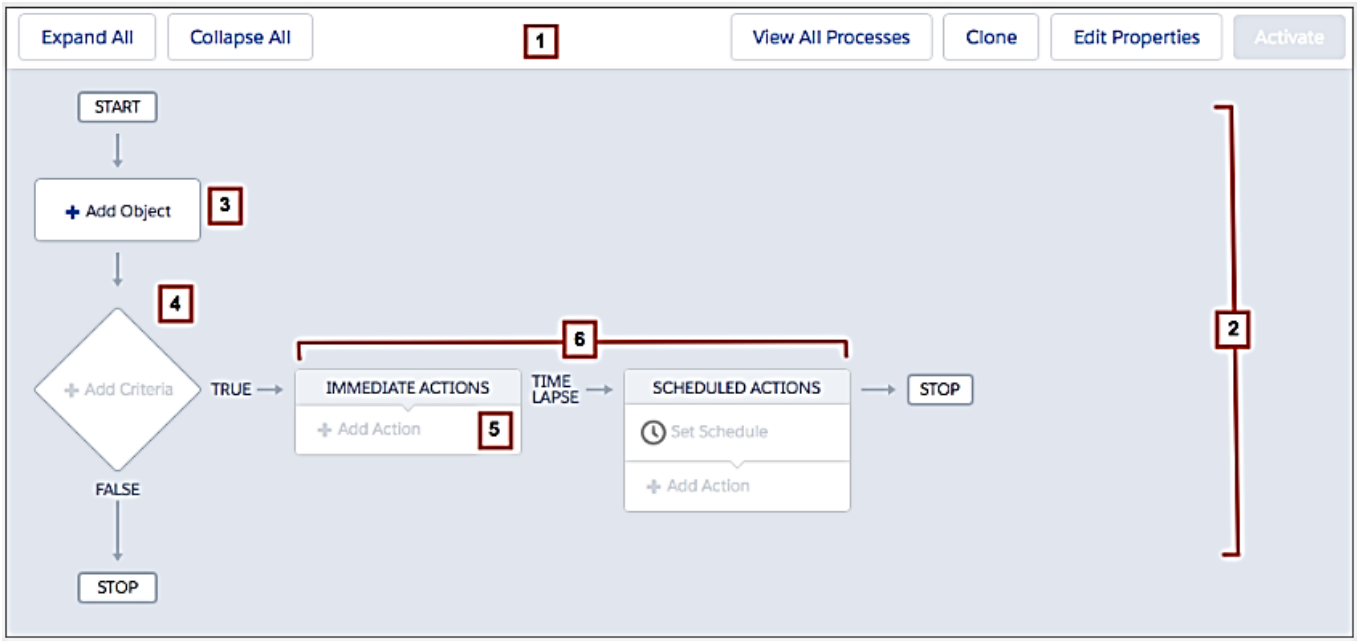
The Process Builder tool allows you to easily automate business processes using a convenient graphical representation of your process as you build it. Automated processes in the Process Builder consist of:

- Criteria that determine when to execute action groups
- Immediate and scheduled actions to execute when those criteria are met

Any change that causes a record to match the criteria automatically triggers the action group.

13.4.1 Process Builder User Interface

Before beginning, click around and become familiar with the user interface. From Setup, enter Process Builder in theQuick Find box, then click Process Builder. First up, click New, name your process, and click Next. Now you get to a page that looks something like this.



Create Automated Processes

Create a simple process that creates a contract record and a follow-up task when a high-value opportunity is closed and won.

1. From Setup, enter Process Builder in the Quick Find box, click **Process Builder**, and then click **New**.
2. Name the process Opportunity Management. The API name updates to Opportunity Management when you tab out of the Name field.
3. For the description, enter If a high-value opportunity is closed and won, create a draft contract and a follow-up task for the account owner.
4. Click **Save**.
5. Click **AddObject** to associate your process with an object and specify when to start the process. For this process, let's choose Opportunity and start the process when the record is created or edited.

6. Click **Save**.

Add Criteria

Now let's define the criteria that must be true before the process can execute the associated actions. For this process, we want to check whether the opportunity has been closed and won, as well as if it's high-value. For this example, "high-value" means it's worth more than $250,000.

1. Click Add Criteria.

2. Name the criteria "Opportunity Won".

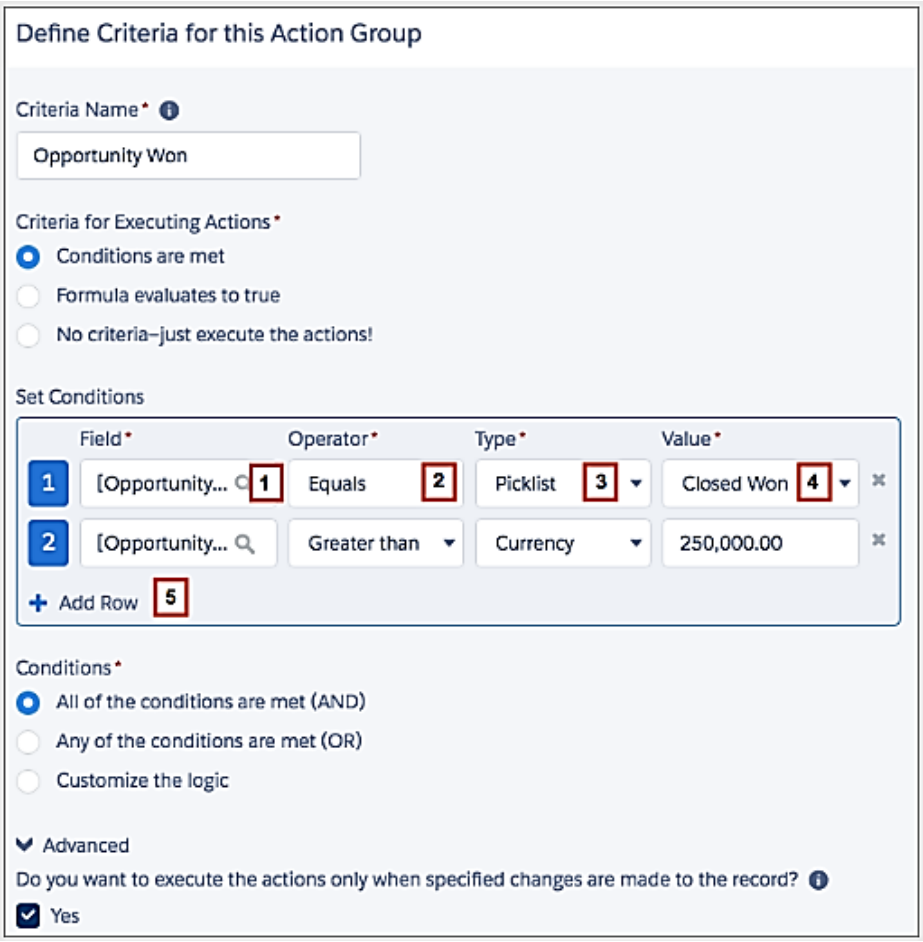
3. Leave Conditions are met selected to evaluate specified field values for the opportunity record.

4. Set the first filter condition. The process needs to check whether the opportunity has been closed and won.

   a. Click in the Field box **(1)**, choose **Opportunity** > **Stage**, and click **Choose**.
   b. For the operator **(2)**, leave **Equals** selected.
   c. For the type **(3)**, leave **Picklist** selected.
   d. For the value **(4)**, select **Closed Won**.

5. Set the second filter condition. The process needs to check whether the opportunity is high-value.

   a. Click **Add Row (5)**.
   b. Click in the Field box, select **Opportunity** > **Amount**, and then click **Choose**.
   c. For the operator, select **Greater than**.
   d. For the type, leave **Currency** selected.
   e. For the value, enter $250,000.00.



6. In the Conditions area, leave **All of the conditions are met (AND)** selected. This field lets you specify which combination of the filter conditions must be true for the process to execute the associated actions.
7. Click **Advanced** and select **Yes (6)**. When you select this option, the process ignores record changes that aren't relevant to your defined criteria. For example, if a user edits the record by adding a description, the process won't execute the associated actions. Note: This setting isn't available if:
   - Your process starts only when a record is created.
   - Your process starts when a record is created or edited and the criteria node doesn't evaluate any criteria.
   - The criteria node evaluates a formula, but the formula doesn't include a reference to the record that started the process.
     - Your process uses the Is changed operator in a filter condition.
8. Click **Save**.

Add Actions to Execute when the Criteria are Met

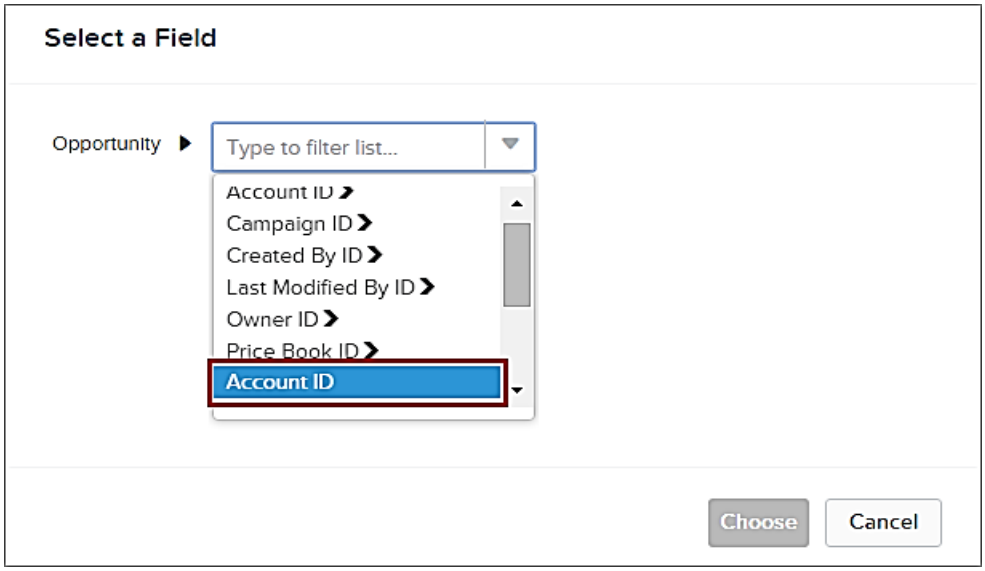
Now let's define the actions that execute when the criteria are met. When an opportunity is closed and won and the amount is greater than $250,000, the process creates a draft contract record associated with the opportunity's account and a follow-up task for the associated account's owner. First up, let's create a contract.

1. Under Immediate Actions, click **Add Action**.
2. For the action type, select **Create a Record**.
3. Name the action Create Draft Contract.

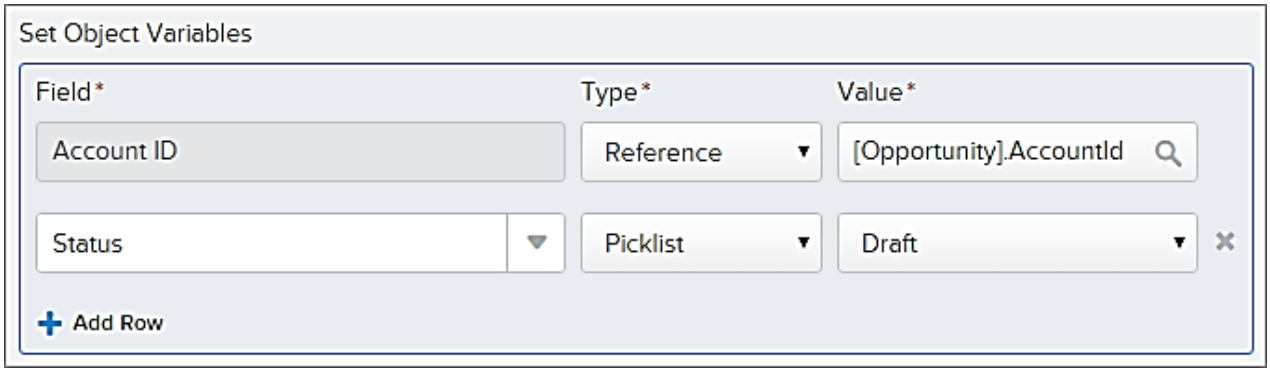
4. For Record Type, select **Contract**.

Certain fields are required when you create a record. When you select the object that you want to create a record for, the Process Builder automatically displays rows for each of that record's required fields. When you select Contract, a row for Account ID shows automatically.

5. Associate the new contract with the opportunity's parent account by selecting a value for Account ID.
   a. For type, select **Reference**.
   b. Click in the Value box to choose a value.
   c. Select **Opportunity** > **Account ID** and then click **Choose**.



6. Make sure the new contract is a draft. In the Value for Status, select Draft from the drop-down list.



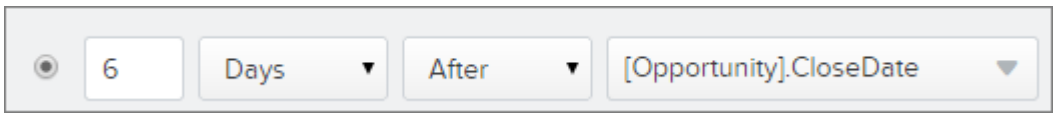
7. Click **Save**.

Now let's create a high priority follow-up task for the associated account's owner. We'll use a schedule so the owner can follow up with the account six days after the opportunity closes.

First, set up a schedule. A schedule lets the process know that it must wait to execute the associated actions. Because you can configure multiple schedules for the same criteria node, each schedule has its own list of actions to execute.

a. Start the process only when a record is created (1). Select this option when you choose an object for your process.
   b. Start the process when a record is created or edited (2). In addition, select the advanced option to execute actions only when specified changes are made (3) when you add criteria to your process.

1. Under Scheduled Actions, click **Set Schedule**.
2. Set the schedule for six days after the opportunity closes.



3. Click Save.

Now that you have a schedule, add the task creation action to it.

1. Under Scheduled Actions, click Add Action.
2. For the action type, select Create a Record.
3. Name the action "Follow-up Task".
4. For the record type, select Task.
5. Configure the task.

Field	Value
Assigned to ID	Opportunity > Account ID > Owner ID
Priority	High
Status	Not Started

6. Click **Save.**

Nice work! You've created a process that automatically manages your high-value business opportunities. To start using this process, just activate it.

13.5 Get Started with Flows

The Power of Automation

Over time, most organizations establish a set of practices that meet various business requirements. These practices are often unique to your business, which makes it hard to find an out-of-the-box solution that exactly meets your requirements. Maybe your company is a high-touch call center with specific scripts that your agents need to follow based on a customer's account data. Maybe you have a practice in place where your salespeople should always, always, create a work order after a deal closes. Or maybe you have a task that your users do over and over and over again, and if you can find a way to shave a few minutes off that task, you can save hundreds of person hours over the course of a year.

One tenet of good user experience is, "If the user can only do one thing, do it for them." If you always create a work order after a deal closes, can you create it automatically? If your call agents always follow a set script when collecting customer information, does it make sense to script that right into the user interface? If your users collect data related to three different objects, can you save time by having them enter the data in one form, and then create the three different records for them?

This is the power of automation. Save your users' time and make sure required tasks are being done. Improve the quality of your data. Be the superhero that your company deserves!

Why Go with the Flow?

In general, it's best to start with the simplest declarative, no-code tools and work your way up to code solutions. While flows are a no-code solution, they're by no means the simplest one. Flows can be better described as visual coding—they're declarative, but they require that you understand some programming concepts like variables and how logic works.

You have a bunch of automation tools at your disposal: formulas, validation rules, processes, flows, and Apex. You also have some more specific tools, like escalation and auto-response rules. Whether you should build a flow depends on what kind of business process you're trying to automate.

Flows are useful for two major use cases: behind-the-scenes automation and guided visual experiences.

Behind-the-Scenes Automation

If your business process runs behind the scenes, and it should start automatically, such as when a record changes, you have your pick of solutions.

Build a process in Process Builder.

Build an auto-launched flow in Flow Builder.

Build an Apex trigger with Apex code.

Don't start with a flow because it isn't the simplest solution for the job. Instead, start with a process. If your business process needs more complex functionality than is available in Process Builder, create an auto-launched flow in addition to your process. The auto-launched flow can be called from the process to handle the complex functionality required.

Guided Visual Experiences

If your business process requires input from a user, you can use a:

Screen flow

Lightning component

Since it's the simpler solution of the two, start with a screen flow. Build rich screens that guide users through your business process, and automate the necessary logic and actions between those screens.

A Note About Terminology

You may have heard several terms used interchangeably when referring to flows. As a reminder, the official terms are:

Lightning Flow—the product that encompasses building, managing, and running flows and processes.

Flow Builder—a point-and-click tool for building flows.

Flow—an application that automates a business process by collecting data and doing something in your Salesforce org or an external system.

In short, the Lightning Flow product includes multiple tools. One of them, Flow Builder, helps you create flows.

Note

You may also have heard the term Visual Workflow. That's a retired product for designing, managing, and running flows. Visual Workflow has been replaced by Lightning Flow.

Business Requirements

As a Salesforce admin, you've been lobbying your stakeholders to invest in automating more of your company's business processes, often focusing on the efficiency improvements that Salesforce automation tools offer. Your company's head of sales asks you to improve the efficiency of the sales teams.

When sales reps log contacts in Salesforce, they often ignore many of the fields, a practice that results in rogue, accountless contacts. To make matters worse, the sales reps frequently create duplicate contacts. They can avoid this by searching Salesforce before they create the contact, but it'd be better if this process were automated. After all, you've been touting how Salesforce can make the company more efficient.

The Use Case

Let's break it down.

Capture values for only the required fields (First Name and Last Name) and the associated account.

If there's a matching contact, update the existing one. If there isn't a matching contact, create one.

To round out this business process, it would be good to provide some sort of confirmation that the business process has finished. If we note what the flow did in Chatter, more users can access that information than if we communicated through a closed channel like email. Let's add two more requirements.

Confirm what happened by posting on Chatter.

Confirm to the user that the business process is complete.

The Solution

Since the whole point of automation is for the system to do things automatically, we need a solution that offers conditional logic and actions. This particular business process requires information from the user, so we also need a form that captures that information. Let's take a look at how you can address this use case in Salesforce.

Solution	Form	Conditional logic and actions
Quick action	Yes	No
Flow	Yes	Yes
Lightning component	Yes	Yes

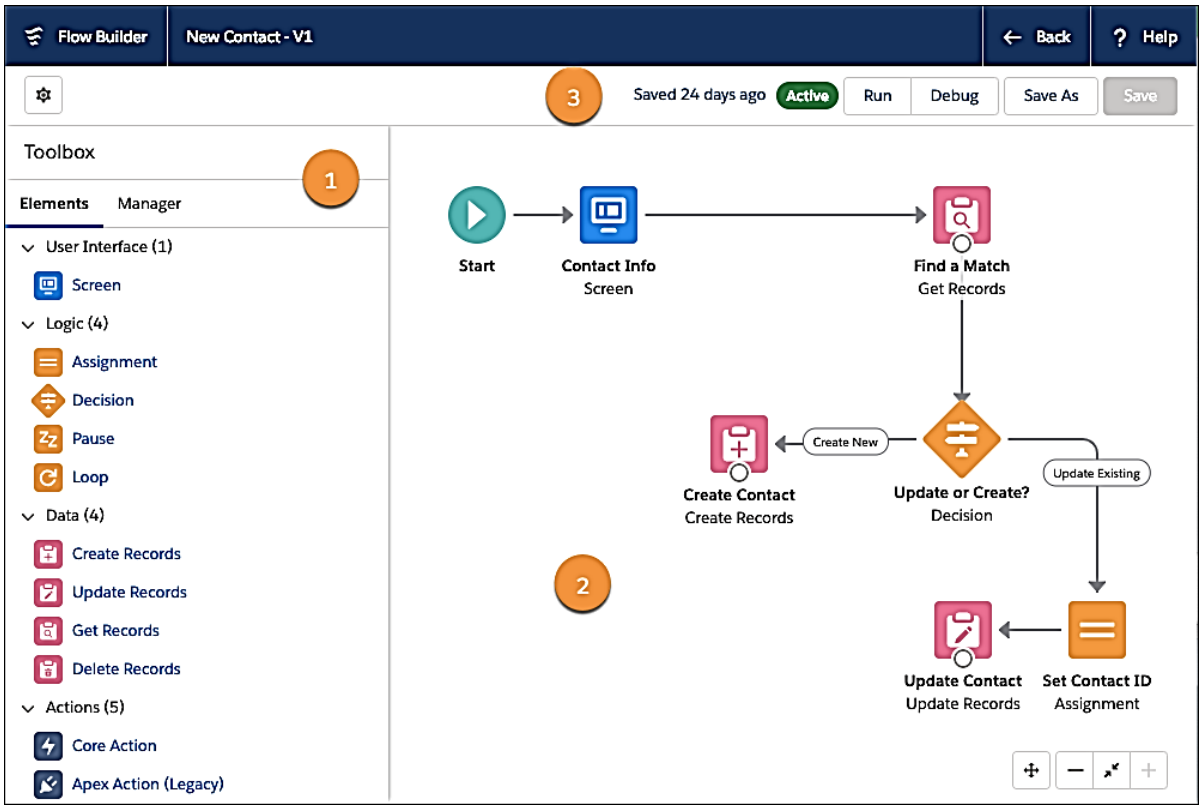
It's best to start with declarative, no-code tools and work your way up to code solutions, so we avoid building a Lightning component, which requires coding. Since our use case requires conditional logic and actions, the simplest solution that fits our requirements is a flow.

Before we move on to actually implementing this business process, we want to introduce you to the Land of Flow: Flow Builder.

13.5.1 Flow Builder

The Flow Builder User Interface

When you build flows, you work from Flow Builder. Here are its parts and pieces.



Toolbox (1)

The toolbox contains the elements and resources you use to build your flow.

- From the Elements tab, add new elements, like Screen and Create Records, to your flow.
- From the Manager tab, create resources, such as variables, stages, and choices, to use in your flow. Or view a list of all elements and resources that you've added to the flow.

Canvas (2)

The canvas is the working area, where you build a flow by adding elements. As you add elements to the canvas and connect them together, you see a visual diagram of your flow.

Button Bar (3)

The button bar provides information about the flow, such as:

- Whether the flow is active or not.
- How long ago the flow was saved.
- Whether the flow has any warnings or errors. To see warnings or errors, click the respective icon.

The button bar also includes two buttons for running a flow: Run and Debug.

- **Run** runs the most recent saved version of the flow that you have open.
- **Debug** lets you enter values for the flow's input variables and display debug details while running the flow. That way, you can verify how the flow processes data.

You'll learn all about testing and debugging flows in the Flow Testing and Distribution module.

Flow Building Blocks

Every flow is made up of three building blocks: elements, connectors, and resources.

- *Elements* (1) appear on the canvas. To add an element to the canvas, click and drag it there from the toolbox.
- *Connectors* (2) define the path that the flow takes at runtime. They tell the flow which element to execute next. Click and drag connectors from one element on the canvas to another.
- *Resources* (3) are containers that represent a given value, such as field values or formulas. You can reference resources throughout your flow. For example, look up an account's ID, store that ID in a variable, and later reference that ID to update the account.

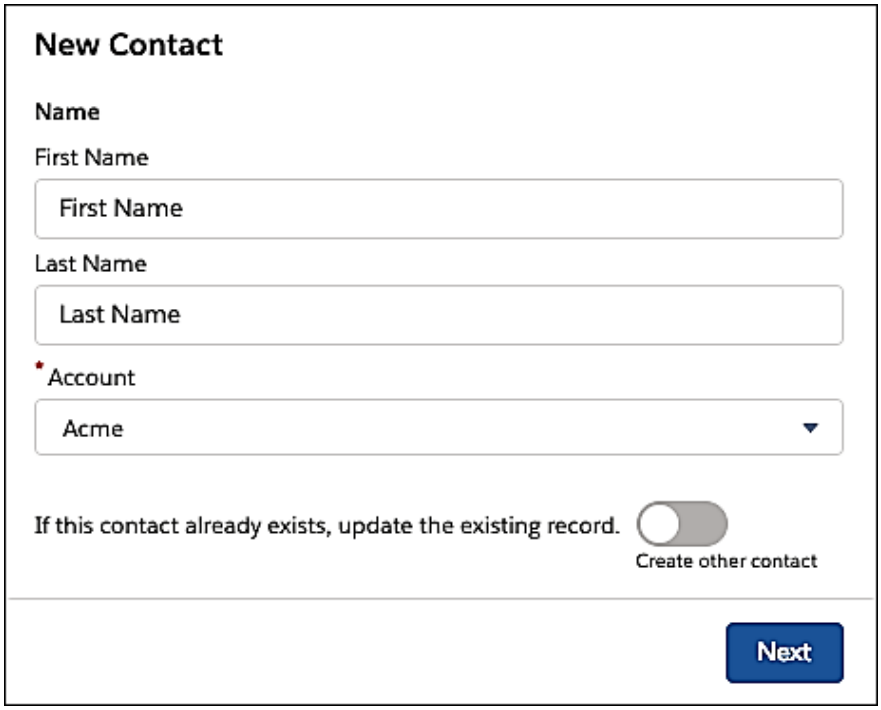
Elements (1)

Each element is a step in the flow that instructs the flow on what to do. Those instructions vary based on which kind of element it is. We like to think of flow elements in terms of three different buckets: screens, logic, and actions.

Screens

Display data to your users or build a form to collect information from them with Screen elements.

Screens includes a treasure trove of out-of-the-box components that let you customize how users interact with your flows. Display text or images to your users. Request free-form responses, such as with a text box, or ask users to select from provided choices, such as with radio buttons. You can even have the user upload a file through the screen.



If you need more from your flow screens, like custom navigation or a data table, install third-party components or work with a developer to build custom ones.

Logic

Once you collect the data you need, what do you want to do with it? With logic elements, evaluate that data and manipulate it according to your business requirements. Create branches, update data, loop over sets of data, or pause until a specified time.

Logic operates only in the flow. If you use logic elements to evaluate or manipulate information in your flow, the resulting data can't be accessed after the flow finishes. Use an action to store that data somewhere outside of the flow.

Actions

Actions instruct the flow to do something outside of itself, such as in the Salesforce database or in an external system. Flows can look up, create, update, and delete Salesforce records. They can also create Chatter posts, submit records for approval, and send emails.

Connectors (2)

Connectors define the path that the flow takes as it runs. They tell the flow which element to execute next.

Resources (3)

Resources are placeholders that you reference throughout your flow. For example, look up an account's ID, store that ID in a variable, and later reference that ID to update the account.

Plan Out the Flow

Now that we've learned a bit about the types of elements that flow offers, let's think back to the request. It's tempting to jump right into Flow Builder, but first map those requirements into a plan.

Note

No matter which tool you use, it's important to plan out your business process before you try to automate it.

Requirement	Element Type to Use
Collect information from user: first name, last name, and account for contact, as well as what to do if a matching contact exists.	Screen
Find a matching contact record.	Action
Check if a matching record was found.	Logic
If no match exists, create the contact.	Action
If a match exists, update that contact.	Action
For both branches, confirm what the flow did in Chatter.	Action
For both branches, confirm that the flow is done.	Screen

To build a flow that addresses these requirements, complete the remaining projects and modules in the Build Flows with Flow Builder trail. The Build a Simple Flow project guides you through building most parts of the flow, and we use the project as a basis to dig deeper on flow concepts in the Flow Builder module.

13.5.2 Build a Simple Flow

Collect Contact Info from Your User

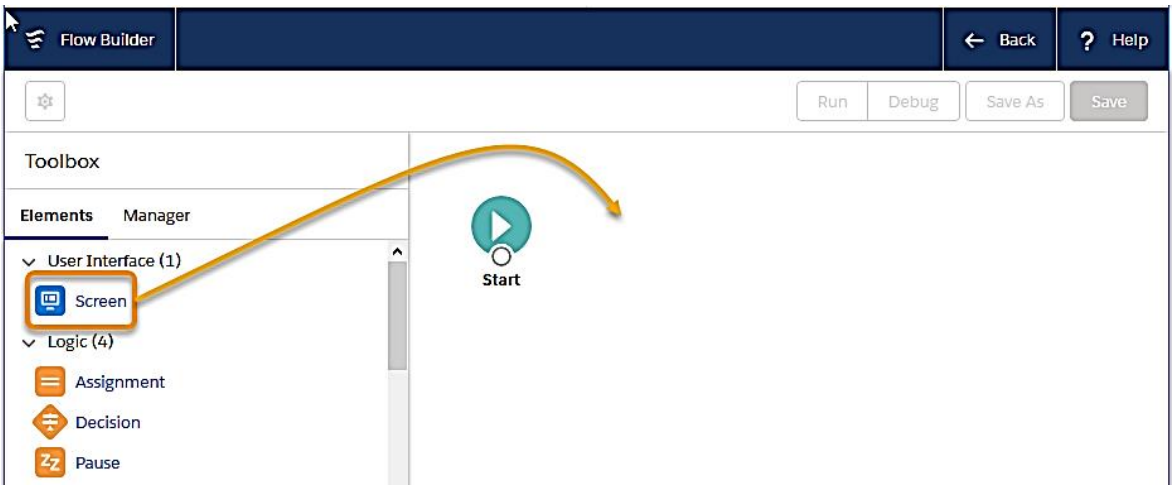
Introduction

This project walks you through the motions of building a simple flow. As you complete this project, don't worry about understanding everything. Follow the instructions and observe how the various controls work in Flow Builder. By the end of this project, you'll have a working flow. Later, you add to

this flow as you complete the Flow Builder module. As you complete that module, you learn all about the flow building blocks that you use in this project. For now, gain some muscle memory as you repeat the flow-building pattern of drag or click, configure, and save. Let's get started!

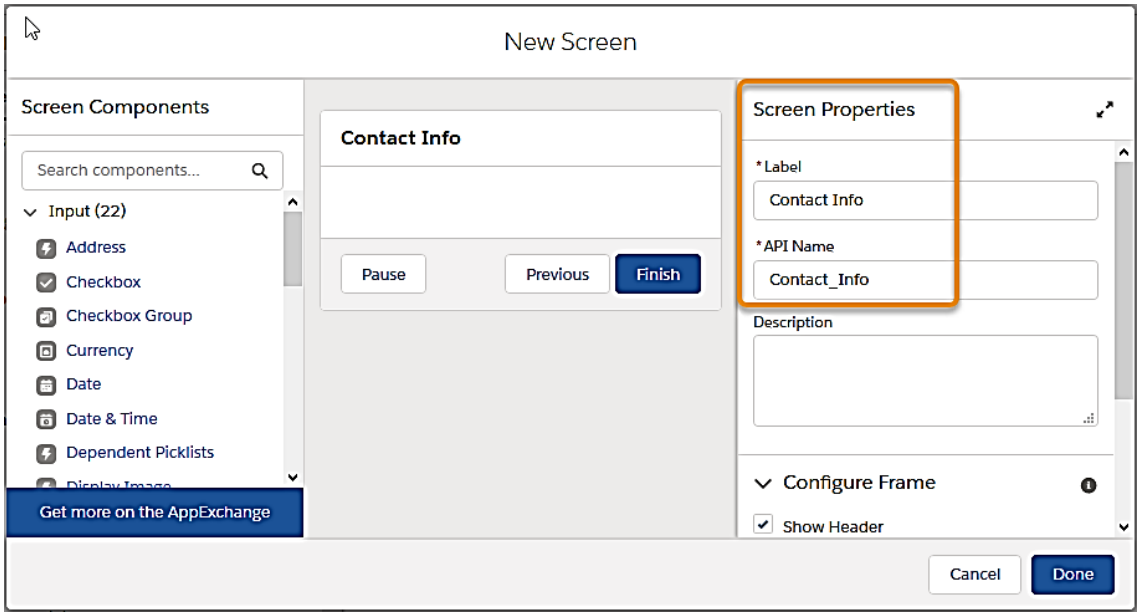
Create the Flow and Add a Screen

1. From Setup, enter Flows in the Quick Find box, then select **Flows**.
2. Click **New Flow**.
   Throughout this project, if you see the Goodbye, Cloud Flow Designer. Hello, Flow Builder! window, click **Start Building Flows**.
3. From the Elements tab in the toolbox, drag Screen onto the canvas.



New screen opens.

4. For Label, enter Contact Info.
5. The API Name is automatically set to Contact_Info.

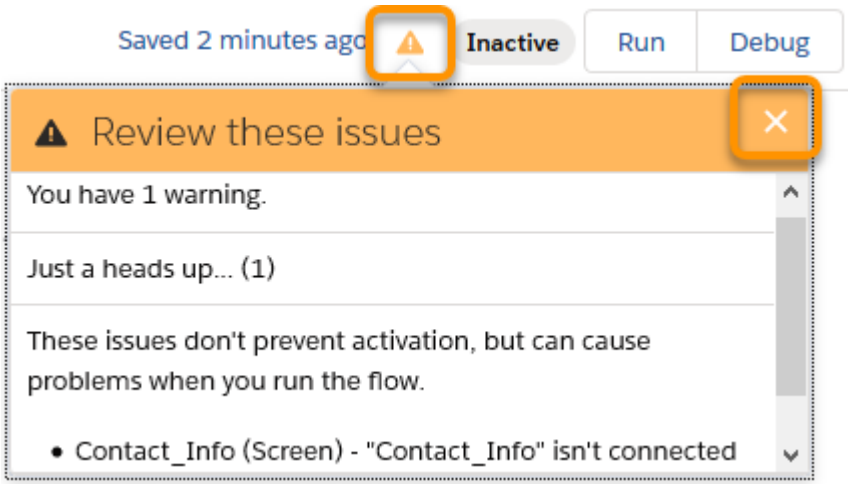


6. Click **Done**.
   The canvas now includes two items: Start and Contact Info.

7. Drag the connector node at the bottom of Start onto Contact Info.
   The two canvas items are now connected.
8. Click **Save**, and enter these values.

Field	Value
Flow Label	New Contact
Flow API Name	New_Contact
Type	Screen Flow

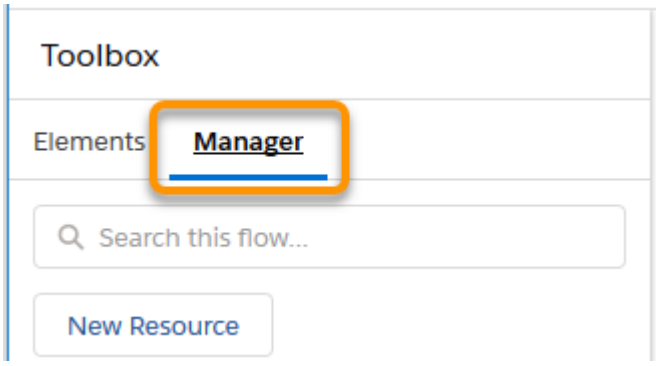
9. Click **Save**.
10. Throughout this project, ignore and close the warnings that appear when you save the flow. You can always read warnings later by clicking.



Create a Record Variable for the Contact

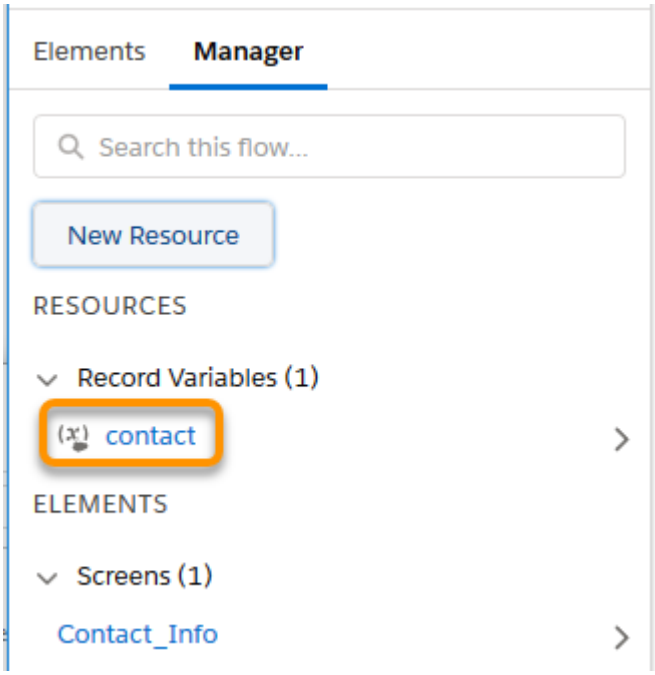
Because you plan to capture contact information from the user, let's create a record variable to store that contact record data in the flow.

1. In the Toolbox, click the **Manager** tab.



2. Click **New Resource**.
3. For Resource Type, select **Variable**.
4. For API Name, enter contact.
5. For Data Type, select **Record**.

6. For Object, enter **Contact**.
7. Click **Done**.
   In the toolbox, the Manager tab now contains a contact record variable resource.

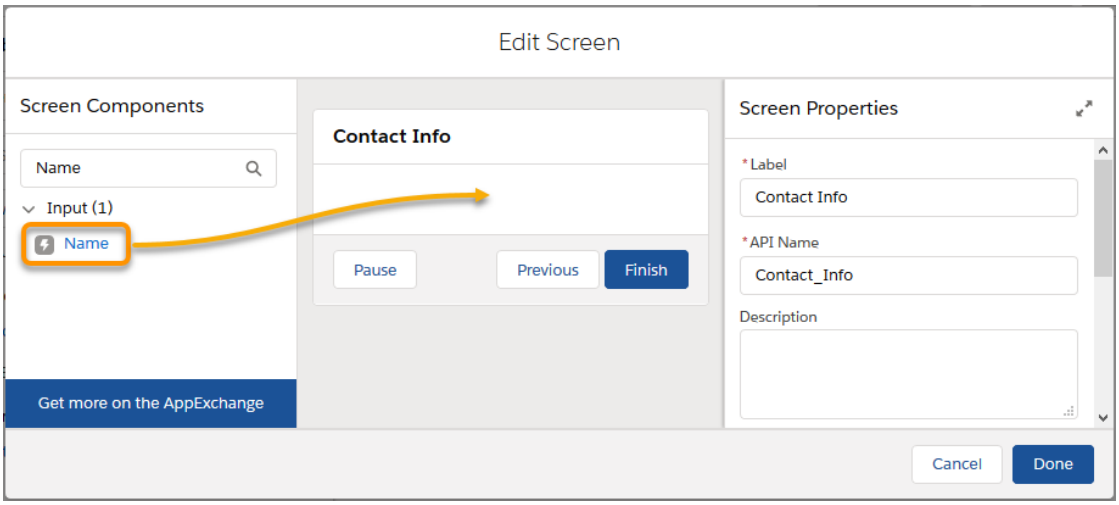


8. Click **Save**.

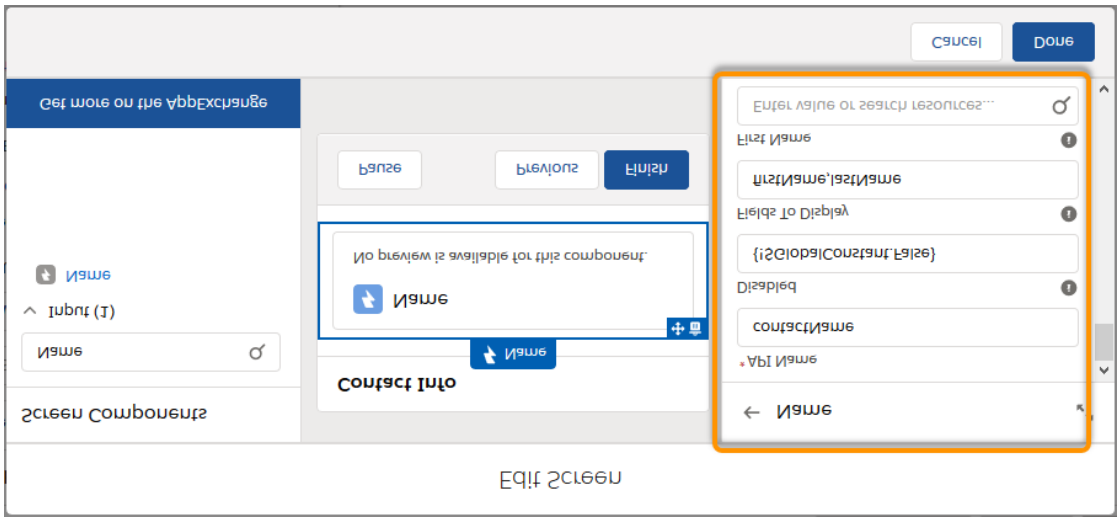
Add a Name Input Component to the Screen

You need to create form fields for the user to enter contact information on the screen. Let's start with an input component that collects the contact's name.

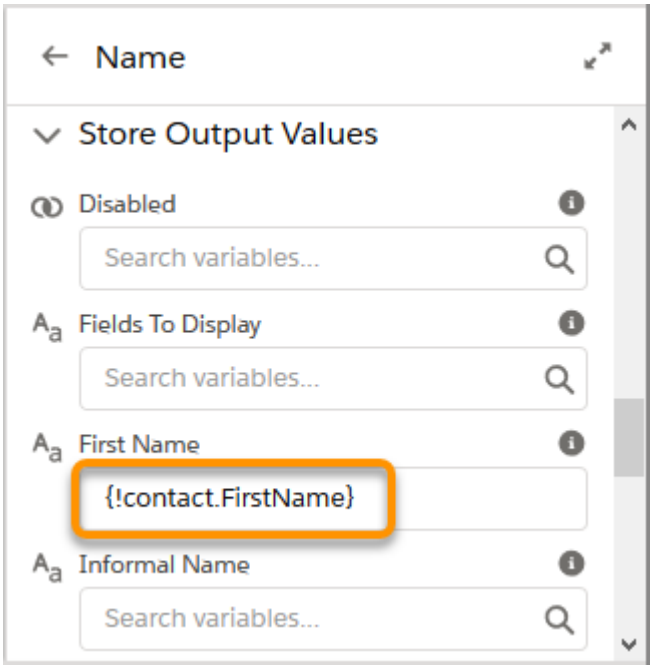
1. On the canvas, double-click the **Contact Info** screen. The Edit Screen window opens.
2. In the Screen Components pane, enter Name in the search box.
3. Drag Name onto the screen canvas.



The properties pane now represents the Name input component.



4. In the properties pane, for API Name, enter contactName.
5. Scroll or tab down the properties pane to the Store Output Values section.
6. Complete these steps to take what the user enters in the First Name and Last Name fields and store those values in the contact record variable.
    a. For First Name, select **contact** and then **FirstName**. The value is displayed as {!contact.FirstName}.



    b. For Last Name, enter {!contact.LastName}.
7. Click **Done**.
8. Click **Save**.

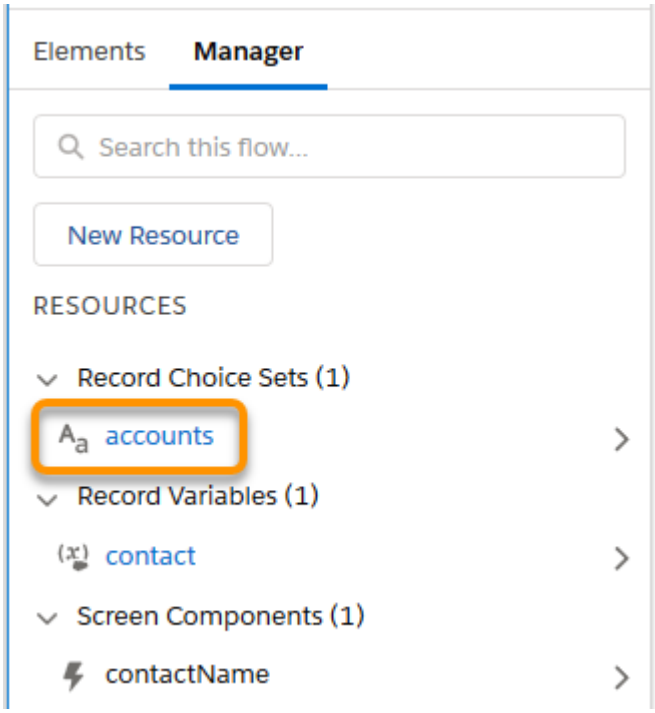
Field	Value
Resource Type	Record Choice Set

API Name	Accounts
Object	Account
Choice Label	Name
Data Type	Text
Choice Value	Id

Create a Record Choice Set for Accounts

You want the user to specify the parent account for the contact. Let's create a record choice set for use later as a picklist on the screen.

1. From the Manager tab in the toolbox, click **New Resource**.
2. In the New Resource window, enter these values.
3. In the Store More Account Field Values section, complete these steps to take the user-selected account and store its ID in the contact record variable.
    a. For Field, select **Id**
    b. For Variable, enter {!contact.AccountId}.
4. Click **Done**.

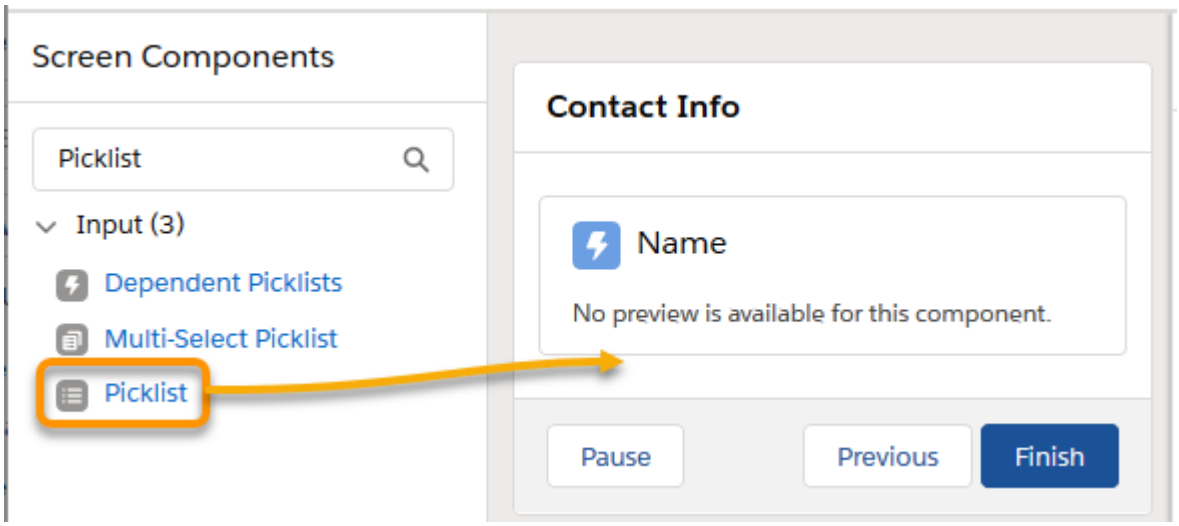


5. Click **Save**.

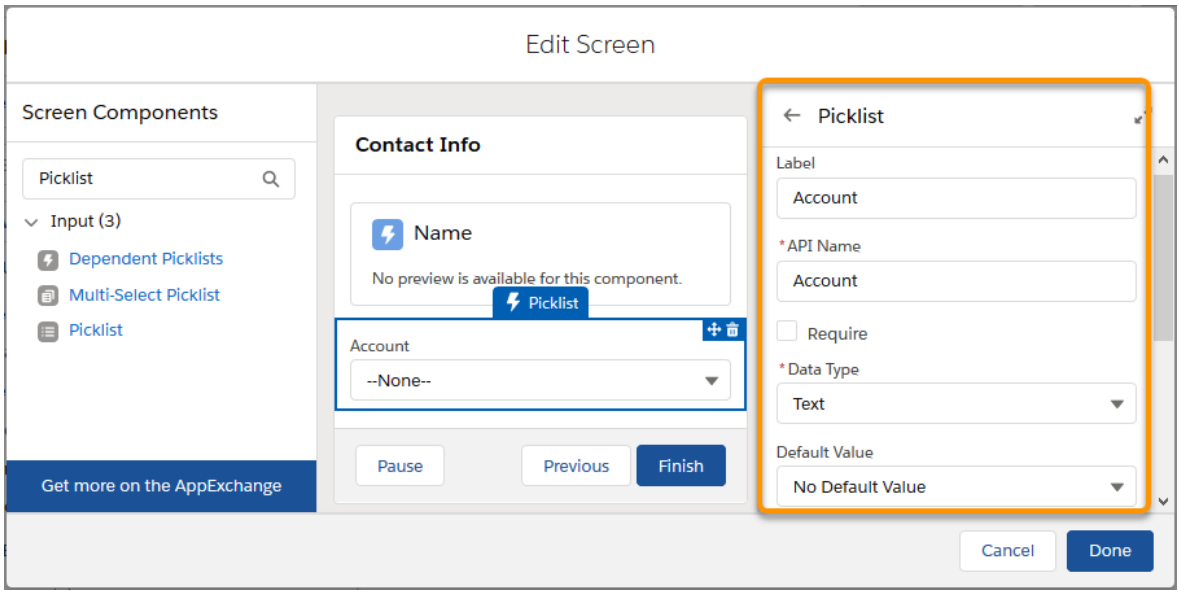
Add a Picklist Input Component to the Screen

Now let's add the picklist for the user to select the contact's parent account.

1. On the canvas, double-click the **Contact Info** screen.
2. In the Screen Components pane, enter Picklist in the search box.
3. Drag Picklist onto the screen canvas just below the Name component.



The properties pane now represents the Picklist input component.



4. In the properties pane, enter these values.

Field	Value
Label	Account
API Name	Account
Data Type	Text

5. In the Select Choices section, for Choice, enter {!accounts}.
6. Click **Done**.
7. Click **Save**.

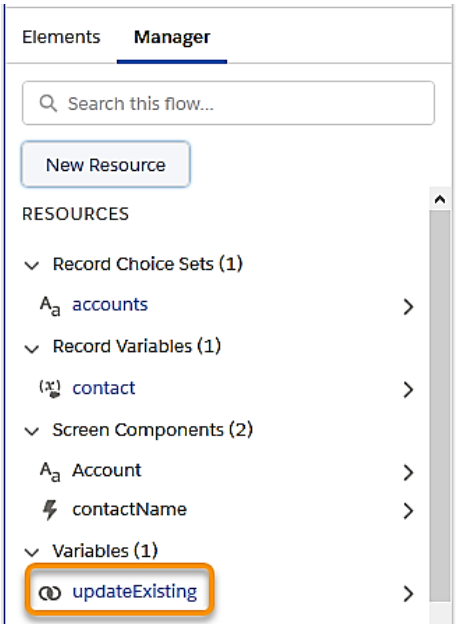
Create a Boolean Variable for Whether to Update a Contact

Because the user chooses whether or not to update an existing contact if one is found, create a Boolean variable to store the user's decision.

1. From the Manager tab in the toolbox, click **New Resource**.
2. In the New Resource window, enter these values.

Field	Value
Resource Type	Variable
API Name	updateExisting
Data Type	Boolean

3. Click **Done**. In the toolbox, the Manager tab now contains the updated Existing variable.

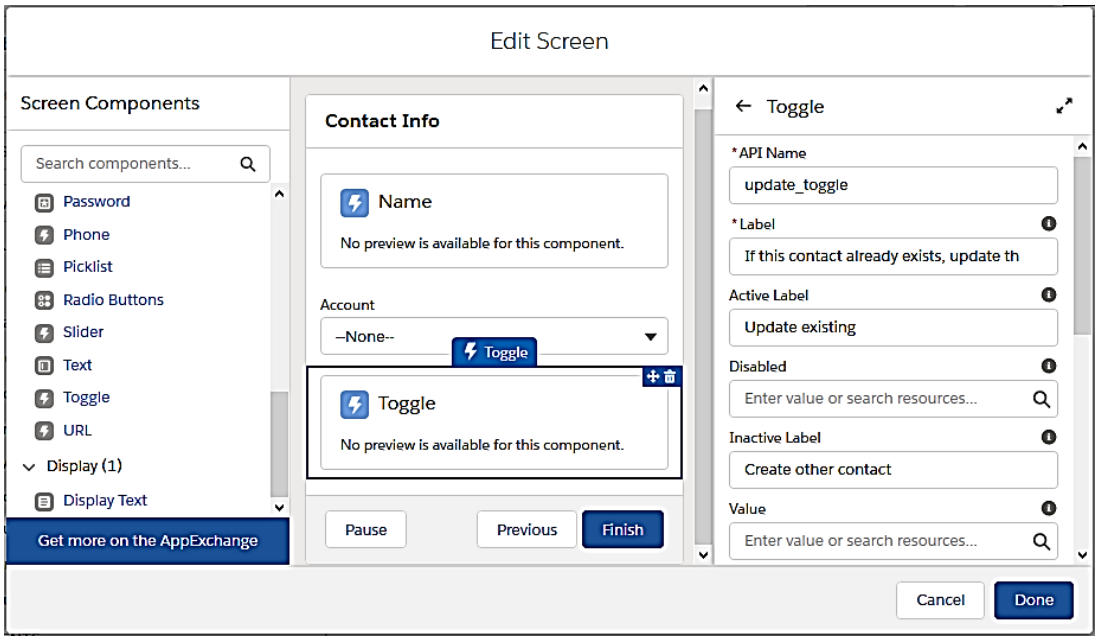


4. Click **Save**.

Add a Toggle Input Component to the Screen

Now let's add the toggle for the user to specify whether (or not) to update an existing contact if one is found.

1. On the canvas, double-click the **Contact Info** screen.
2. From the Screen Components pane, drag Toggle onto the screen canvas just below the Account picklist component.



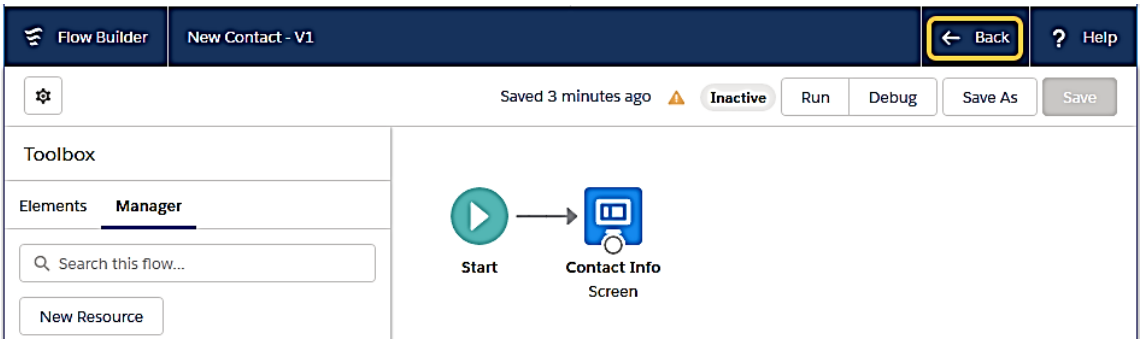
3. In the properties pane, enter these values.

Field	Value
API Name	update_toggle
Label	If this contact already exists, update the existing record.
Active Label	Update existing
Inactive Label	Create other contact

4. In the Store Output Values section, for Value, enter {!updateExisting}.
5. Click **Done**.
6. Click **Save**.

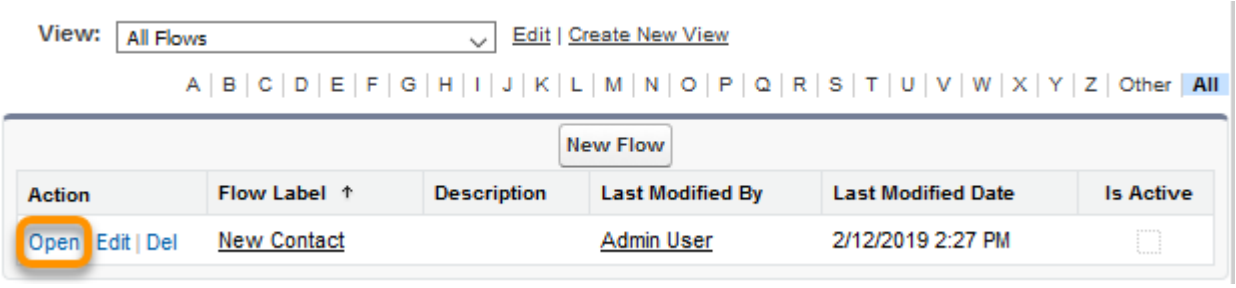
Close and Open the Flow

1. To exit Flow Builder, click Back in the top right corner.



The flow list page appears.

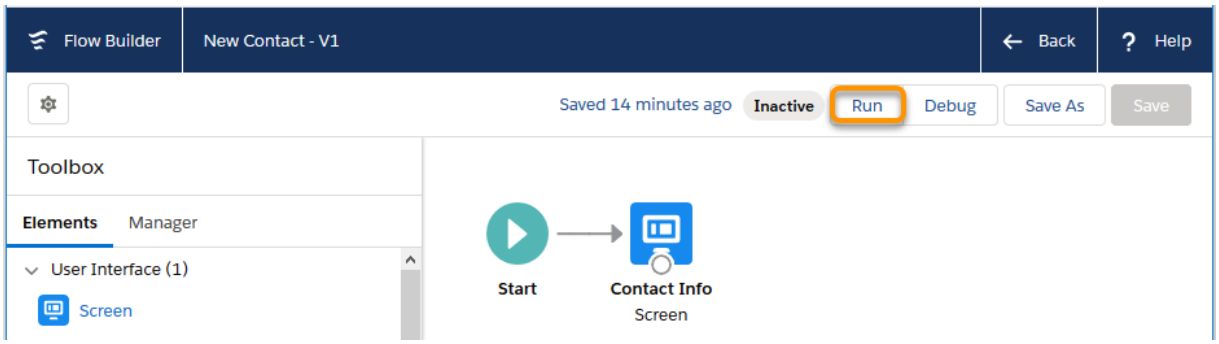
2. To reopen the flow in Flow Builder, find New Contact on the flow list page, and click **Open**.



Run the Flow to View the Screen

At this point, your flow can collect contact information from the user and store that information in flow variables. Let's see what the screen looks like to users who run the flow.

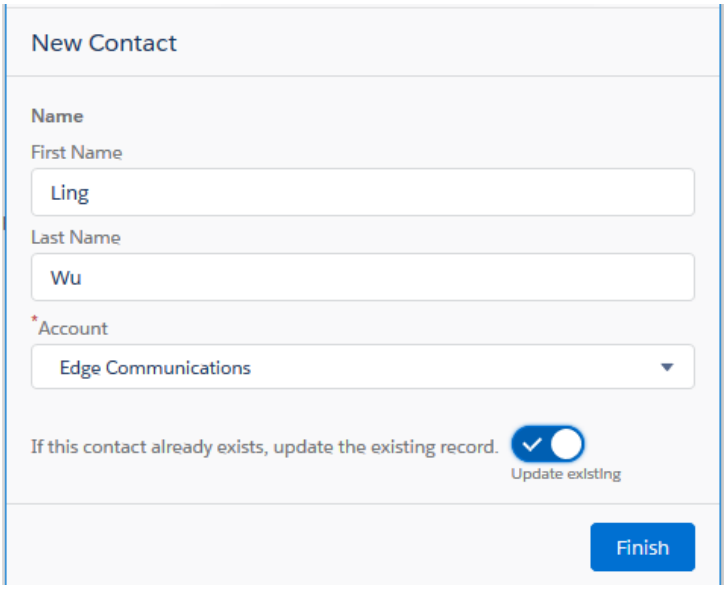
1. Above the canvas, click **Run**.



Note

If you see an error message that says we can't display a component because it isn't supported in Classic runtime that means you're using an older Salesforce org. Not to worry! A quick settings change is all you need. From Setup, enter Process Automation Settings in the Quick Find box, then select **Process Automation Settings**. Select **Enable Lightning runtime for flows**, and click **Save**. Now reopen your flow, and click **Run**.

2. Play with the screen components and click **Finish**.



Because the flow has no elements to execute after the screen, nothing happens when the flow finishes. The flow simply runs again, which means that you see the screen again.
Congrats on building and even running a flow! So far, the flow only displays a screen that accepts user input. Next, you configure the flow to compare some of that user input with data in your org.

Check for a Matching Contact in Your Org

Create a Variable for the Contact Record ID

Now that your flow prompts the user to enter information about a contact, the next step is to look for a matching contact record in your org.
If a contact record in the org matches the contact name that the flow user enters, you want a variable to store the existing contact's record ID in the flow.

1. Open the New Contact flow in Flow Builder.
2. From the Manager tab in the toolbox, click **New Resource**.
3. In the New Resource window, enter these values.

Field	Value
Resource Type	Variable
API Name	existingId
Data Type	Text

4. Click **Done**.
5. Click **Save**.

Use a Get Records Element to Find a Matching Contact

Add a Get Records element to your flow to query your org for any existing contact whose name matches what the flow user enters.

1. From the Elements tab in the toolbox, drag Get Records onto the canvas.
2. In the New Get Records window, enter these values.

Field	Value
Label	Find a Match
API Name	Find a Match
Object	Contact

3. In the Filter Contact Records section, for Condition Requirements, select **Conditions are Met**.
4. Click **Add Condition**, and enter these conditions.

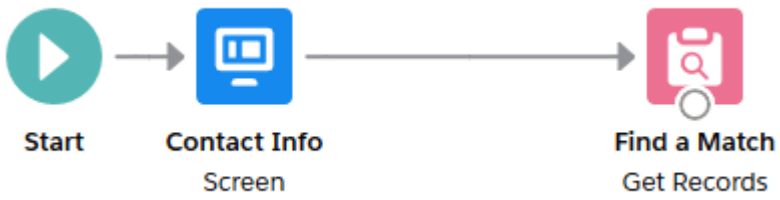
Field	Operator	Value
FirstName	Equals	{!contact.FirstName}
LastName	Equals	{!contact.LastName}

5. For Where to Store Field Values, select **In separate variables**.

6. Select **When no records are returned, set specified variables to null**.
7. In the Select Variables to Store Contact Fields section, enter this variable assignment.

Field	Value
Id	{!existingId}

8. Click **Done**.
9. Drag the connector node from Contact Info onto Find a Match.



10. Click **Save**.

13.5.3 Branch the Flow

Add a Decision Element to the Flow

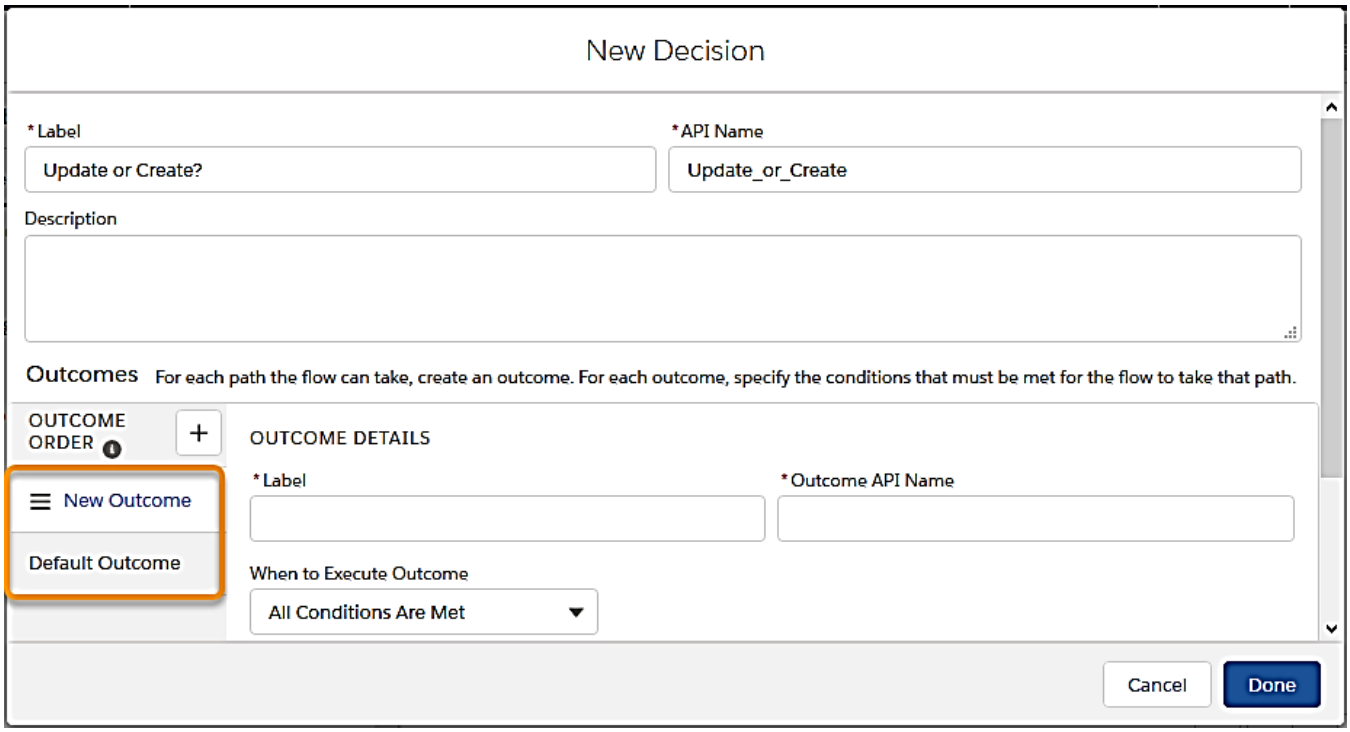
Now that your flow checks for a matching contact record in your org, let's branch the flow. The path the flow takes depends on whether a matching contact is found, and whether the user chooses to update the matching contact or create a new contact.

Add a Decision element to create two branches in the flow: one that creates a contact record, and one that updates any existing record that's found.

1. From the Elements tab in the toolbox, drag Decision onto the canvas.
2. In the New Decision window, enter these values.

Field	Value
Label	Update or Create?
API Name	Update_or_Create

3. In the Outcomes section, notice the two listed outcomes: New Outcome and Default Outcome.



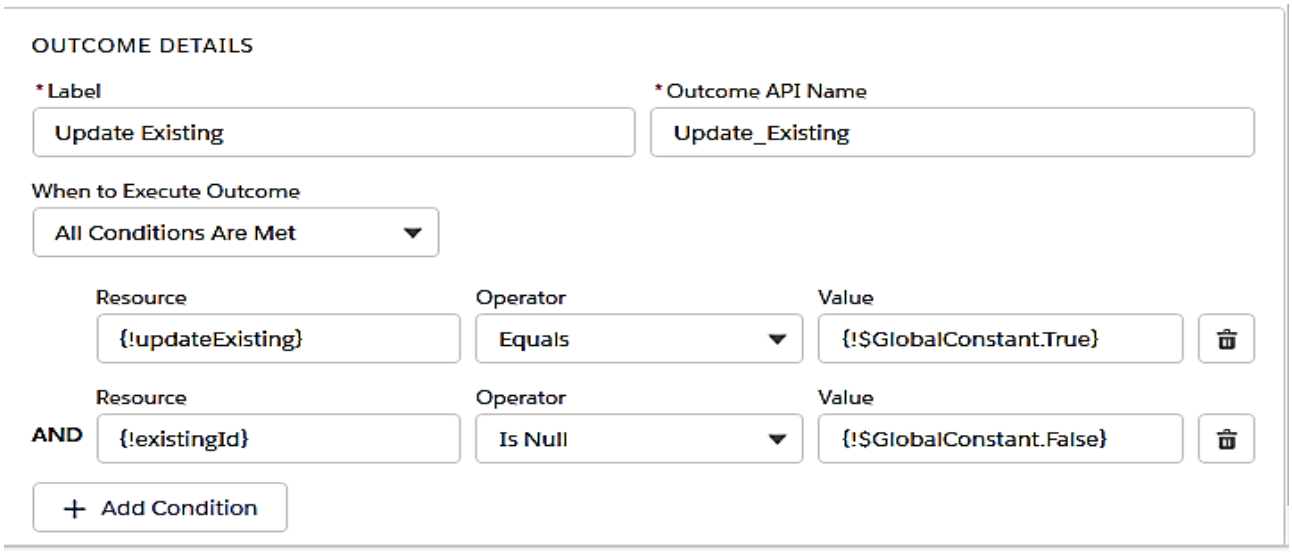
4. Make sure that New Outcome is selected, and enter these values into the Outcome Details section.

Field	Value
Label	Update Existing
API Name	Update_Existing

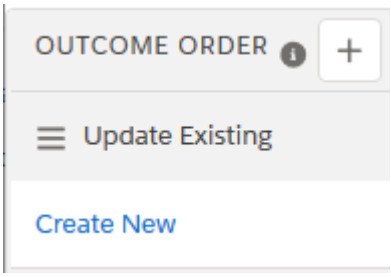
5. For When to Execute Outcome, select **All Conditions are Met**.

6. Click **Add Condition** and then enter these conditions.

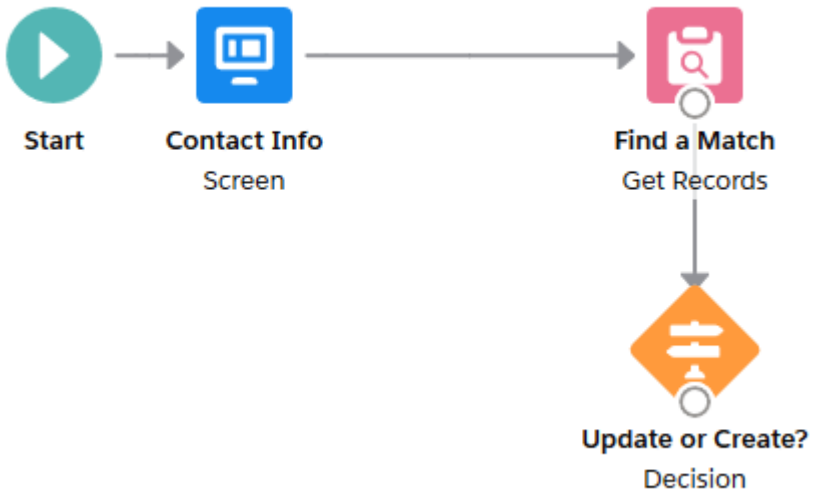
Resource	Operator	Value
{!updateExisting}	Equals	{!$GlobalConstant.True}
{!existingId}	Is Null	{!$GlobalConstant.False}



7. In the list of outcomes in the sidebar, select **Default Outcome**.
8. In the Outcome Details section, for Label, enter Create New.
   Notice that the list of outcomes now displays the outcome labels that you entered.



9. Click **Done**.
10. Drag the connector node from Find a Match onto Update or Create?.



11. Click **Save**.

Now you've equipped your flow with the logic to take one of two paths. Next, you build out those two paths.

Create or Update a Contact

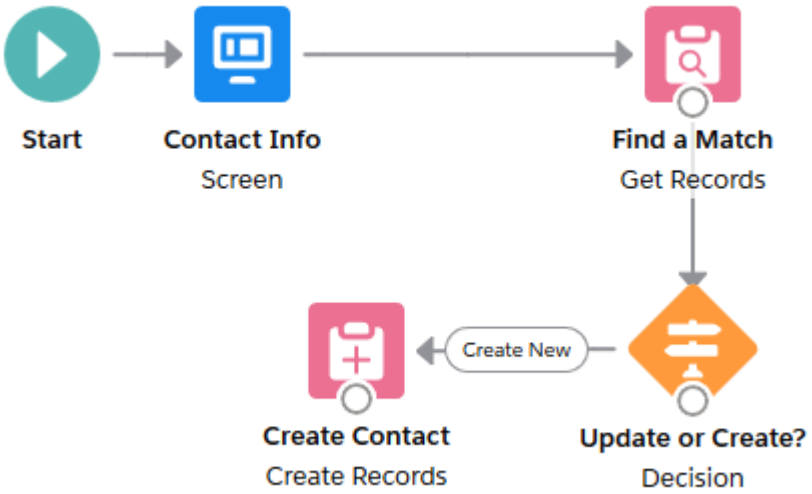
Use a Create Records Element to Create a New Contact

Now that your flow has the logic to decide between two paths, let's build those two paths. First, build the path for the default Create New outcome, which contains only one element.

1. From the Elements tab in the toolbox, drag Create Records onto the canvas.
2. In the New Create Records window, enter these values.

Field	Value
Label	Create Contact
API Name	Create_Contact
Record Variable	{!contact}

3. Click **Done**.
4. Drag the connector node from Update or Create? onto Create Contact. Because the Decision element has multiple outcomes, you're prompted to choose the outcome to associate with the connector.
5. For Outcome, select **Create new**, and click **Done**.



6. Click **Save**.

Now the flow can create a contact record when the Decision element executes and takes the Create New outcome's path.

Use an Assignment Element to Set the Contact ID in the Record Variable

Let's build the path for the Update Existing outcome. First, you need to get the record ID of the existing contact into the contact record variable.

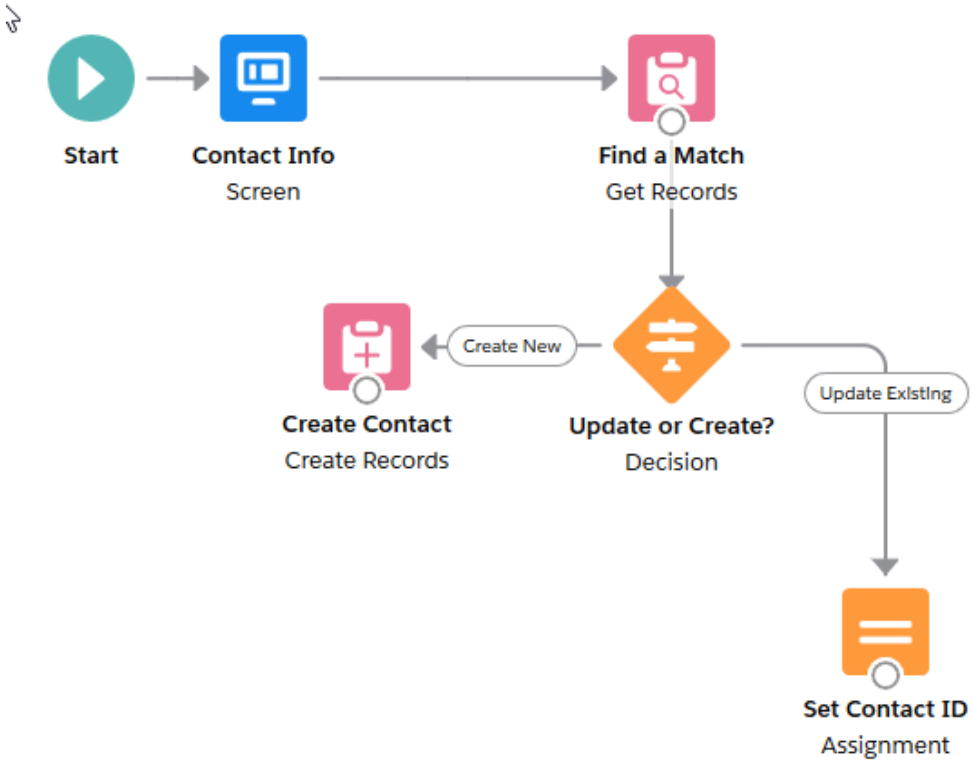
1. From the Elements tab in the toolbox, drag Assignment onto the canvas.
2. In the New Assignment window, enter these values.

Field	Value
Label	Set Contact ID
API Name	Set_Contact_ID

3. In the Set Variable Values section, enter these values.

Variable	Operator	Value
{!contact.Id}	Equals	{!existingId}

4. Click **Done**.
5. Drag the connector node from Update or Create? Onto Set Contact ID. Because the Decision element has only one unused outcome at this point, the connector is automatically associated with the Update Existing outcome.



6. Click **Save**.

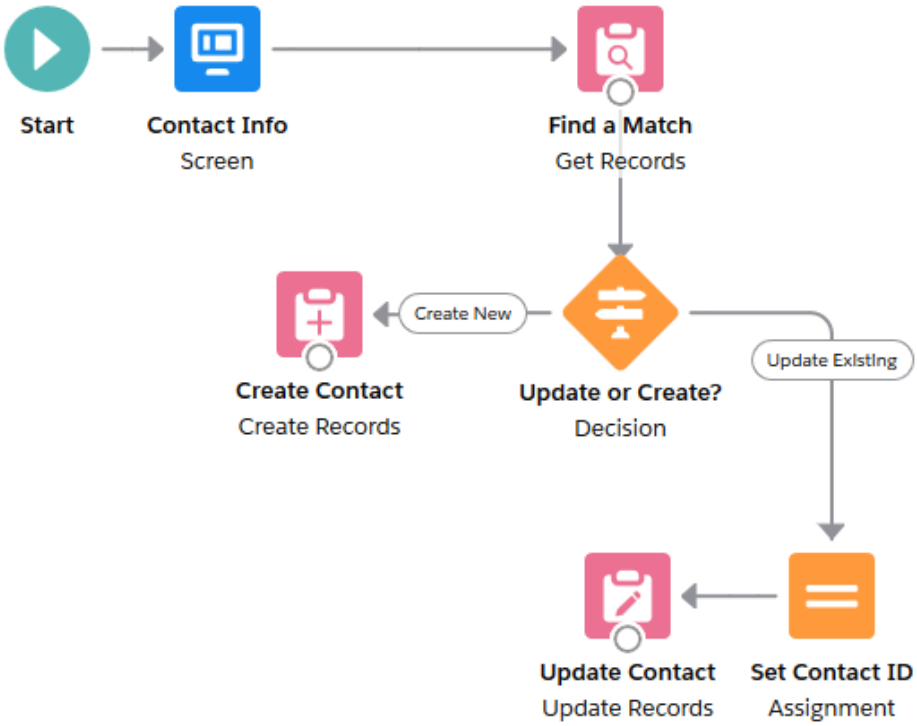
Use an Update Records Element to Update an Existing Contact

Now that the contact record variable contains the record ID of the matching contact, you can reference that contact record variable in an Update Records element.

1. From the Elements tab in the toolbox, drag Update Records onto the canvas.
2. In the New Update Records window, enter these values.

Field	Value
Label	Update Contact
API Name	Update_Contact
Record Variable or Record Collection Variable	{!contact}

3. Click **Done**.
4. Drag the connector node from Set Contact ID onto Update Contact.



5. Click **Save**.

You're all done building the flow! You can run the flow and confirm that contacts in your org are created or updated. If you're feeling adventurous, click Debug and then Run to see the details of what your flow does as it runs.

13.5.4 Flow Distribution

Make Sure Your Flow Works

As an admin or developer, you know you should test all customizations before rolling them out to users, and flows are no different. Testing lets you fine-tune the flow's behaviour, identify and fix bugs, and otherwise make sure your users have a successful experience. And of course, you also benefit, because you're much less likely to spend time later responding to panicked emails from your users.

Before You Start

Before you can complete this module, make sure that you complete the Flow Builder module. The work you do here builds on the concepts and work you complete in those modules.

Create a Test Plan

Before you start testing, draft a list of test cases, and identify what you expect the result to be. Consider things like:

- When you expect actions to occur
- When you expect actions to not occur
- How formulas should resolve

For the "New Contact" flow you've been working on, there are four major test cases.

Toggle setting	Matching record	Expected result
Deselected	Doesn't exist	• A contact is created • Confirmation screen includes a working link to the record • A Chatter post was created on the record's feed • Confirmation screen and Chatter post specify "created"
Deselected	Exists	• A contact is created • Confirmation screen includes a working link to the record • A Chatter post was created on the record's feed • Confirmation screen and Chatter post specify "created"
Selected	Doesn't exist	• A contact is created • Confirmation screen includes a working link to the record • A Chatter post was created on the record's feed • Confirmation screen and Chatter post specify "created"
Selected	Exists	• A contact is updated • Confirmation screen includes a working link to the record • A Chatter post was created on the record's feed • Confirmation screen and Chatter post specify "updated"

Now that you know what to test for, let's see how to test the flow.

Testing Options in Flow Builder

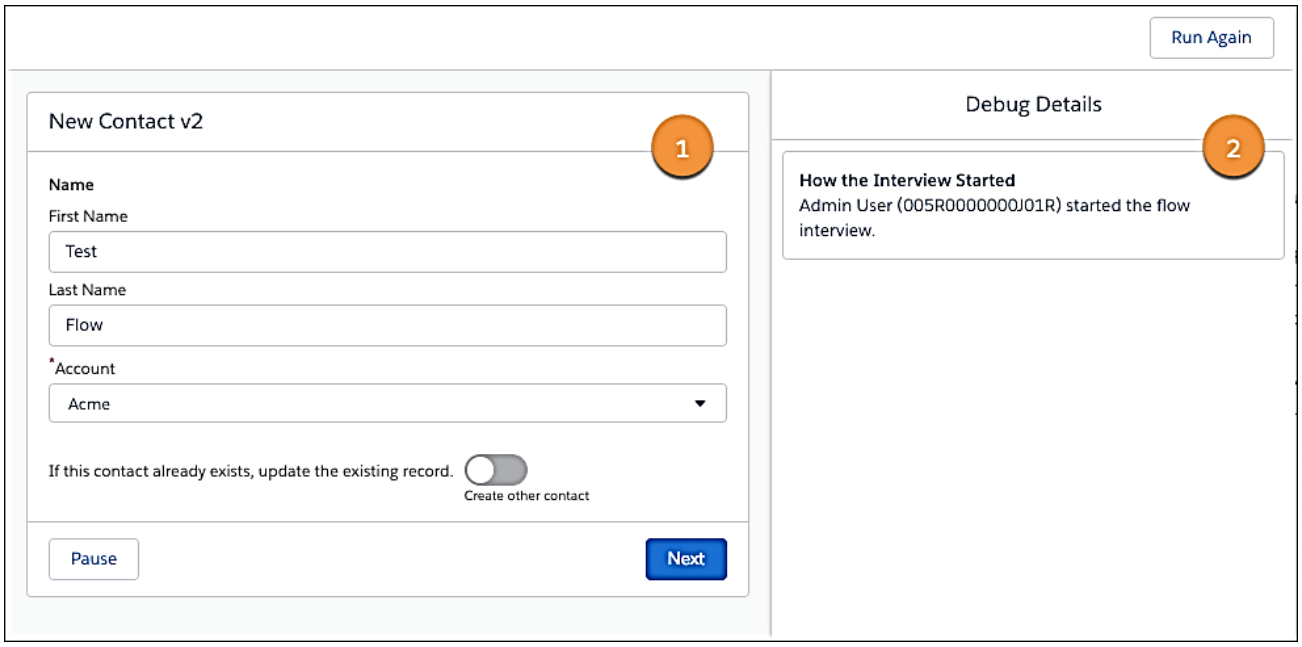
You don't have to leave Flow Builder to make sure your flow works. The button bar includes two buttons for running a flow: Run and Debug.

- **Run** runs the most recent saved version of the flow that you have open.
- **Debug** does everything that Run does, but with some superpowers thrown in. It lets you enter values for the flow's input variables and display debug details while running the flow. That way, you can verify how the flow processes data.

Tip

Unless you need to test how your flow works in Classic runtime, always use Debug to test your flows. While Debug always uses Lightning runtime, Run obeys the Enable Lightning runtime preference in your org's Process Automation settings.

When you click Debug and opt to show details, you see the flow's first screen (1) and the debug details (2). As you step through the flow, new details are added to the right-hand panel.



Introducing Flow Interviews

Each time a flow runs, a flow interview starts. A flow interview is an instance of a flow.
Think of Choose-Your-Own-Adventure books. A flow is like the book itself, which provides choices to the reader and instructions for each choice. A flow interview is like the reader. As you read, you make choices and follow the instructions for those choices. Each time you or another person reads the book, you can take a different path through the book and experience a different story.
The same goes for interviews. Based on the data provided for that interview, either by input variables or input components on a screen, each interview can take a different path through the flow and result in different actions being performed.
To see interviews in action, let's verify the four cases from our test plan.

Test Your Flow from Flow Builder

1. From Flow Builder, click **Debug**. Make sure that the second checkbox is selected. Otherwise, you won't see any debug details. Since this flow doesn't have Subflow elements or input variables, don't worry about those settings.
2. Click **Run**.
3. Validate the first test case.
    a. Enter a first and last name, and choose an account.
    b. Leave the toggle unselected.
    c. Click **Next**.
    d. Review the debug details.
        - The first card identifies who started the flow interview. Since you started it, you should see your name and user ID.
        - The second card summarizes how the inputs from the first screen were stored for the flow interview to use. For example, because you left the toggle unselected, the {! update Existing} variable was set to false

SCREEN: Contact_Info
Lightning Component: contactName
Lightning Component: flowruntime:name
Inputs:
Outputs:
firstName = contact.FirstName (Test)
lastName = contact.LastName (Flow)

Dropdown List: Account
Label: Account
Data Type: Text
Choices selected at runtime:
Choice selected at runtime: Acme (accounts)
Value of choice: 001B000000ukV5kIAE
Field assignments:
{!contact.AccountId} = Id (001B000000ukV5kIAE)

Lightning Component: update_toggle
Lightning Component: flowruntime:toggle
Inputs:
label = (If this contact already exists, update the existing record.)
messageToggleActive = (Update existing)
messageToggleInactive = (Create other contact)
Outputs:
value = updateExisting (false)

Selected Navigation Button: NEXT

- The third card summarizes how the “Update If Existing?” decision was evaluated. Because {!updateExisting} was false, the interview took the default path. On the canvas, that’s the connector labeled “No”, which goes straight to the Create Records element.

DECISION: Update_If_Existing
Skipped this outcome because its conditions weren't met:
Update_Yes
Outcome conditions:
{!updateExisting} (false) Equals true

Executed the default outcome

- The fourth card summarizes the “Create Contact” element. The interview used the values in the {!contact} variable to create a contact record.

FAST CREATE: Create_Contact
Create Contact records using the values that are stored in {!contact}.
Variable Values
[FirstName=Test,LastName=Flow,AccountId=001B000000ukV5kIAE]
Result
One or more records are ready to be created when the next Screen or Wait element is executed or when the interview finishes.

- The fifth card summarizes the Post to Chatter action. The message posted was "This contact was created"

POST TO CHATTER: Post_to_Contact_s_Feed
Inputs:
text = {!chatterMessage} (The contact was created.)
subjectNameOrId = {!contact.Id} (003B000000Db9doIAB)
Outputs:
None.

e. Verify the expected results for this test case.
   - The confirmation screen specifies "created".
   - The link goes to a new contact. The contact's name matches what you entered and is a child of the account you selected.
   - On the contact's Chatter tab, a post specifies that this contact was "created".

4. Repeat for the other three test cases. For cases that include a matching record, use the same first name, last name, and account you used for the first test case.

If any of the test cases have unexpected results, use the Debug Details to backtrack and figure out what went wrong. Once all the test cases pass, you're ready to put the flow in your users' hands.

Enable User Access For Your Flow

Activate the Flow

Only flow admins (users with the Manage Flows permission) can run inactive flows.

1. From the button bar in Flow Builder, click **Back**.
   Flow Builder closes, and the flow list page opens.
2. Click **New Contact**.
3. For the version you saved, click **Activate**.

Grant Permissions to Flow Users

Now any user can run this flow, as long as they have either:

- The Run Flows user permission
- The Flow User field enabled on their user detail page

Make sure that all the appropriate users have permission to run flows.

1. Identify the users who will run the flow.
2. Create a permission set that includes the Run Flows permission.

3. Assign the permission set to the users you identified.

Distribute the Flow

Users can't run anything if they don't know where to find it. Put the flow somewhere useful so they can discover it. Flows can be distributed to users in a myriad of ways, depending on the type of flow. Here are the declarative options for distributing screen flows and autolaunched flows.

Type	Declarative Option
Screen Flow	• Flow actions • Lightning pages • Lightning community pages • Custom buttons or custom links • Web tabs • Direct flow URLs • Snap-in deployments
Autolaunched Flow	• Process Builder • Custom buttons or custom links • Web tabs

For a list of all the distribution options for each flow type, check out the resources at the end of this unit.

As you can see, screen flows support many distribution methods, but we recommend using Lightning pages, flow actions, the utility bar, or Lightning community pages. To choose between those options, here are some guiding questions.

If...	Use this option...
The flow should be available to external users	Lightning community page
The flow needs the context of a record	Lightning record page or flow action
The flow should be available from every page	Utility bar
The flow will be used every time someone accesses a page	Lightning page
The flow will be used occasionally	Flow action

Based on the business requirements, the New Contact flow:

- Should be available to sales reps, who are internal users (so no Lightning community pages)
- Doesn't require any context (so no flow actions or Lightning record pages)

If you're not using Lightning community pages or flow actions, here are your remaining options.

- Make the flow available on every page in an app with the **utility bar**.
- Make the flow available on the **Home page** of an app.
- Make the flow available on a standalone **App page**.

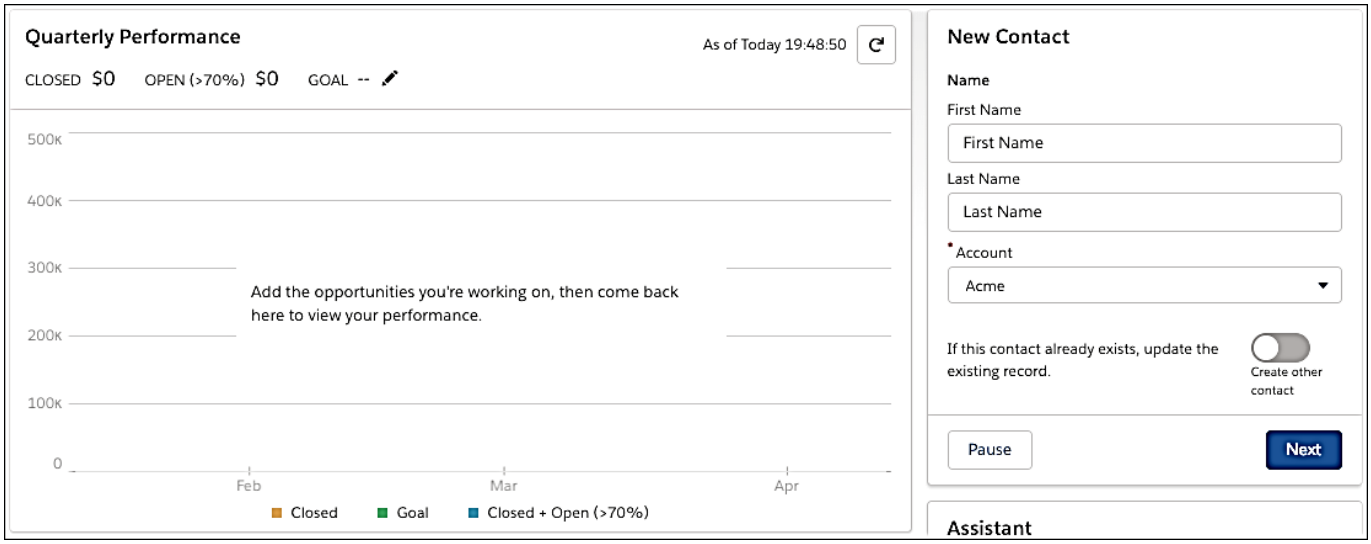
When offered those three choices, your stakeholder chooses to put the flow on the Home page of the Lightning Sales app.

Add the Flow to the Home Page

1. From Setup, enter builder in the Quick Find box, and then select **Lightning App Builder**.
2. Open the appropriate home page, or create a new one.
    a. Click **New**.
    b. Select **Home Page**, and click **Next**.
    c. Give the page a name, and click **Next**.

   d. Click **Clone Salesforce Default Page**, select **Home Page Default**, and click **Finish**.

3. Drag a Flow component to the top of the right column.
4. For Flow, select **New Contact**.
5. Save your changes and activate the page. Make sure to mark this page as the default Home page.
6. To see your flow in action, go to your Home page.
   a. Click **Back** to return to Setup.
   b. Click the App Launcher icon and under All Items click **Home**.



14.0 Data Management

An overview of data import and export utilities

Salesforce provides data utilities that are available for the import and export of data to and from Salesforce. Looking at the available Salesforce supported facilities to import and export data, we have the following specific options:

14.1 Data Import Wizard

- **Import Accounts/Contacts**
- **Import Leads**
- **Import Solutions**
- **Import Custom Objects**

14.2 Data Loader

Data Import Wizard

The Data Import Wizard opens in a full browser window and provides a unified interface that lets you import data for a number of standard Salesforce objects, including **Account**, **Contact**, **Lead**, **Custom Object**, and **Solution.**

Data Loader

The Data Loader is a client application from Salesforce for bulk import or export of data. When importing data, the Data Loader loads data from CSV files or from a database connection, and the exporting of data is done using CSV files.

Data Loader and import wizards compared

Use the Data Loader when:

- You need to load 50,000 or more records where the maximum is 5 million records
- You need to load into an object that is not yet supported by web-based import wizards
- You want to schedule regular data loads, such as nightly imports
- You want to save mappings for later use
- You want to export your data for backup purposes

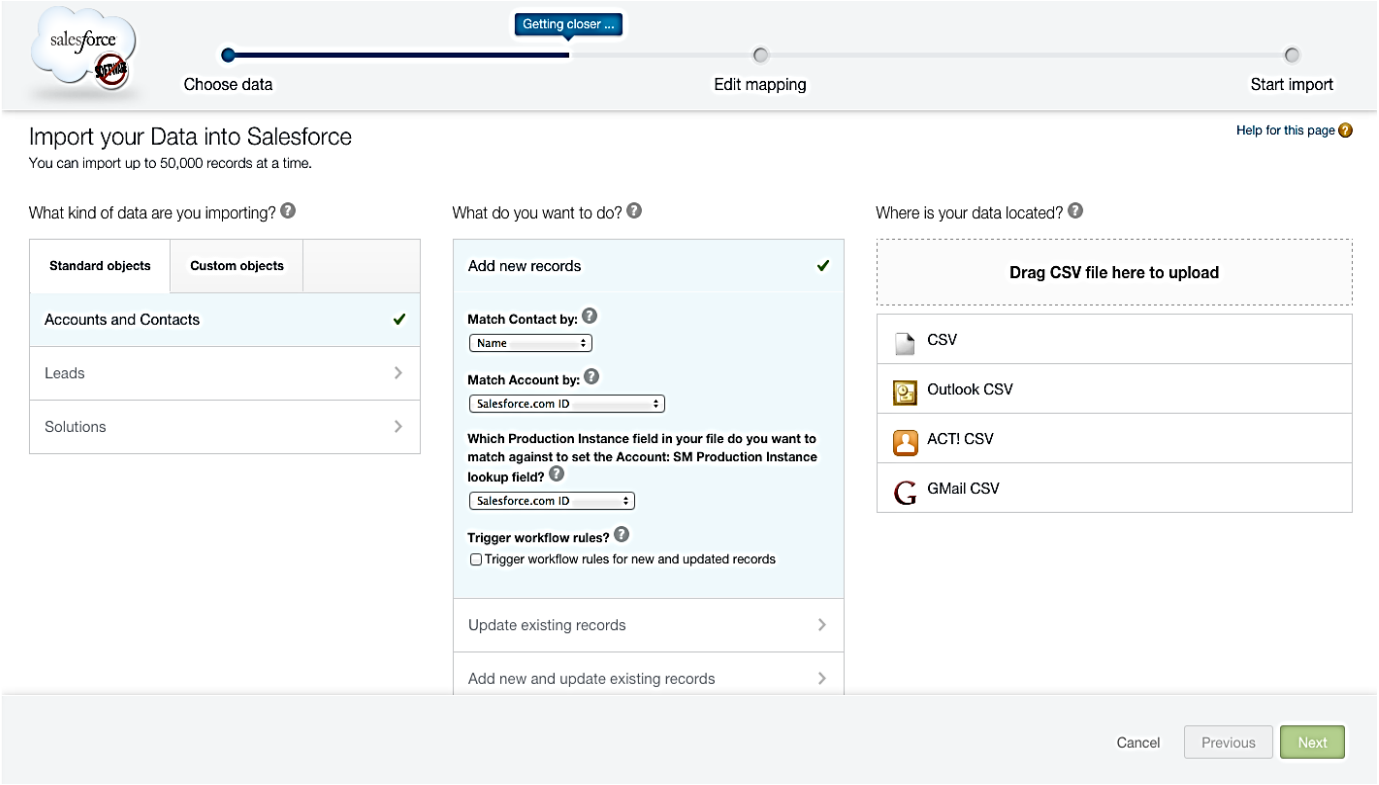
Use web-based import wizards when:

- You are loading fewer than 50,000 records
- The object you need to import is **Account**, **Contact**, **Lead**, **Solution**, and **Custom object**
- You want to prevent duplicates by uploading records according to the account name and site, contact e-mail address, or lead e-mail address

Using the Data Import Wizard
Once you have created an export file and cleaned up the data for import, follow these steps to import data using the **Data Import Wizard.**

1. Start the wizard.
   a. From Setup, enter Data Import Wizard in the Quick Find box, then select **Data Import Wizard.**
   b. Review the information provided on the welcome page, then click **Launch Wizard!**

2. Choose the data that you want to import.
   a. Click **Standard Objects** to import accounts, contacts, leads, or solutions.
      Click Custom Objects to import **custom objects.**
   b. Specify whether you want to add new records to Salesforce, update existing records, or add and update records simultaneously.
   c. Specify matching and other criteria as necessary. Hover over the question marks for more information about each option.
   d. 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.
   e. Choose a character encoding method for your file. Most users will not need to change their character encoding.
   f. Click Next.



3. Map your data fields to Salesforce data fields.
   The Data Import Wizard tries to map as many of your data fields as possible to standard Salesforce data fields. If Salesforce can't automatically map fields, however, you'll need to do it manually. Unmapped fields are not imported into Salesforce.
   To see a list of standard Salesforce data fields, from Setup click **Customize,** click the object whose fields you're interested in, and click **Fields**. For example, if you want to see a list of standard Salesforce fields for leads, click **Customize | Leads | Fields.**
   a. Scan the list of mapped data fields and locate any unmapped fields.
   b. Click Map to the left of each unmapped field.
   c. In the Map Your Field dialog box, choose the Salesforce field(s) you want to map to and click **Map**.
   d. To change mappings that Salesforce performed automatically, click **Change** to the left of the appropriate field, then choose the Salesforce field(s) you want to map to and click Map.
   e. Click **Next.**
4. Review and start your import.
   a. 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.
   b. Click **Start Import.**
5. Check **import status.**

15.0 Data Analytics with Reports and Dashboards

These analytics tools allow you and your users to customize and manage the reporting and visual representation of data. For example, the sales team can produce reports that show the sales pipeline, the marketing team can report on the progress of campaigns, and you can create reports that display the number of active users in your Salesforce organization.

The features available to report data are described in detail and include details of how to create, customize, and export purpose-built report data.

Salesforce CRM analytics consist of the basic mechanisms of reports, dashboards, and folders:

Reports: These are the key building blocks for analytics in Salesforce CRM, where a resulting set of records is displayed in rows and columns to match the specified criteria. Report results can be further filtered and grouped and can also be displayed as graphical summaries.

Dashboards: These are visual components generated from data in reports. These components can include these five types: charts, gauges, tables, metrics, and Visual-force pages.

Folders: These are used to store the reports and dashboards and can either be set to read-only or read/write. To configure which of your users have access to a folder, you can set it to be either accessible by all users, hidden from all users, or accessible only by certain users. When restricting to certain users, the options exist to restrict by **Public Groups**, **Roles**, and **Roles and Subordinates**.

15.1 Reports

When we click on the **Reports** tab, the **Reports and Dashboards** home page presents the following features:

1: This represents the **New Report** and **New Dashboard** buttons.
2: This represents the **Folders** search box, which allows users to search for Specific report and dashboard folders.
3: This represents the **New Report Folder** and **New Dashboard Folder** Selections.
4: This represents the report search box, which allows users to search for specific reports and dashboards.
5: This represents **Reports and Dashboards Folders** pane with different icons to show whether the folder is a report or dashboard.
6: This represents the main **Reports** and **Dashboards** list view section. This Allows the filtering of **Recent Reports** tabs, displays appropriate reports, and Allows the creation of new reports.

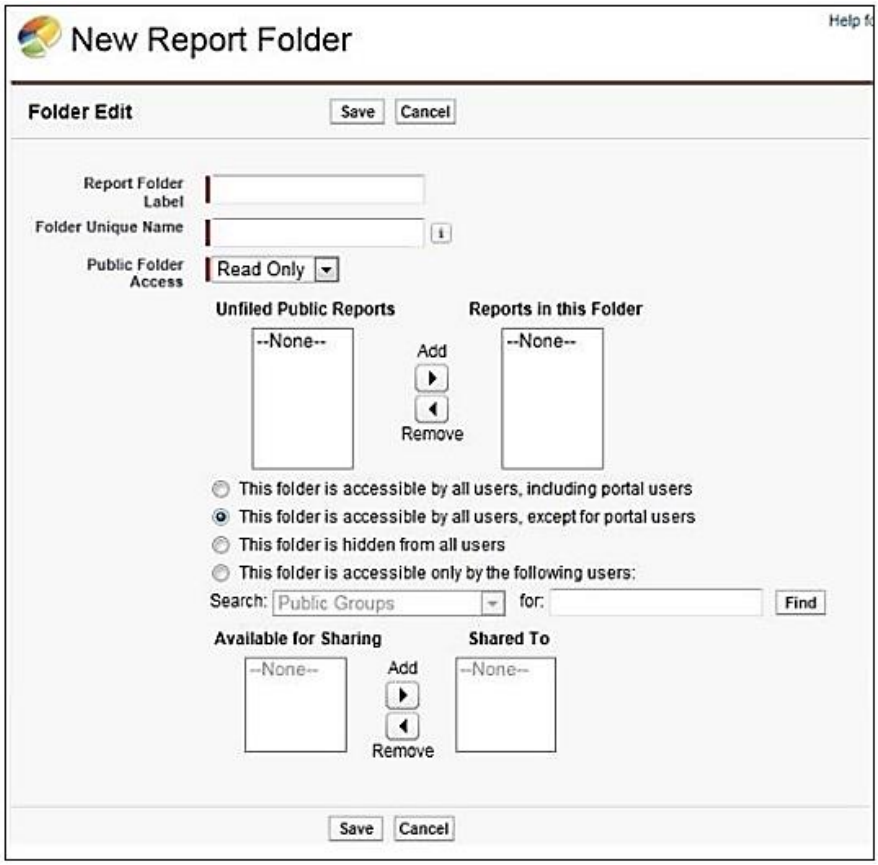
These features can be identified by their respective number, as shown in the following screenshot:

In Salesforce CRM, you cannot save reports to the standard report folders. You can save reports only to the **My Personal Custom Reports** folder, the **Unfiled Public Reports** folder, or any custom report folder for which you have the appropriate read/write access.

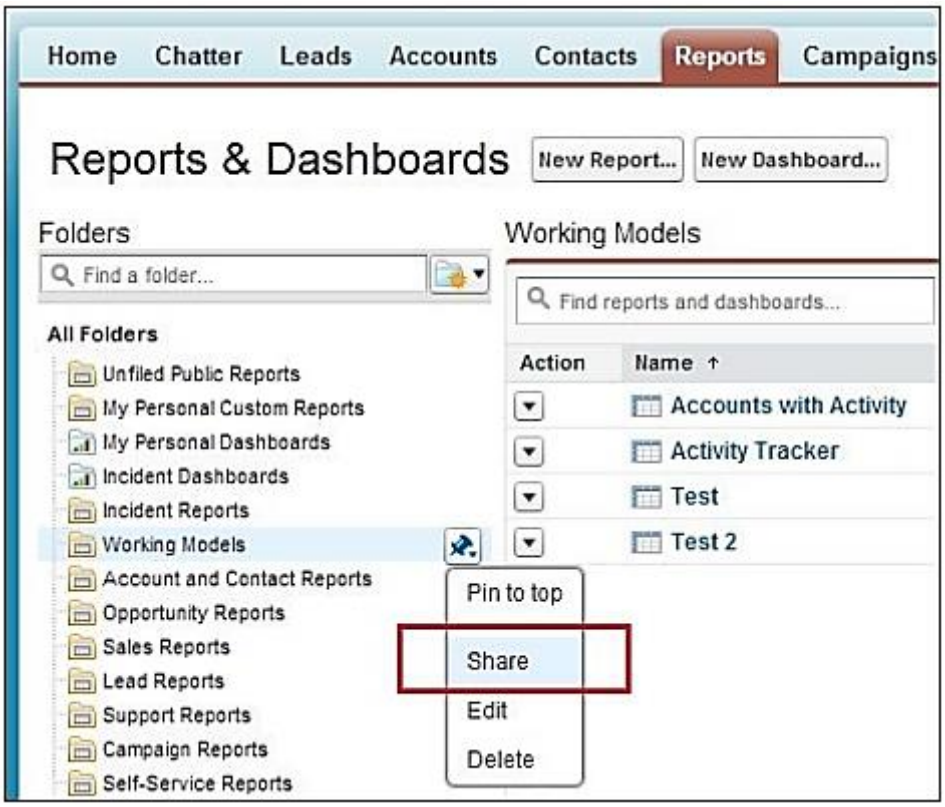
Note

Standard reports cannot be deleted or removed.

To create new report folders, click on the **Create New Folder** link where the **New Report Folder** page is presented, as shown in the following screenshot:



To activate enhanced sharing for reports and dashboards, navigate to
Setup| Customize | Reports & Dashboards | Folder Sharing.
Select the **Enable access levels for sharing report and dashboard folders** checkbox and then click **Save**.
Users are then enabled to share the reports and dashboards folders by clicking to the right of the folder in the folders pane and selecting the **Share** option, as shown in the following screenshot:



The Viewer access setting

The **Viewer** access setting allows users to view, refresh, and run reports and dashboards.

The Editor access setting

The **Editor** access setting provides the same access as **Viewer**, plus the ability to edit, move, save, and delete reports and dashboards.

The Manager access setting

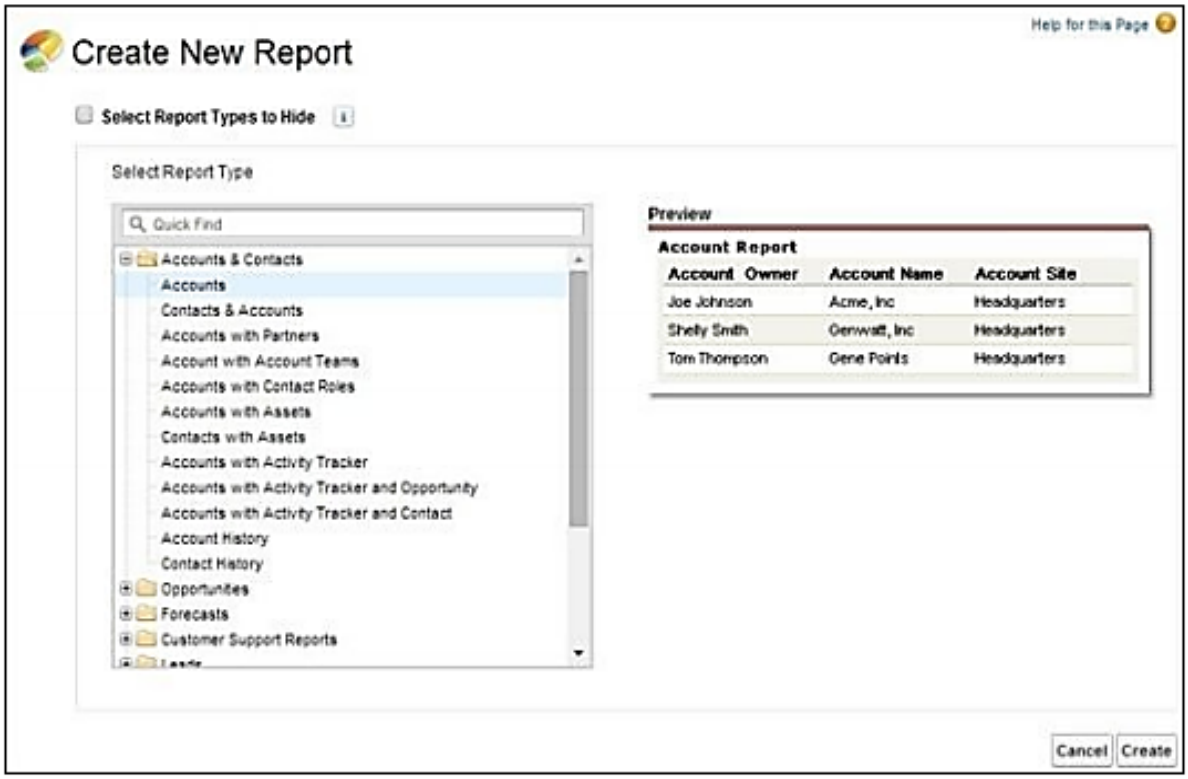
The **Manager** access setting provides the same access as **Editor**, plus the ability to share and rename the folder.

Creating reports

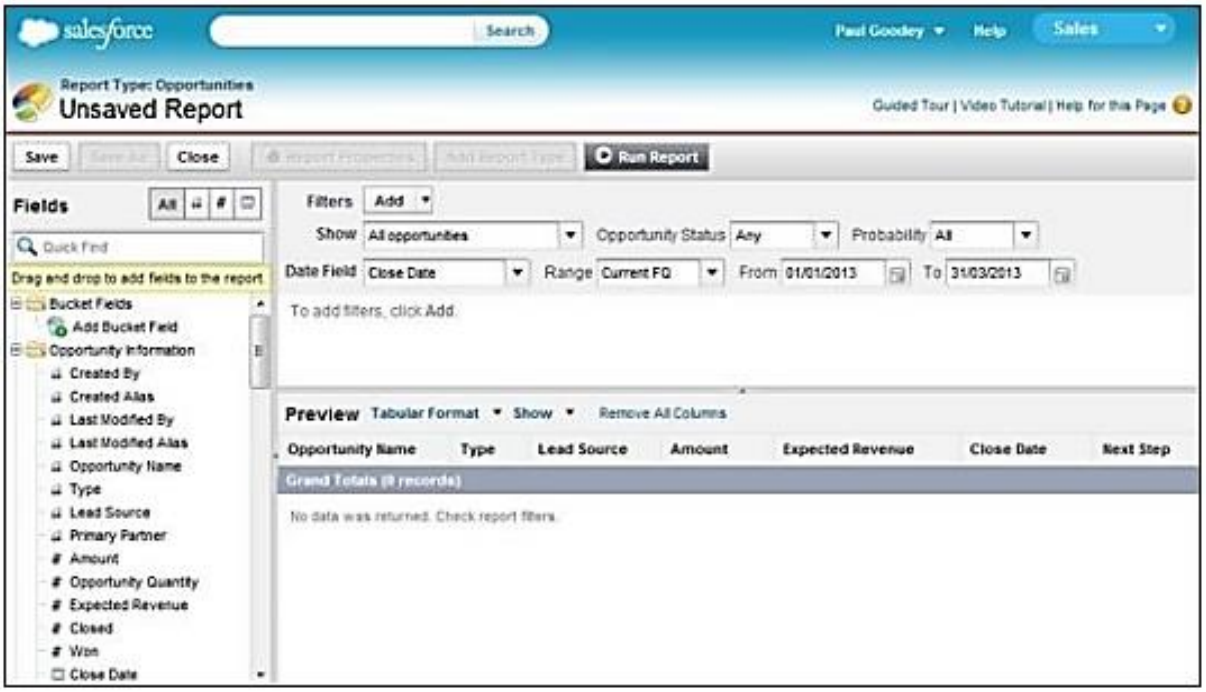
The basic steps to create new reports are as follows:

1. From the **Reports** tab, click on the **New Report…** button.
2. Select the report type for the report and click on the **Create** button.

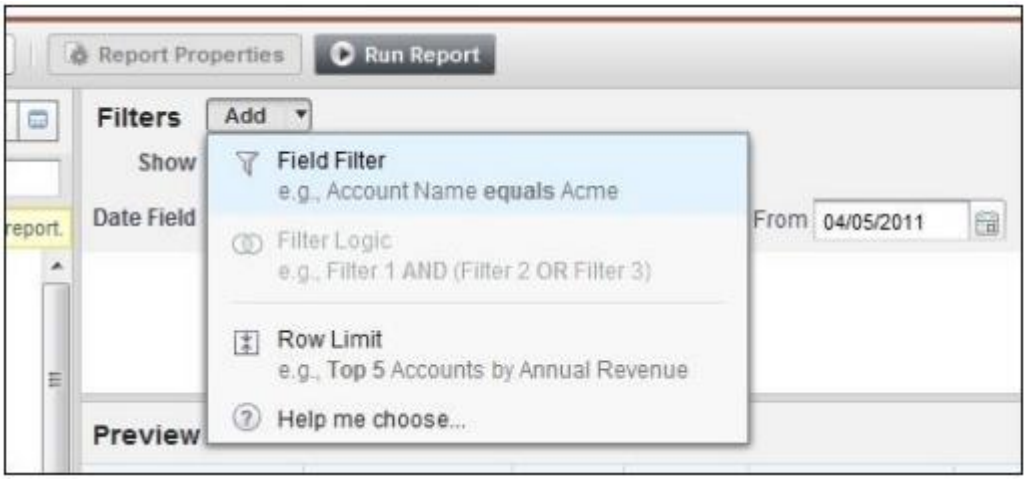
Standard report types



1. Customize your report, and then save or run it.



Within the **Filters** pane, you click on the **Add** box to add report filters, as shown here:



Report formats

These four report formats are available in Salesforce CRM: **Tabular**, **Summary**, **Matrix**, and **Joined**.

The Tabular report format

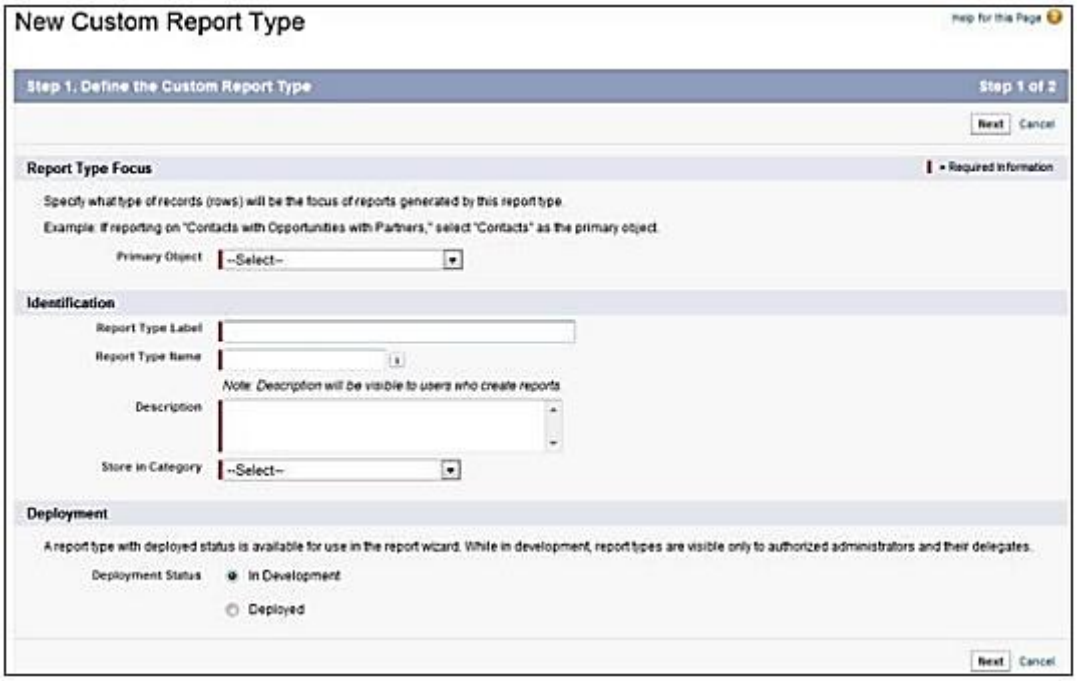
Tabular reports are the easiest and quickest way to report data. They can be linked to a spreadsheet, where they comprise a set of records listed in rows and fields (ordered in columns). Tabular reports are most suited for creating lists of records or a list with
a single grand total, as they cannot be used to group data.

Note
Tabular reports cannot be used in dashboards unless the number of rows that are returned is limited.

The Summary report format
Summary reports are similar to tabular reports, except that they allow the grouping of rows of data. They can be used for reports to show subtotals based on the value of a field. Summary reports with no groupings are simply displayed as tabular reports.

Note
Summary reports can be used as the source report for dashboard components.

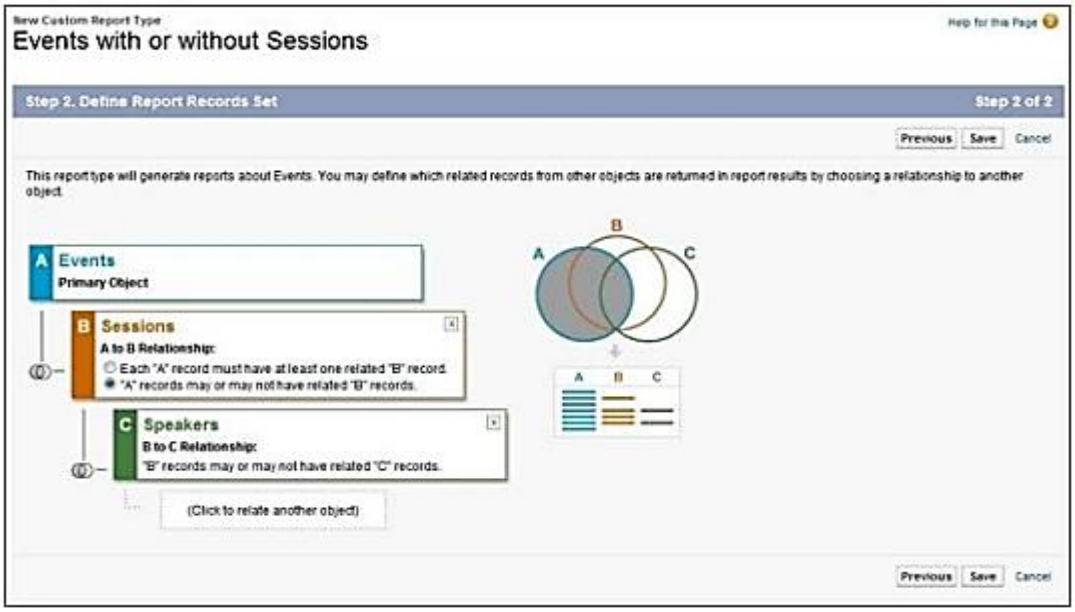
The Matrix report format
Matrix reports are similar to summary reports, but they also allow the grouping and summarization of data by both rows and columns and can be used to compare related totals. Matrix reports are useful for summarizing large amounts of data to compare values in several different fields or to analyse data by date or by product, person, region, and so on.
Note
Matrix reports can be used as the source report for dashboard components.
The Joined report format
Joined reports are reports that can store and group multiple reports together and allow you to build a single report that contains data from multiple report types. A joined report can have up to five report blocks that can be added from either standard or custom report types, but can only be included if they share a common object relationship. For example, if you have a joined report that contains the Opportunities report type, you can then add the Contacts report type as both Opportunity and Contact objects have a relationship with the Accounts object. For joined reports with multiple report types, any field that is shared by all report types is known as a common field. Common fields appear in the **Common Fields** area in the **Fields** pane and can be used to group together the separate report blocks.
Note
Joined reports can be used as the source report for dashboard components if the joined report includes a report chart by configuring the dashboard component with the **Use charts defined in the source report** setting.
These features are not available in joined reports: **Bucket** fields, **Cross** filters, and **The Rows to Display** filters.
Custom report types
Creating custom report types
Custom report types are set up using the following steps:
1. Define a custom report type by name, description, primary object, development status, and the category of report to store it.
2. Choose the related objects for the custom report type.
3. Specify the layout for the resulting standard and custom fields that a report can display when created using the custom report type.
4. Create a report from the Custom Report Type template to verify that all of the objects and field definitions are correct.
Defining custom report types
To navigate to the **Custom Report Types** page, navigate to **Setup** | **Create** | **Report Types**. Now, click on **New Custom Report Type**.



Step 1 – Define the Custom Report Type

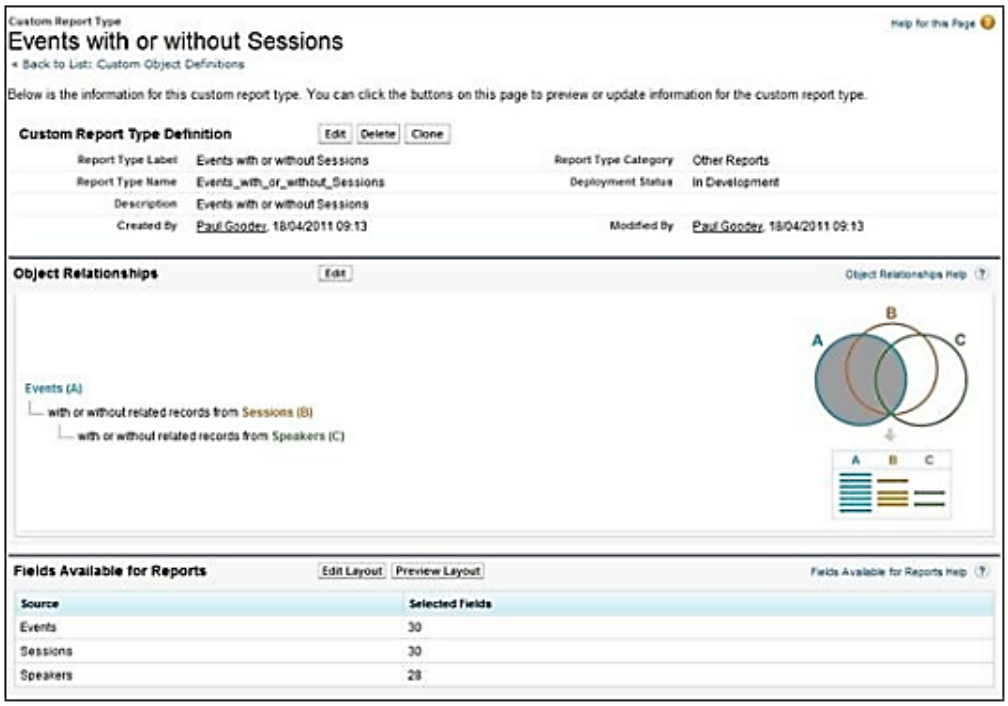
From the **Primary Object** drop-down list, select the primary object from which you want to build your custom report type.

To add an object that is associated with another object to the report type, click on the rectangle section, which is **(Click to relate another object)**. Then, select the object from the picklist. The objects available for you to choose from are based on the primary object's relationships with other objects. You can add up to 4 related objects.

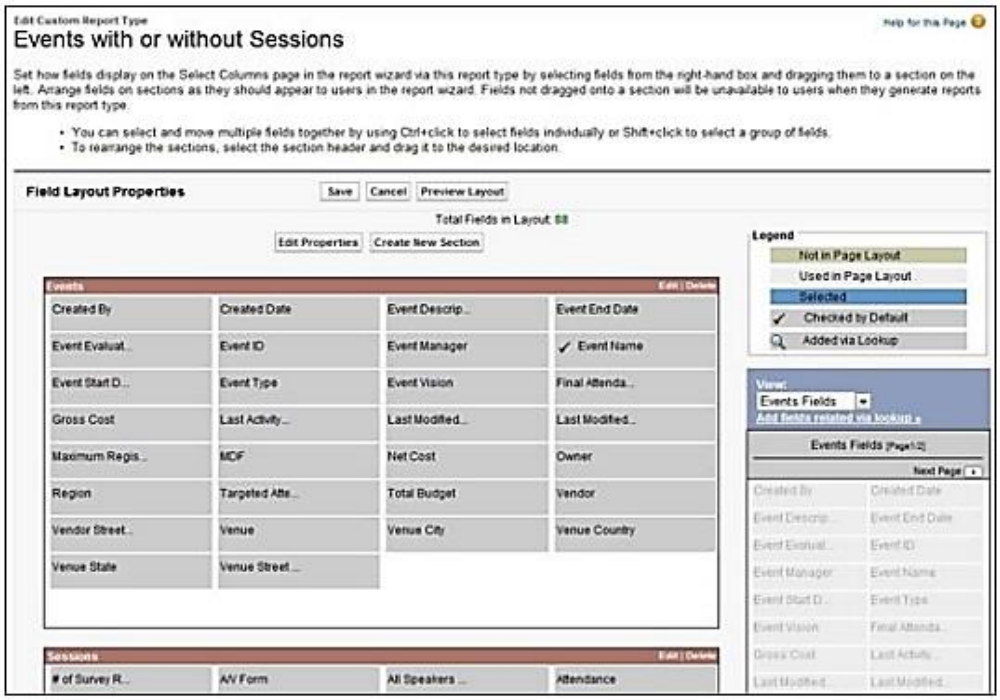


Edit layout

After clicking on **Save**, the Custom Report Type definition and the object relationships are set, as shown in the following screenshot:



To start configuring the layout, click on **Edit Layout** and select fields from the right-hand side box and drag them to a section on the left-hand side, as shown in the following screenshot:



15.2 Dashboards

Dashboards are visual information snapshots that are generated from the data in associated reports and are presented as graphical elements. These graphical elements are known as dashboard components, of which there are these five types:

- charts,
- gauges,

- tables,
- metrics,
- Visualforce pages.

Dashboards can have up to 20 components, and you can control users' access to dashboards by storing them in folders with appropriate permissions, where folders can be public, hidden, or restricted to groups or roles.

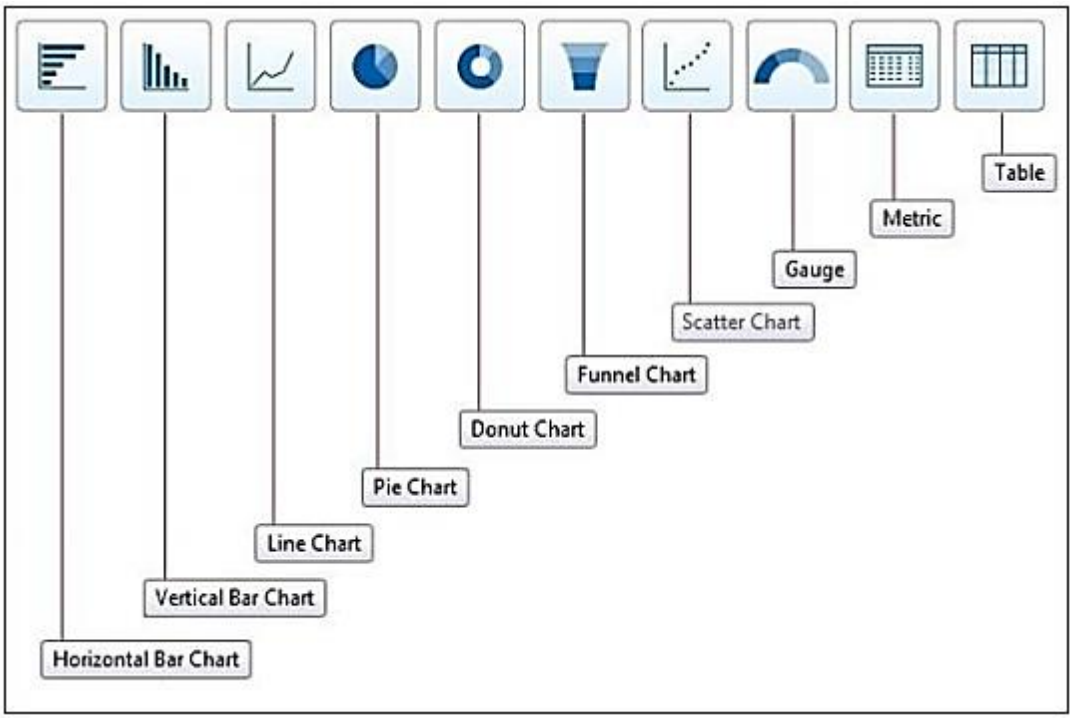
Dashboards can be further configured to run with the concept of a **running user**, which means that the named user's security settings determine what data is to be displayed. Here, all dashboard viewers see data according to the security settings of the user who has been set as the running user irrespective of the dashboard viewer's own personal security settings.

A more flexible and dynamic approach, however, allows you to set the running user to be the logged-in user so that each user is presented with the dashboard according to their own data access level. This is known as **dynamic dashboards**.

Dashboard component types

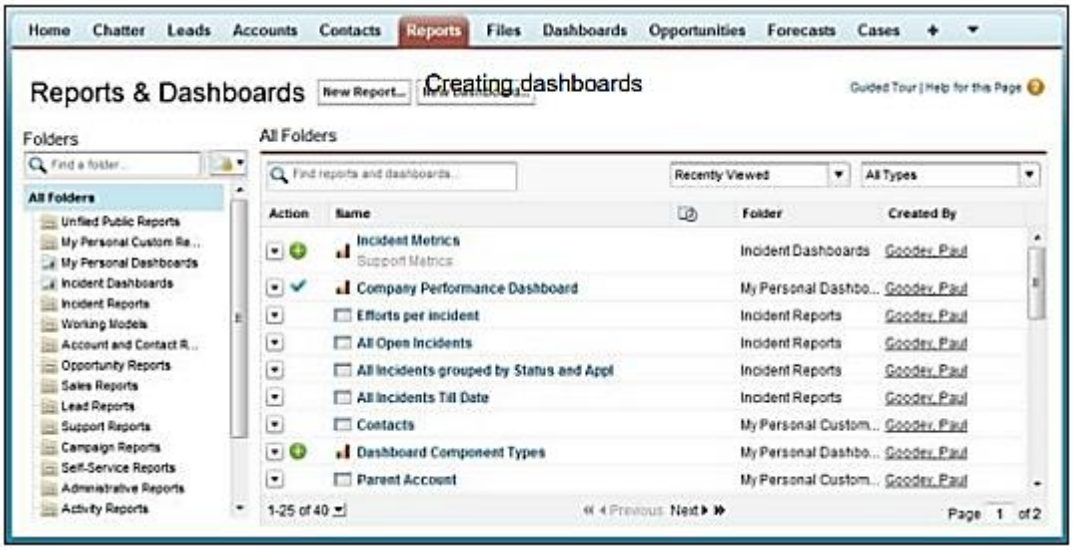
In Salesforce CRM, the following dashboard component types are available:

- **Horizontal Bar Chart**
- **Vertical Bar Chart**
- **Line Chart**
- **Pie Chart**
- **Donut Chart**
- **Funnel Chart**
- **Scatter Chart**
- **Gauge**
- **Metric**
- **Table**



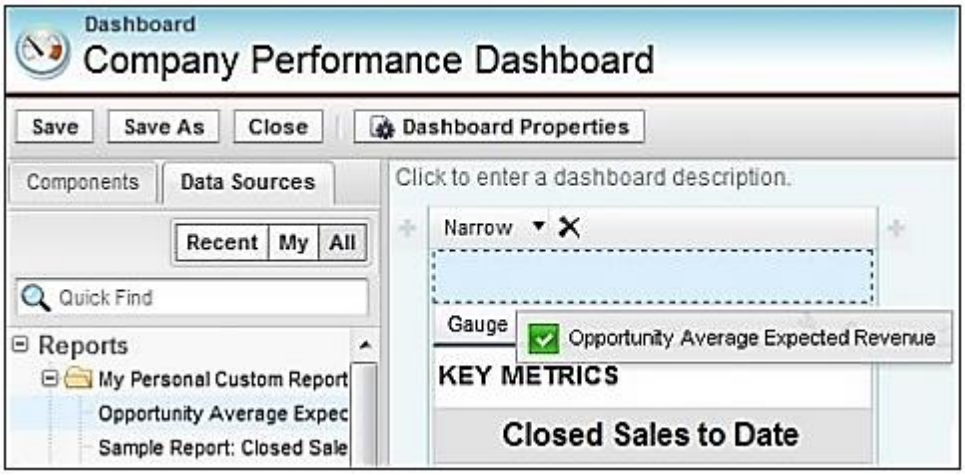
Creating dashboards

To create a dashboard, click on the **Reports** tab. This then presents the common reports and dashboards' main page with the **Reports & Dashboards** heading. On this page, click on the **New Dashboard…** button, as shown in the following screenshot:

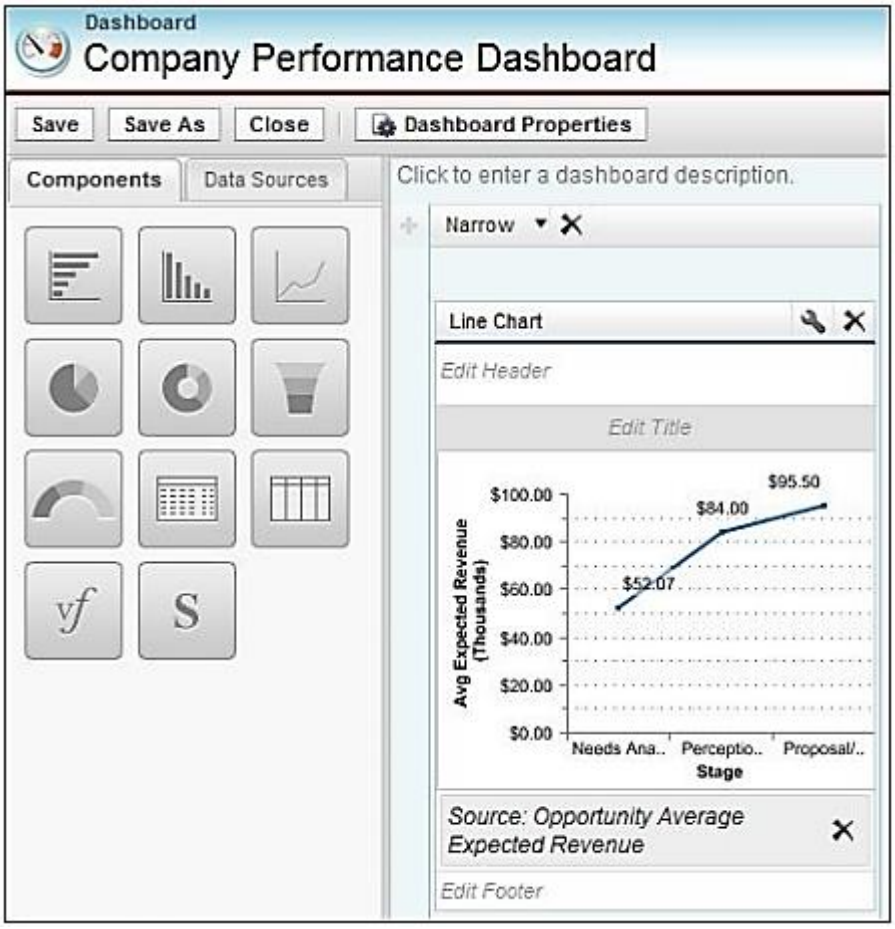


To drag and drop a line chart from the **Components** tab, you simply select, hold, and drag the icon onto the source,

The following screenshot shows you the drag and drop feature using a report from the **Data Sources** tab:



The following is the result:



Setting dashboard properties

To set dashboard properties, follow these steps:

1. Edit a dashboard and click on **Dashboard Properties**.
2. Enter a title for the dashboard.
3. Select a folder to store the dashboard.
4. Under **Component Settings**, select the title color and size, text color, and background fade. If you don't want a gradient, choose the same color for both

Starting Color and **Ending Color**.

5. Click on **Save**.

Analytic snapshot

An analytic snapshot is used to run a report on historical data. We can save tabular or summary report results to fields on a custom object, then map those fields to corresponding fields on a target object. We then schedule when to run the report to load the custom object's fields with the report's data. For example, the library manager can setup an analytic snapshot that reports on the books checkout every day at 5:00 p.m. and store the data in a custom object. The library manager can then spot the trends in book issuing and the readership.

Setting up analytical snapshot

Before we setup an analytical snapshot we require two things:

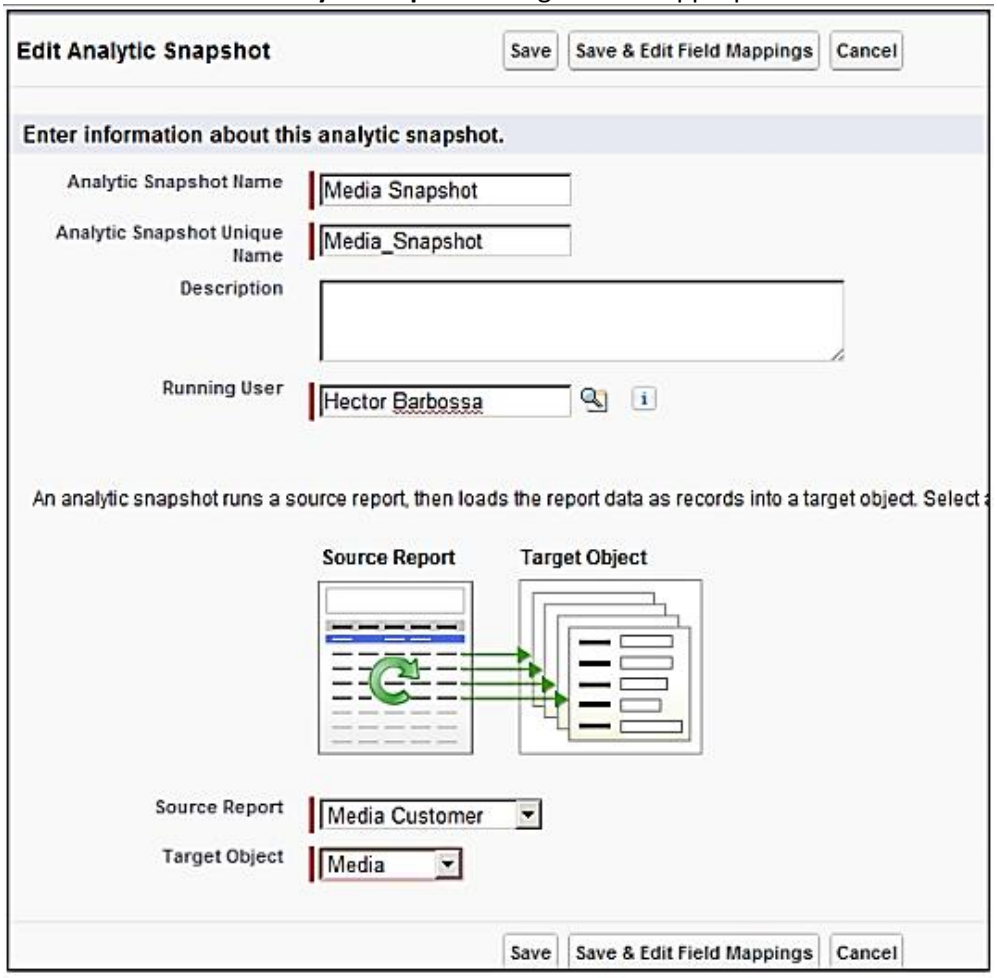
- **Source report**: The source report is the custom report created by using the report builder. It can have columns of more than one object.
- **Target object**: The target object is the custom object that contains fields that are similar to columns in the report.

We need to make sure all the columns in the report map with the fields in the target object. The fields should be created on the target object.

If you need help in setting up the report or the object, refer to the previous chapter.

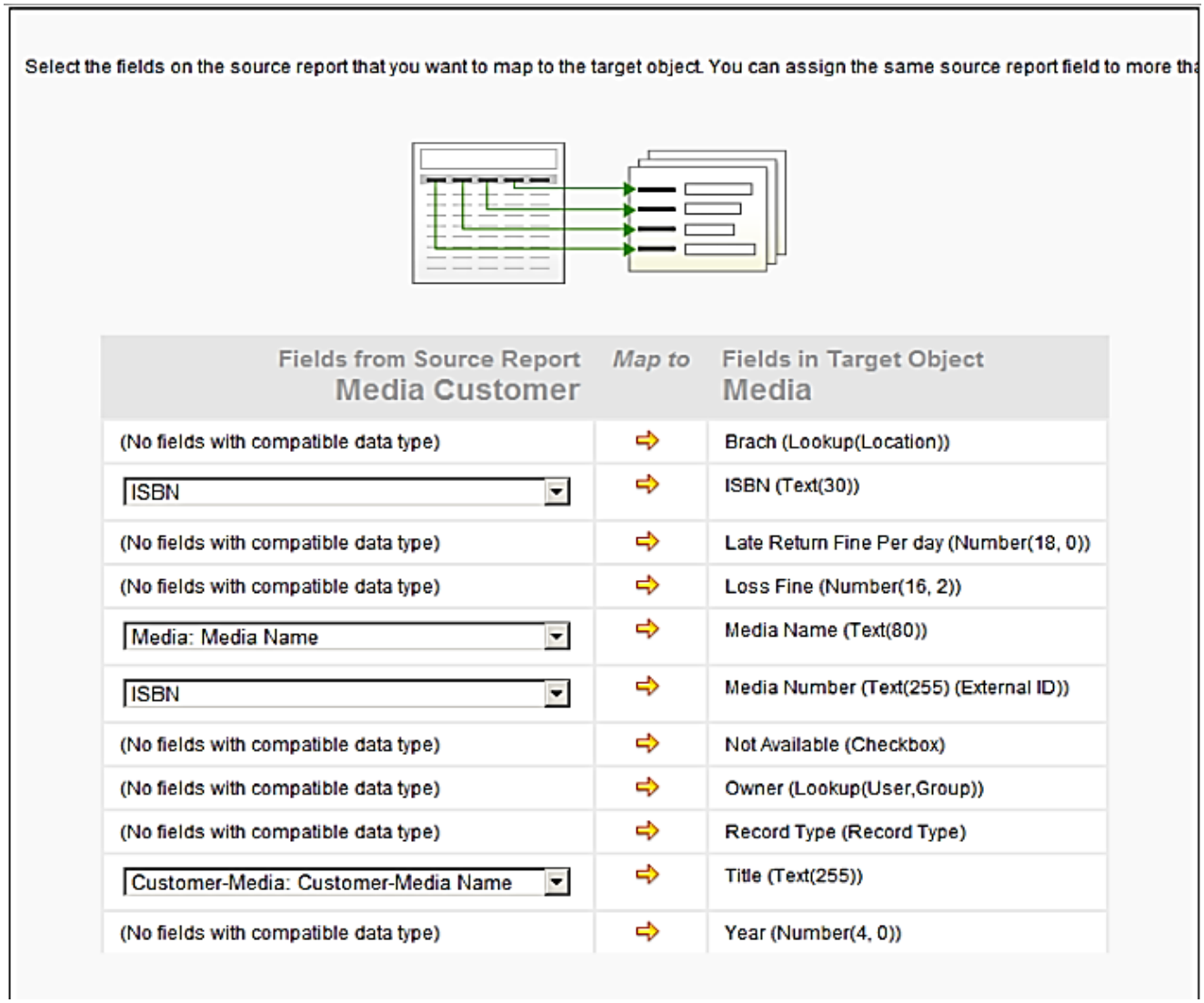
Once these two are in place, let us proceed to setup the analytical snapshot.

1. Go to **User Name | Setup | Administration Setup | Data Management | Analytic Snapshots**.
2. Click on **New Analytic Snapshot** and give it the appropriate name.



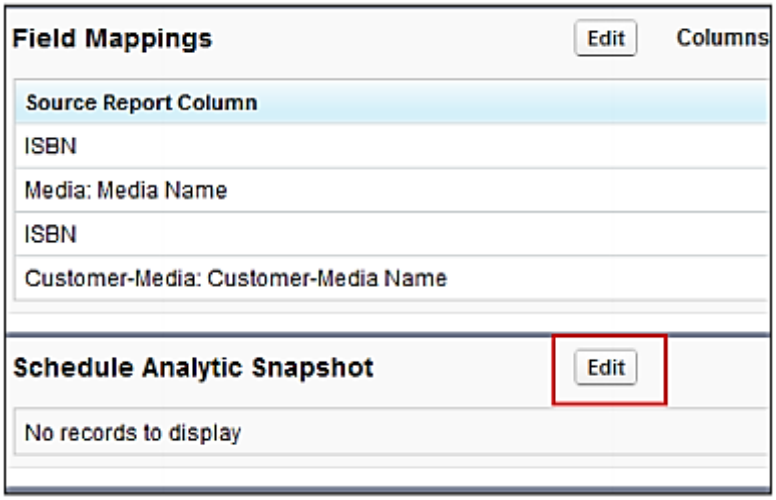
3. As shown in the screenshot, the data loaded in the object will be based on the sharing settings for the running user.
4. Select the target object and the source report, now click on **Save and Edit Field Mappings**. The mapping page allows us to map the fields from the report to the fields in the object.
5. If the source report is a summary report select the level of grouping as shown in the following screenshot.

6. Map the fields from the source report to the target object as shown in the following screenshot:

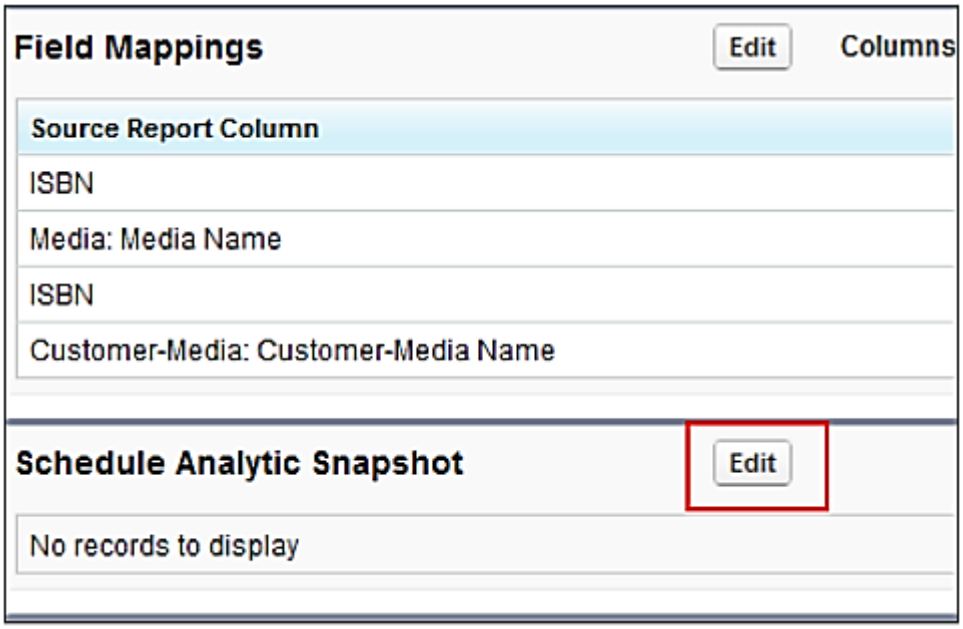


Fields from Source Report Media Customer	Map to	Fields in Target Object Media
(No fields with compatible data type)	⇨	Brach (Lookup(Location))
ISBN	⇨	ISBN (Text(30))
(No fields with compatible data type)	⇨	Late Return Fine Per day (Number(18, 0))
(No fields with compatible data type)	⇨	Loss Fine (Number(16, 2))
Media: Media Name	⇨	Media Name (Text(80))
ISBN	⇨	Media Number (Text(255) (External ID))
(No fields with compatible data type)	⇨	Not Available (Checkbox)
(No fields with compatible data type)	⇨	Owner (Lookup(User,Group))
(No fields with compatible data type)	⇨	Record Type (Record Type)
Customer-Media: Customer-Media Name	⇨	Title (Text(255))
(No fields with compatible data type)	⇨	Year (Number(4, 0))

7. Click on **Edit** on the **Schedule Analytical Snapshot** related list as shown in the following screenshot:



8. The scheduler is shown in the preceding screenshot, it is similar to the one used in reports and dashboards and has similar options hence won't be repeated again.



Now that we have a record of the historical data in the target object we can run the report on it, to analyze trends.