



[Salesforce CPQ by Smriti Sharan](#)

Salesforce CPQ Bible

Covered Every Salesforce CPQ Interview

Question Possible  





[Salesforce CPQ by Smriti Sharan](#)

My name is Smriti Sharan. I am an avid blogger and YouTuber. I break down complex concepts with fun relatable real-time examples so that

learning is fun. Wow!!!

You Can connect with me on:

[LinkedIn](#)

[Subscribe to my YouTube Channel](#)

[Follow my Blog](#)



[Salesforce CPQ by Smriti Sharan](#)

Topics Covered

**I have covered every interview question on Salesforce CPQ
in detail.**

- CPQ Basics
- Types of Products
- Product Rule
- Price Rule
- Quote
- Quote Line
- Quote Template
- Contract Management
- General Errors



[Salesforce CPQ by Smriti Sharan](#)

CPQ

Basics



[Salesforce CPQ by Smriti Sharan](#)

What is Salesforce CPQ?



Imagine you're at a car dealership, and you want to buy a new car.

Salesforce **CPQ** is like a very-smart assistant that helps the salesperson put together the perfect car for you, quickly and accurately.

Configure: First, you choose what you want in your car. Do you want a sunroof? Leather seats? A specific color? **The assistant makes sure that all the parts you choose fit together perfectly.**



[Salesforce CPQ by Smriti Sharan](#)

Price: Next, the [assistant calculates the total cost](#) of the car with all your chosen features. It also checks if there are [any special deals or discounts](#) you can get, so you don't miss out on saving some money.

Quote: Finally, the assistant gives the salesperson a quote, which is a word for the [final price and summary](#) of what you're buying. This quote is super detailed and shows everything you've chosen for your car, how much each part costs, and the total amount you need to pay.



[Salesforce CPQ by Smriti Sharan](#)

What are the key Features of Salesforce CPQ?



1. Product Configuration: Salesforce CPQ allows users to configure products or services based on [predefined rules and constraints](#). It ensures that only valid and feasible products or services can be selected, eliminating configuration errors and increasing customer satisfaction.



[Salesforce CPQ by Smriti Sharan](#)

2. Pricing and Discounting: Salesforce CPQ enables businesses to define [pricing rules and discounts based on various criteria](#) such as quantity, customer type, or specific promotions. It ensures consistent and accurate pricing, eliminating manual calculations and reducing pricing errors.

3. Quoting and Proposal Generation: With Salesforce CPQ, sales representatives can [quickly generate professional-looking quotes](#) and proposals customized to meet specific customer requirements.

4. Guided Selling: Salesforce CPQ provides guided selling capabilities, helping sales representatives [recommend the most suitable products](#) or services based on customer needs, preferences, and constraints. [It ensures upsell and cross-sell opportunities are maximized](#), increasing sales effectiveness.

5. Integration with CRM: As a native Salesforce application, [Salesforce CPQ seamlessly integrates with the Salesforce CRM platform](#). This



[Salesforce CPQ by Smriti Sharan](#)

integration enables a unified view of customer data, including quotes, orders, and customer history. It allows sales teams to have a holistic understanding of customer interactions, facilitating better decision-making and customer relationship management.

Explain the overview of the Salesforce CPQ life cycle?

Lead Generation: The CPQ life cycle [begins with lead generation](#), where [potential customers show interest](#) in your products or services.

Opportunity Creation: Once a [lead is identified](#), it is converted into an [opportunity in Salesforce](#). This stage involves gathering customer information and assessing their needs.

Product Configuration: In this stage, the sales team utilizes Salesforce [CPQ to configure products or services](#) according to the customer's requirements. CPQ allows for [complex product configurations](#), [bundling options](#), and [customization](#).



[Salesforce CPQ by Smriti Sharan](#)

Pricing and Discounting: Salesforce CPQ enables the sales team to apply appropriate pricing and discounts based on customer-specific requirements, volume-based pricing, or contractual agreements. This stage ensures accurate pricing and maximizes revenue potential.

Quote Generation: With the product configuration and pricing determined, Salesforce CPQ generates a professional quote tailored to the customer's needs. The quote includes detailed information about the products or services, pricing, and terms.

Quote Approval and Negotiation: The generated quote goes through an approval process, where appropriate stakeholders review and provide their consent. This stage may involve negotiation with the customer to finalize the quote details.

Quote Acceptance: Once the customer accepts the quote, it progresses to the next stage of the CPQ life cycle.



[Salesforce CPQ by Smriti Sharan](#)

Order Creation: The [accepted quote is converted into an order](#), which triggers the fulfillment process, including inventory management, order processing, and shipping.

Contract Management: Salesforce CPQ provides contract management capabilities, [allowing the sales team to generate and manage contracts based on the accepted quote](#). This stage ensures legal compliance and streamlines contract lifecycle management.

Revenue Recognition: Salesforce [CPQ integrates with billing systems and financial processes to facilitate accurate revenue recognition](#). This involves tracking and reporting revenue based on the delivered products or services.

Renewal and Upsell: As contracts near their expiration, Salesforce CPQ assists in [managing renewal opportunities](#). The system can identify upsell or cross-sell opportunities and guide the sales team in maximizing customer value.



[Salesforce CPQ by Smriti Sharan](#)

Reporting and Analytics: Throughout the CPQ life cycle, Salesforce CPQ provides reporting and analytics capabilities to [monitor key metrics](#), [track sales performance](#), [identify trends](#), and [make data-driven decisions](#).

What is the history of CPQ software?



The 1980s was a huge decade for technology. Personal computers, CD's, video game consoles and CPQ software all made their debut during the '80s. [CPQ software](#), known then as the “[configurator](#),” helped sales reps manage [how customers choose their specifications](#).



[Salesforce CPQ by Smriti Sharan](#)

As the '90s rolled in and sales force automation took over the industry, CPQ entered the front office. Through the 2000's it evolved to play a key role in eCommerce.

CPQ software really hit its stride in the 2010s as technological innovations allowed for streamlined service. Businesses could utilize CPQ software as a highly detailed quote generator.

What is the difference between Salesforce CPQ and Conga CPQ?

Conga CPQ was [originally built for the Salesforce platform](#). Conga was first released on the [Salesforce AppExchange in 2006](#). Later Conga found success as a standalone CPQ that integrates numerous systems. Conga CPQ in Salesforce refers to [Conga's ability to integrate with Salesforce CRM](#), allowing businesses to manage pricing, contracts, and customer data from the same platform while syncing information between the two.

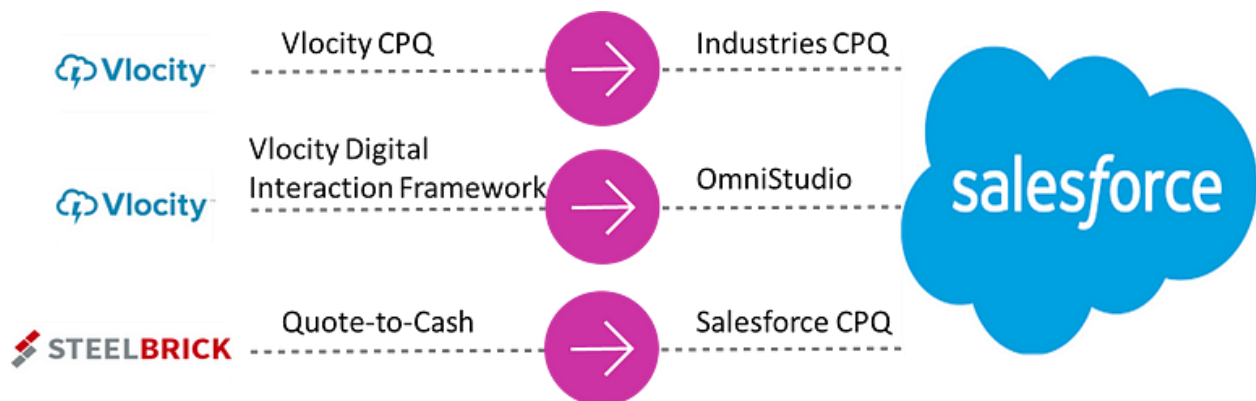


[Salesforce CPQ by Smriti Sharan](#)

Apttus was an innovator in Configure, Price, Quote (CPQ) solutions and robust CLM tools. After the [merger of Apttus and Conga in 2020](#), Apttus CPQ continues to operate under the name Conga CPQ.

Conga's software is designed on a microservices architecture, granting it the ability to manage [substantial volumes of quotes and contracts with ease](#). This makes the platform highly scalable for businesses that need to [manage complex pricing structures and high volumes of orders](#). Conga is designed for complex pricing models.

Explain the Difference between Salesforce CPQ and Salesforce Industry CPQ?





[Salesforce CPQ by Smriti Sharan](#)

A. Vlocity i.e. Industry Cloud

Vlocity (Salesforce Industries) offers pre-built solutions on the Salesforce platform that solve the needs of industries that [require CRMs to be heavily customized beyond Salesforce in its out-of-the-box state.](#)

[Founded in 2014](#), Vlocity quickly gained force as an “[Industry Cloud](#)” pioneer. In 2020, Salesforce acquired Vlocity. As a result of the transaction, Salesforce was able to add Vlocity’s industry-specific CRMs to its existing products, known as Salesforce Industries.

B. Salesforce CPQ

[Steelbrick started in 2010](#). Steelbrick was natively built on Salesforce Platform which offered quote-to-cash solution. [Salesforce acquired Steelbrick in 2015 for \\$360 million](#). [Steelbrick’s CEO was Godard Abel](#). It’s not specific to any Industry. It’s a general application which can fit with configuration and customization to any Industry. Salesforce CPQ is sold as a separate product unlike Industries CPQ.



[Salesforce CPQ by Smriti Sharan](#)

Which CPQ is better?

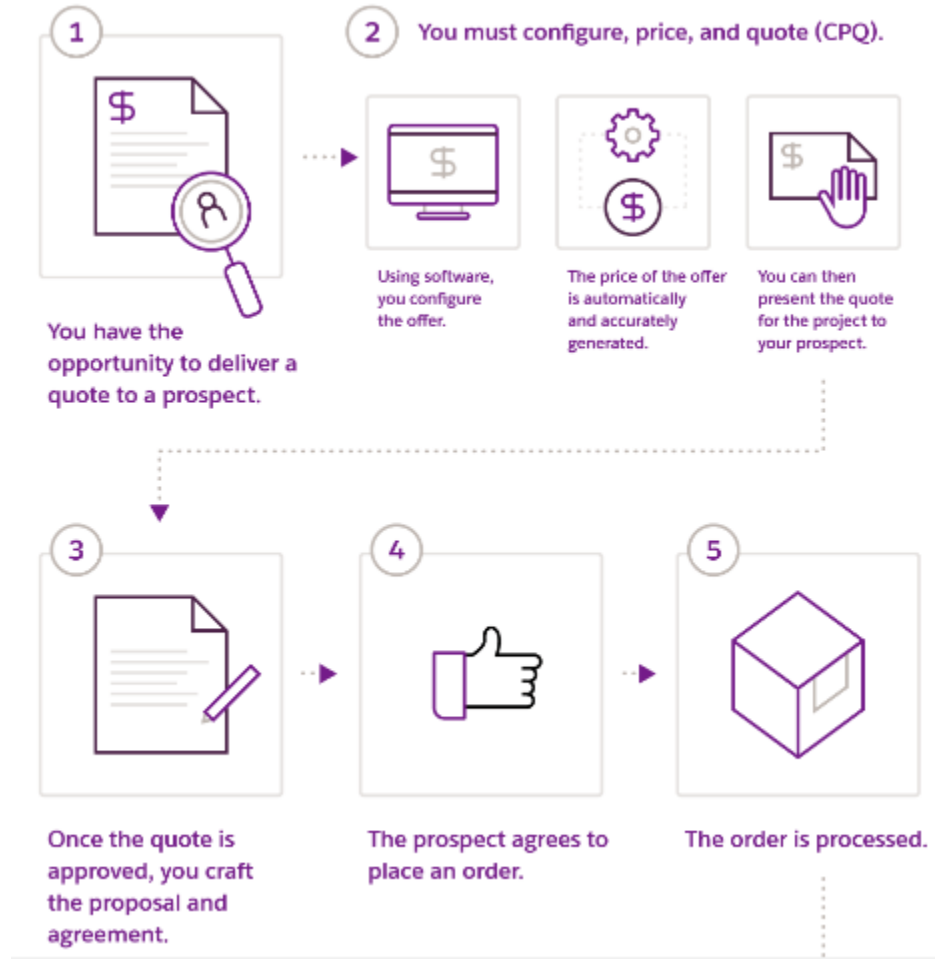
If the customer is from Communication, [Media & Energy](#) then go for [Industry Cloud](#) (Industries CPQ) and If the customer is from any other Industry go for Salesforce CPQ.

Explain the concept of quote-to-cash (QTC) in Salesforce?

*“Research indicates that **66 percent of all sales quotes end without a purchase.** Salespeople can beat these odds when they are thorough, accurate, and timely with their sales quotes.”*

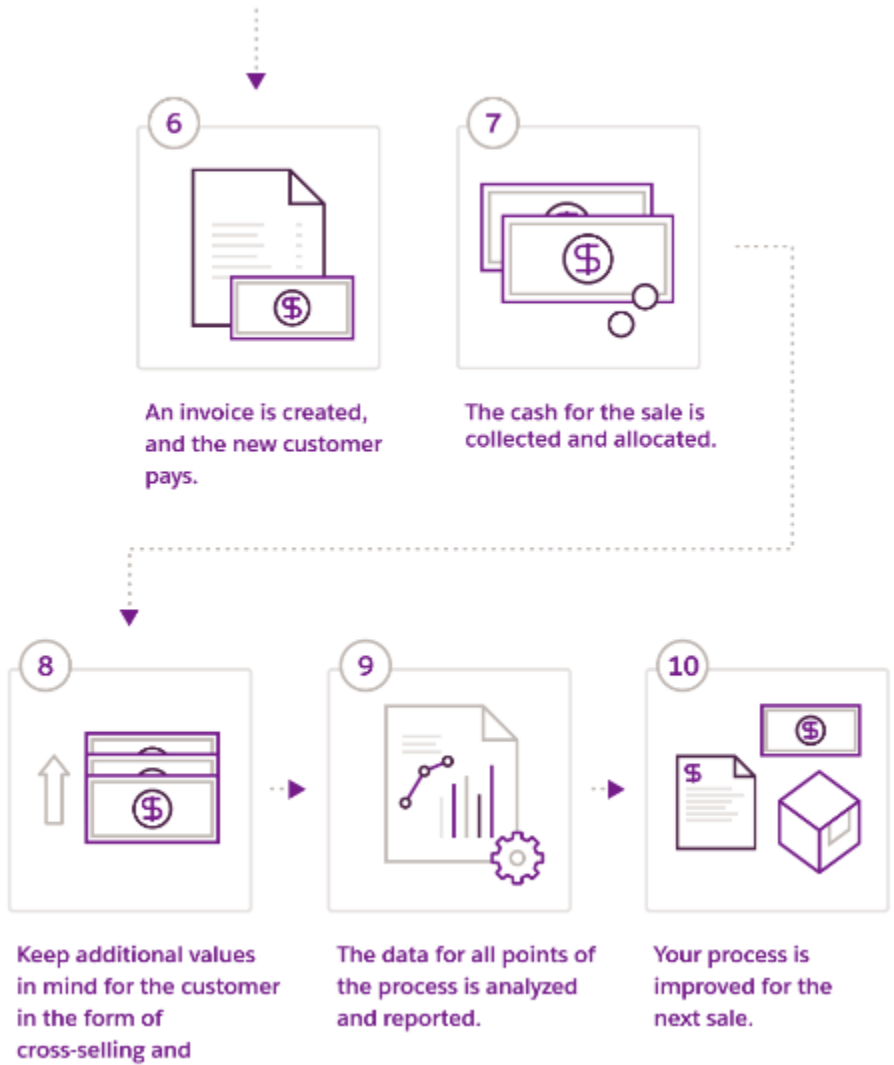


[Salesforce CPQ by Smriti Sharan](#)





[Salesforce CPQ by Smriti Sharan](#)





[Salesforce CPQ by Smriti Sharan](#)

To understand the Quote-to-Cash (QTC) process. Let's use **frozen yogurt business** 'Frosty Bliss':



1. Quote Creation: A customer contacts Frosty Bliss wanting a bulk order of 150 custom frozen yogurt cups for an event, with unique flavor combinations and toppings. The sales rep, using [Salesforce](#), [inputs the customer's requirements](#):

- Choice of 3 flavors: chocolate, strawberry surprise, and magnificent mango.



[Salesforce CPQ by Smriti Sharan](#)

- 5 exclusive toppings.
- Custom-branded cups with the event's logo.

Salesforce calculates the ingredients, labor, and customization, automatically generating a quote for \$750.

2. Quote Approval: The quote is sent to the customer through an [integrated app like DocuSign, offering a seamless review and electronic signature process](#). The customer makes adjustments, requesting an additional 50 cups, bringing the total to 200 cups and the updated quote to \$1,000. They electronically sign the adjusted quote, and Salesforce is instantly updated with this commitment.

3. Order Management: Salesforce automatically [converts the approved quote into an order](#), eliminating the need for the sales rep to manually re-enter details. The system sends an order confirmation to the customer and notifies the production team to schedule the order for a specific date.



[Salesforce CPQ by Smriti Sharan](#)

4. Production and Delivery: The production team, informed by Salesforce, starts preparing the custom order. They use a [third-party logistics integration like ShipStation or Shopify to manage the delivery](#). Once the order is ready, the team ships the 200 custom frozen yogurt cups, updates the status to “shipped” in Salesforce, and the customer receives a real-time notification of the shipment.

5. Billing: The finance department receives an automatic prompt from Salesforce to generate an invoice once the order is shipped. The invoice, totaling \$1,000, is emailed to the customer with a note indicating a due date within 15 days, thanks to an integrated accounting tool like QuickBooks.

6. Payment: The customer completes the payment using an online link facilitated by a secure payment gateway integrated with Salesforce, like Stripe. Upon successful transaction, the payment status is recorded in Salesforce, and the order status is updated to “completed.” The sales rep, customer, and finance department are all automatically notified of the order’s successful completion.



[Salesforce CPQ by Smriti Sharan](#)

Types of Products



[Salesforce CPQ by Smriti Sharan](#)

What is the product?

It is an item, bundle or service offered to customer for a price.

What is Product-based Selling?

Item: It is offering a single item to the customer. Eg shoes, ice-cream etc.

Bundle: Package multiple products together and sell it to customer

A perfect example is Cadbury celebrations.



[Salesforce CPQ by Smriti Sharan](#)



What is Service Based Selling?

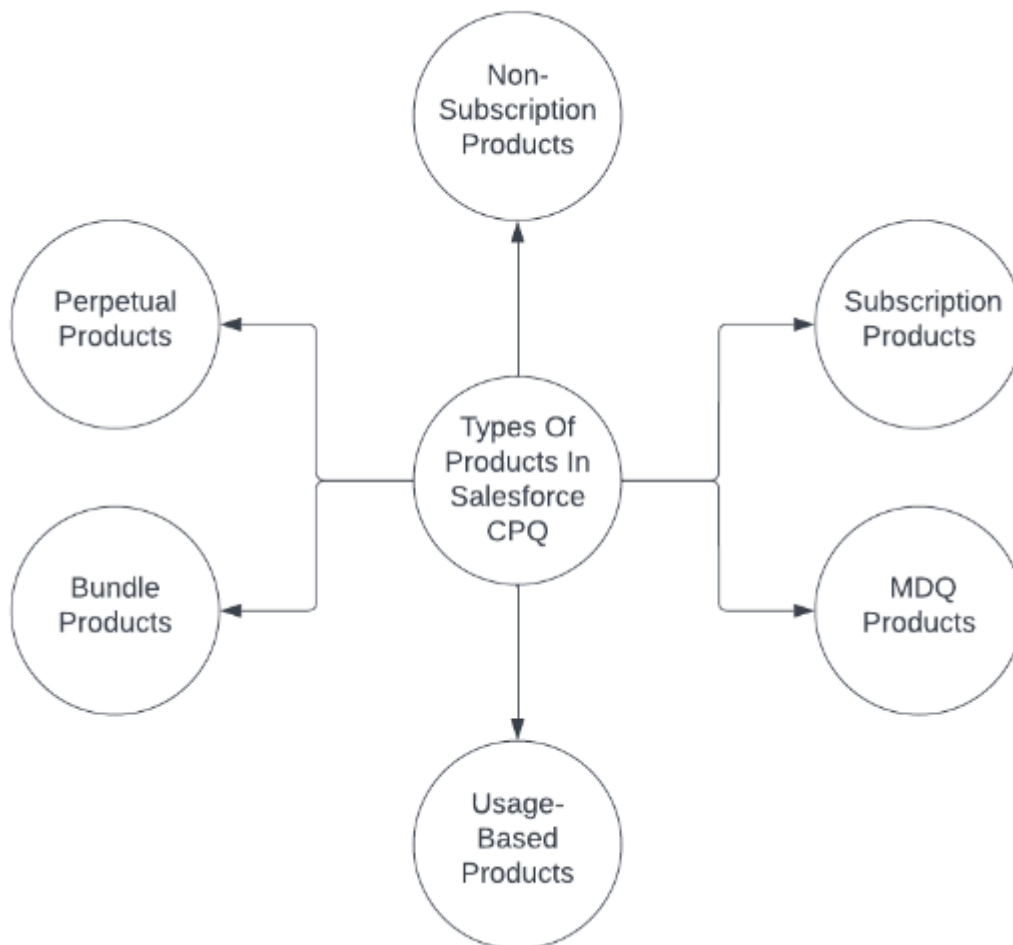
When selling a service, it's important to highlight what makes service personal and how we can meet the customers' needs. Marketing services requires building trusting relationships with customers and customizing them as necessary.

[Example: Urban Clap come home and gives variety of services.](#)



[Salesforce CPQ by Smriti Sharan](#)

Type of Product in Salesforce CPQ?





[Salesforce CPQ by Smriti Sharan](#)

What is Non-Subscription / One-Time Products?



These are products you pay for once, without future payments. For instance, if you buy [a laptop for \\$1000](#), you pay the full amount upfront.

There's no need to renew the purchase or pay for it again in the future.

After purchase, it might be tracked for warranty or other purposes, but you own it outright after the single payment.



[Salesforce CPQ by Smriti Sharan](#)

Explain Product Bundle in CPQ?

There are three key objects that define **bundle structure**:

- Product
- Options
- Features

Product

Bundles are made of products. Products can be referenced in a bundle either as a **bundle parent** or as **product options**. Think of Laptop as product

Product options

They are like [children that sit under the bundle parent](#), and they are what really brings the bundle together because bundles are created when you add Options to a Product in the Options related list. [Product options for laptop](#) can be mouse, keyboard, charger, Microsoft software.



[Salesforce CPQ by Smriti Sharan](#)

Features

They are categories of product options within the configuration. They are **not mandatory**, but they can be used for a **better visual experience**, and they also help drive **additional logic such as Min/Max Options**. Features are like putting options inside features e.g.:

Software

- Microsoft Software

Hardware

- Mouse
- Keyboard

Here Hardware and Software are features which categorize product options.

There are **three types of bundles**:



[Salesforce CPQ by Smriti Sharan](#)

Static bundle. This would be the [traditional, pre-packaged, fixed bundle](#) where you sell certain products together at a fixed price, and the user doesn't need to go into the configuration because no adjustments are allowed.

Configurable Bundle. This type [of bundle can be configured with certain restrictions to avoid impossible configurations.](#) The user can select different options to customize the bundle to meet the customer's needs.

Nested Bundle. These are bundles within other bundles. It's recommended to keep the [nested bundles on to three levels deep.](#) Nested bundles are easily created by adding the parent product of a bundle as an option to another bundle.

Virtual Bundle. This type of bundle [works as a container for other products.](#) To set up a virtual bundle parent, it should be associated with a price of \$0.00.



[Salesforce CPQ by Smriti Sharan](#)

Dynamic Bundles. These functions in conjunction with product filter rules let your sales reps' select products from a pre-filtered list of options they are allowed to choose from.

What are Features in CPQ?

The feature object is related to [products](#), and it "sits" within bundles.


A bundle can have multiple Features or none, and a Feature can have multiple Options.

Features serve two purposes:
















- They offer a **better user experience** as they're navigating through product options within the bundles.
- The fields on the feature records help us **drive business logic**, like setting the **minimum** and **maximum options**, preventing the user from selecting too few or too many options. If the user fails to meet the min/ max requirements and tries to save the configuration.



[Salesforce CPQ by Smriti Sharan](#)

 Product Feature
Cables

Related **Details**

Feature Name Cables 	Number  30
Min Options  0 	Configured SKU  6" Smartphone
Max Options  1 	Option Selection Method 
Category  	Dynamic Option Filter Field Set 
Discount Schedule  	Dynamic Option Lookup Field Set 
	Configuration Field Set 



[Salesforce CPQ by Smriti Sharan](#)

Q-00002 **Configure Products** ▼ Apply Rules Cancel Save

6" Smartphone

Chargers
Choose at least 1 but no more than 5 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input checked="" type="checkbox"/> 1.00	SMARTPHONECHARGER	Smartphone Charger		€15.00

Service
Choose no more than 3 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input checked="" type="checkbox"/> 1.00	SMARTPHONEACTIVATION	Smartphone Activation		€50.00

Cables

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input type="radio"/> 1.00	USBCCHARGE1M	USB-C Charge Cable 1m		€20.00
<input checked="" type="radio"/> 1.00	USBCCHARGE2M	USB-C Charge Cable 2m		€30.00
<input type="radio"/> 1.00	USBCTOUSB	USB-C to USB Adapter		€20.00

The Number field sets the display order of the features, the feature with the lowest number coming first on the list. It is recommended to use increments of 10 when assigning numbers.

Configured SKU field is automatically populated when the user creates a feature from the bundle product's detail page, and it represents the parent product that is housing the feature.



[Salesforce CPQ by Smriti Sharan](#)

Explain different values of option selection method field in features object?

The Option Selection Method field controls how the options attached to this feature appear to the user.

Click – options appear as a list with checkboxes next to them:

Product Feature
Cables

Related **Details**

Feature Name Cables	Number ⓘ 30
Min Options ⓘ 1	Configured SKU ⓘ 6" Smartphone
Max Options ⓘ 5	Option Selection Method ⓘ Click
Category ⓘ	Dynamic Option Filter Field Set ⓘ
Discount Schedule ⓘ	Dynamic Option Lookup Field Set ⓘ
	Configuration Field Set ⓘ



[Salesforce CPQ by Smriti Sharan](#)

Q-00002 **Configure Products** ▼ Apply Rules Cancel Save

6" Smartphone

Chargers
Choose at least 1 but no more than 5 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input checked="" type="checkbox"/> 1.00	SMARTPHONECHARGER	Smartphone Charger		€15.00

Service
Choose no more than 3 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input checked="" type="checkbox"/> 1.00	SMARTPHONEACTIVATION	Smartphone Activation		€50.00

Cables

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input type="radio"/> 1.00	USBCCHARGE1M	USB-C Charge Cable 1m		€20.00
<input type="radio"/> 1.00	USBCCHARGE2M	USB-C Charge Cable 2m		€30.00
<input type="radio"/> 1.00	USBCTOUSB	USB-C to USB Adapter		€20.00

Add – the options collapse, and an Add Options button appears instead.

The Add Options button should be clicked to navigate to a separate page that shows the products for this feature.



[Salesforce CPQ by Smriti Sharan](#)

Product Feature **Cables**

Related	Details
Feature Name Cables	Number 30
Min Options 1	Configured SKU 6" Smartphone
Max Options 5	Option Selection Method Add
Category	Dynamic Option Filter Field Set
Discount Schedule	Dynamic Option Lookup Field Set
	Configuration Field Set



[Salesforce CPQ by Smriti Sharan](#)

Q-00002 **Configure Products** ▼ Apply Rules Cancel Save

6" Smartphone

Chargers
Choose at least 1 but no more than 5 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input checked="" type="checkbox"/> 1.00	SMARTPHONECHARGER	Smartphone Charger		€15.00

Service
Choose no more than 3 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input checked="" type="checkbox"/> 1.00	SMARTPHONEACTIVATION	Smartphone Activation		€50.00

Cables
Choose at least 1 but no more than 5 of the following:

QUANTITY	PRODUCT CODE	PRODUCT NAME	PRODUCT DESCRIPTION	UNIT PRICE
<input type="checkbox"/> 1.00	USBCCHARGE1M	USB-C Charge Cable 1m		€20.00

[Add Options](#)

Dynamic – renders an Add Options button, which brings up a pre-filtered list of products available for selection. To use this functionality, you should set up a Salesforce CPQ Filter Product Rule for Dynamic Bundles.



[Salesforce CPQ by Smriti Sharan](#)

Product Feature
Cables

Related **Details**

Feature Name Cables	Number i 30
Min Options i 1	Configured SKU i 6" Smartphone
Max Options i 5	Option Selection Method i Dynamic
Category i	Dynamic Option Filter Field Set i
Discount Schedule i	Dynamic Option Lookup Field Set i
	Configuration Field Set i

Q-00002 **Configure Products** 3 Add Cancel

Cables
Choose at least 1 but no more than 5 of the following:

<input type="checkbox"/> PRODUCT CODE	PRODUCT NAME	PRODUCT FAMILY	PRODUCT DESCRIPTION
<input type="checkbox"/> HEADPHONEEXT	Headphone Extension Cable	Miscellaneous	
<input type="checkbox"/> USBCCHARGE1M	USB-C Charge Cable 1m	Miscellaneous	
<input type="checkbox"/> USBCCHARGE2M	USB-C Charge Cable 2m	Miscellaneous	

Filter ×

Product Code

Product Name

Product Family

Apply Clear Fields



[Salesforce CPQ by Smriti Sharan](#)

How to setup Subscription Products in Salesforce CPQ?

There are three essential fields for setting up Subscription Products in Salesforce CPQ.

Product Field	Data Type	Description
Subscription Pricing	Picklist	The product would not be considered as a Subscription without this field being set to either Fixed Price or Percent of Total.
Subscription Term	Number	This is the period associated with the purchase, so you can set it to 12 months if it is an Annual Subscription.



[Salesforce CPQ by Smriti Sharan](#)

<p>Subscription Type</p>	<p>Picklist</p>	<p>This field determines whether the product is Renewable or One-Time. Renewable Subscriptions transfer to Renewal Quotes. One-Time Subscriptions will not.</p>
---------------------------------	-----------------	---

To create **Subscription Products**, create new products Record. Fill the fields – **Subscription Pricing**, **Subscription Type**, and **Subscription Term** for the Product. Note that during contract creation, Subscription Products convert to Subscription records.



[Salesforce CPQ by Smriti Sharan](#)

New Product

Product Information

Product Name Active

Product Code Product Family

Salesforce CPQ Pricing

Pricing Method Optional

Pricing Method Editable Hide Price In Search Results

Price Editable Percent Of Total (%)

Subscription Pricing

Subscription Term

Subscription Type

Percent Of Total Base

Percent Of Total Category

Percent Of Total Constraint

What is Salesforce Guided Selling?

Guided selling implemented in CPQ is a tool that asks questions about the needs of the customer based on the product selection available. It is done



[Salesforce CPQ by Smriti Sharan](#)

by creating a prompt that asks sales reps about the types of products they want to add to the quote and their specifications.

What is Search Filter?

sales reps have a problem that the [large catalog of products](#) makes it hard to find specific products on the Product Selection page so for that **Search filter** comes in handy.

The [Product Code](#), [Product Name](#), and [Product Family](#) fields are already in that field set. Any custom field you create can also be added to the Search Filters field set.

A screenshot of the Salesforce Product Selection interface. At the top, there's a search bar labeled 'Search Products' and a '156' notification badge. Below the search bar is a table with columns: PRODUCT CODE, PRODUCT NAME, PRODUCT FAMILY, and PRODUCT DESCRIPTION. The table lists various products like '10KWHBATTERY', '2FAUSBKEY', '4KVIDEOCAM', etc. On the right side, a 'Product Filter' overlay is visible, containing input fields for 'Product Code', 'Product Name', a dropdown for 'Product Family', and another dropdown for 'Subscription Pricing'. There are 'Apply' and 'Clear Fields' buttons at the bottom of the filter overlay.

Search filter can only search on product, asset n subscriptions.



[Salesforce CPQ by Smriti Sharan](#)

To create search filter on product> Create new Search filter record-
>choose target object, field, conditions.

Edit Bundles

* = Required Information

* Filter Name	<input type="text" value="Bundles"/>	* Display Order ⓘ	<input type="text" value="1"/>
* Target Object ⓘ	<div style="border: 1px solid black; padding: 2px;"><div style="background-color: #f0f0f0; padding: 2px;">Product ▼</div><div style="padding: 2px;">--None--</div><div style="background-color: #e0e0e0; padding: 2px; margin: 2px 0;">✓ Product</div><div style="padding: 2px;">Asset</div><div style="padding: 2px;">Subscription</div></div>	Filter Value ⓘ	<input type="text" value="ed"/>
		Hidden ⓘ	<input checked="" type="checkbox"/>
		Action ⓘ	<div style="border: 1px solid #ccc; padding: 2px; display: flex; align-items: center;">+ Add Bundles Prefilter ×</div>

How to hide a product from product selection page?

Create search filter record with following value:

-Product Name

-Target object –Product



[Salesforce CPQ by Smriti Sharan](#)

-Target Field – Field you want to hide

-Filter Value – False – Since this time [you don't want sales reps to control the filtering](#), tell CPQ that the value to filter for is “false”, which is another way of saying “unchecked.”

-Hidden – True – You've preloaded the “false” value for filtering, but we need to make sure sales reps can't undo your work. So, make the filter hidden from view (and tampering). It'll still be in effect even though you can't see it.



[Salesforce CPQ by Smriti Sharan](#)

Edit Retired

* = Required Information

* Filter Name	Retired	* Display Order	1
* Target Object	Product	Filter Value	false
* Target Field	Retired__c	Hidden	<input checked="" type="checkbox"/>
* Operator	equals	Action	Search Custom Actions...

Advanced Information

Filter Source Object	Hidden Source Object
--None--	--None--

Cancel Save & New Save

How to show a product based on a particular region?

-The Region multi-select picklist on the product record identifies region-exclusive products.

-You create a Region field on the user object to capture the sales rep's region.



[Salesforce CPQ by Smriti Sharan](#)

-You create a Region formula on the quote object to pull and format the user's region information.

-You create a search filter that compares the product region with the formatted region, and made it hidden.

Edit 5G Antenna

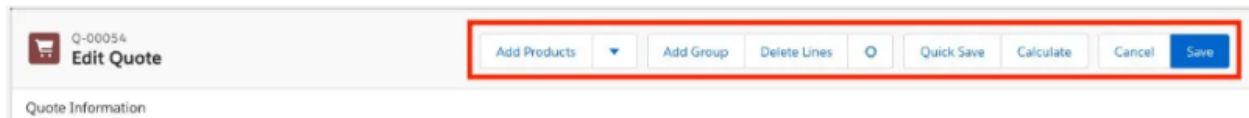
Product Name <input type="text" value="5G Antenna"/>	Active <input checked="" type="checkbox"/>						
Product Code <input type="text" value="5GANTENNA"/>	Product Family <input type="text" value="Hardware"/>						
Retired <input type="checkbox"/>	Region						
	<table border="0"><tr><td>Available</td><td>Chosen</td></tr><tr><td><input type="text" value="North"/></td><td><input type="text" value="East"/></td></tr><tr><td><input type="text" value="South"/></td><td><input type="text" value="West"/></td></tr></table>	Available	Chosen	<input type="text" value="North"/>	<input type="text" value="East"/>	<input type="text" value="South"/>	<input type="text" value="West"/>
Available	Chosen						
<input type="text" value="North"/>	<input type="text" value="East"/>						
<input type="text" value="South"/>	<input type="text" value="West"/>						

What is custom Actions?

Custom actions are part of the CPQ toolbox. They are of three Type -
button, menu, separator



[Salesforce CPQ by Smriti Sharan](#)



Check Brand Button on custom action – This makes your button stand out by coloring it blue instead of white.

New Custom Action

* = Required Information

Information

<p>* Name <input type="text" value="Add Bundles Prefilter"/></p>	<p>Active <input checked="" type="checkbox"/></p>
<p>* Display Order <input type="text" value="10"/></p>	<p>Default <input type="checkbox"/></p>
<p>Type <input type="text" value="Button"/></p> <ul style="list-style-type: none">--None--✓ ButtonMenuSeparator	<p>Brand Button <input checked="" type="checkbox"/></p>
	<p>URL Target <input type="text" value="--None--"/></p>
	<p>URL <input type="text"/></p>

Layout

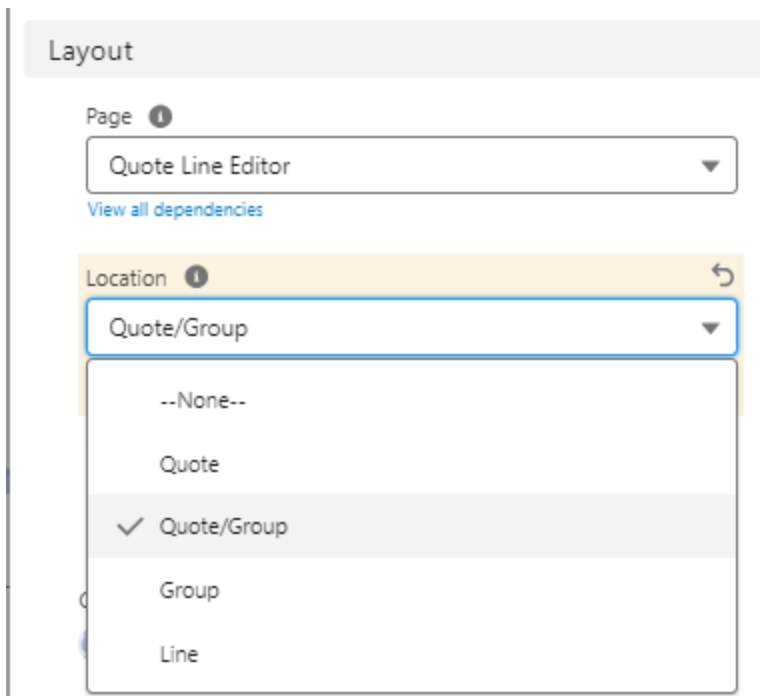


[Salesforce CPQ by Smriti Sharan](#)

Location-

If you use groups, this makes your button appear in every quote line group.

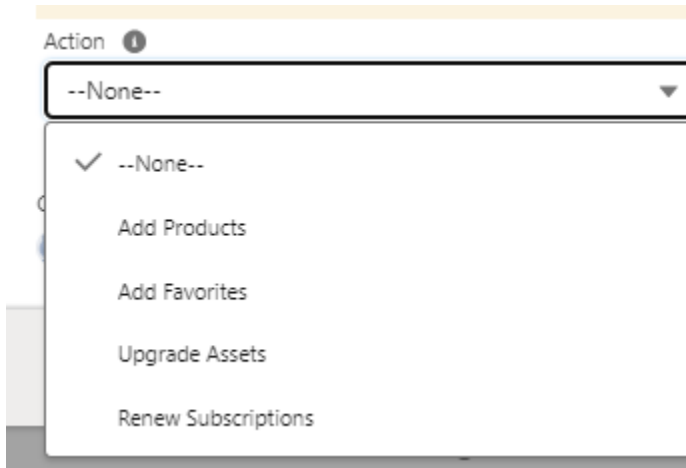
Otherwise, it appears in the quote header.



Action -this tells CPQ to what action to take. Like 'Add product' takes user to product selection page.



[Salesforce CPQ by Smriti Sharan](#)





[Salesforce CPQ by Smriti Sharan](#)

Price Book

In CPQ



[Salesforce CPQ by Smriti Sharan](#)

What is Standard Price Book?

In the “Standard Price Book,” we list the [regular prices for the product](#). These were the prices that most customers paid when they visited her shop.

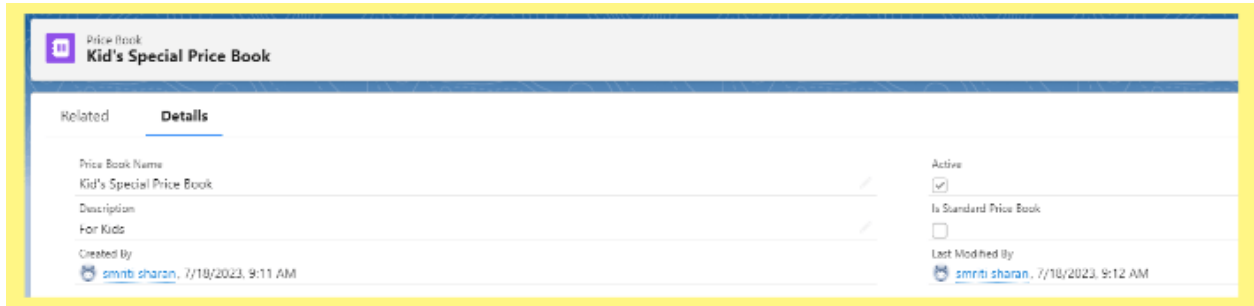


What is Custom Price Book?

It is the pricebook created to [provide special prices to different customers based on various scenarios](#). Example, price can vary based on region, country etc so there will be different pricebook for each region to sell that product for that price in that specific region.

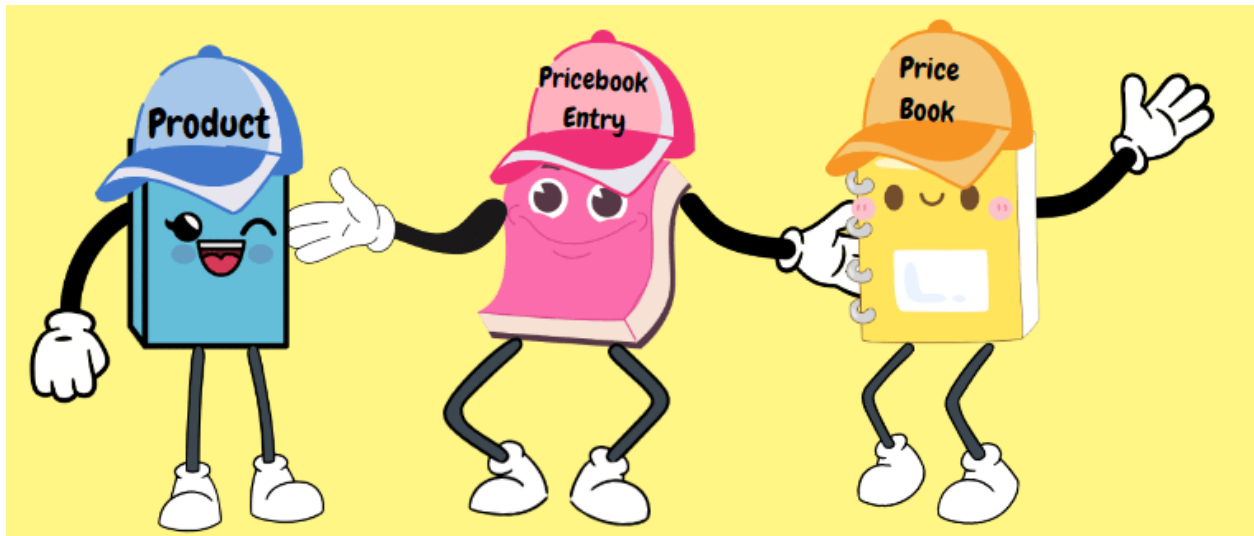


[Salesforce CPQ by Smriti Sharan](#)



What is Pricebook Entry?

To connect the products with their respective price books we create price book entries. You can think PriceBook Entry is the common friend which connects Product and Pricebook together.





[Salesforce CPQ by Smriti Sharan](#)

What is the link between product, price book and pricebook entry?

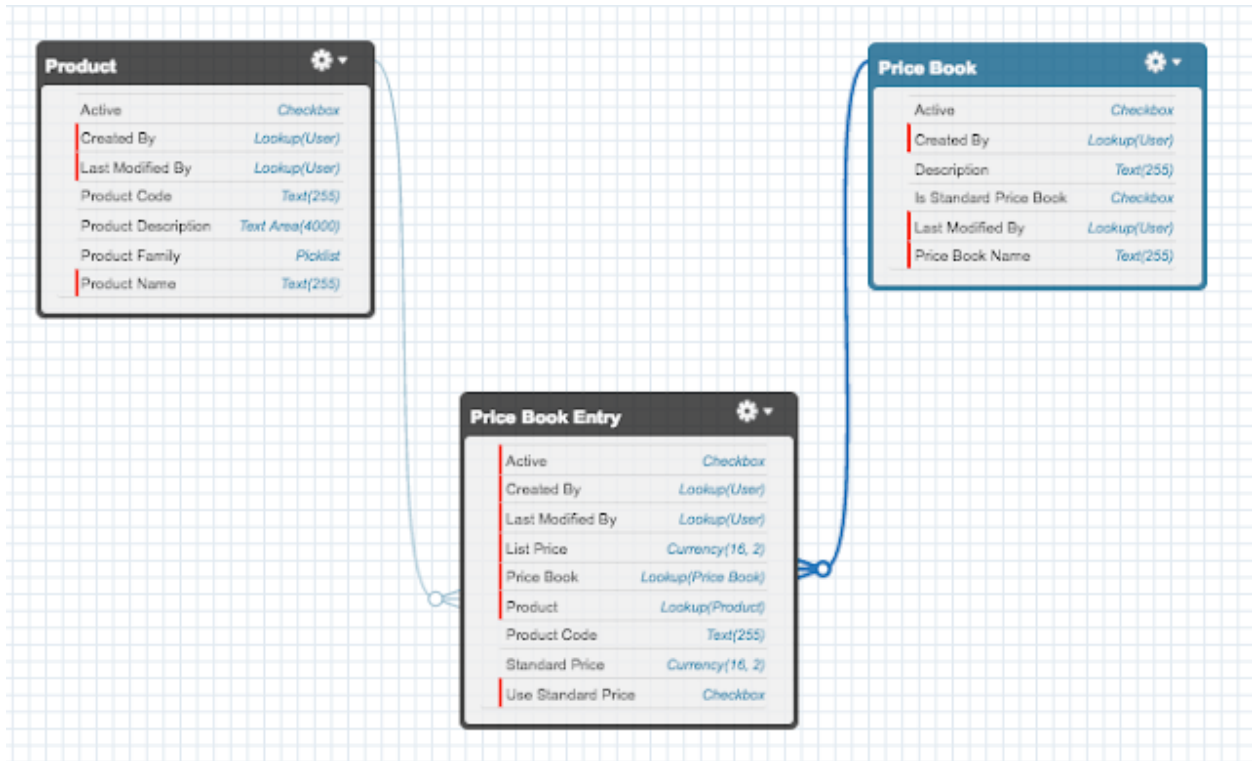
- Products are represented by Standard Object – **Product2**.
- PriceBooks are represented by Standard Object – **Pricebook2**

Data Model:

- **PricebookEntry** is the **junction object** between **Product** and **Price Book**.



[Salesforce CPQ by Smriti Sharan](#)



How do we define price?

We manage the price of product using price book. There are two types of price book:

- Standard Price Book
- Custom Price Book

Standard price book



[Salesforce CPQ by Smriti Sharan](#)

Every product that we create, we should have a standard price.

It is the default price that is associated.

Go to Price book Object -> Click on New and Create Standard Price book

Select checkbox besides Standard Price Book and Active

A screenshot of the Salesforce 'Standard Price Book' details page. The page has a header with 'Price Book Standard Price Book' and buttons for 'Edit' and 'Clone'. Below the header, there are two tabs: 'Related' and 'Details'. The 'Details' tab is active. The details are organized into two columns. The left column contains: 'Price Book Name' (Standard Price Book), 'Description', and 'Created By' (Smriti Sharan, 8/14/2018 6:47 pm). The right column contains: 'Active' (checked), 'Is Standard Price Book' (checked), and 'Last Modified By' (Smriti Sharan, 9/15/2021 4:52 pm).

Go to Related -> In the Price book -> Select Add Standard Price

A screenshot of the 'Related' section in Salesforce. It shows a list titled 'Price Books (0)'. At the top right of the list, there is a button labeled 'Add Standard Price'.



[Salesforce CPQ by Smriti Sharan](#)

New Price Book Entry

* Product	Active
<input type="text" value="Aesthetica Cosmetics Contour Kit"/>	<input checked="" type="checkbox"/>
* Price Book	Product Code
<input type="text" value="Standard Price Book"/>	
* List Price	Currency
<input type="text" value="30.00"/>	<input type="text" value="USD - U.S. Dollar"/>
Use Standard Price	
<input type="checkbox"/>	

So, we have created our product and we have set the standard price.

Custom price books

When we offer our products to different people with different prices, then we go for custom price books.



[Salesforce CPQ by Smriti Sharan](#)

For example, I am selling iPhone in India and US. The price will differ for iPhone in India and US market. So, there comes the concept of custom price book.

Go to Pricebook -> Click New -> Create the price book

Price Book
US

Related **Details**

Price Book Name US	Active <input checked="" type="checkbox"/>
Description	Is Standard Price Book <input type="checkbox"/>
Created By Smriti Sharan, 9/15/2021 4:30 pm	Last Modified By Smriti Sharan, 9/15/2021 4:30 pm

Price Book
India

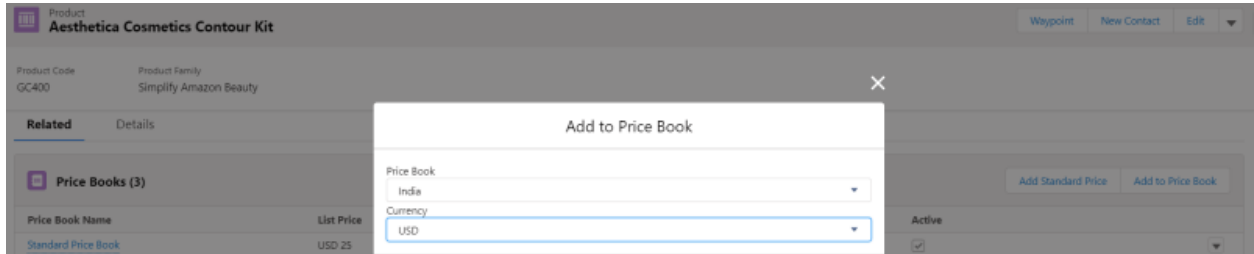
Related **Details**

Price Book Name India	Active <input checked="" type="checkbox"/>
Description	Is Standard Price Book <input type="checkbox"/>
Created By Smriti Sharan, 9/2/2021 5:12 am	Last Modified By Smriti Sharan, 9/15/2021 4:30 pm

Go to Product -> Related -> Click on Add to Pricebook



[Salesforce CPQ by Smriti Sharan](#)



Add the List Price \$30 for US and \$25 for India

New Price Book Entry

* Product	<input type="text" value="Aesthetica Cosmetics Contour Kit"/>	Active	<input checked="" type="checkbox"/>
* Price Book	<input type="text" value="US"/>	Product Code	<input type="text"/>
* List Price	<input type="text" value="30"/>	Currency	<input type="text" value="USD - U.S. Dollar"/>
Use Standard Price	<input type="checkbox"/> USD 25		

What is OWD Setting for Pricebook Object?



[Salesforce CPQ by Smriti Sharan](#)

If we go to sharing settings, go for price book, and we will see that by default setting is set to Use.

SETUP Sharing Settings

Default Sharing Settings

Organization-Wide Defaults Edit Organization-Wide Defaults Help ?

Object	Default Internal Access	Default External Access	Grant Access Using Hierarchies
Lead	Public Read/Write/Transfer	Public Read/Write/Transfer	✓
Account and Contract	Public Read/Write	Private	✓
Contact	Controlled by Parent	Controlled by Parent	✓
Order	Controlled by Parent	Controlled by Parent	✓
Asset	Controlled by Parent	Controlled by Parent	✓
Opportunity	Private	Private	✓
Quote	Controlled by Parent	Controlled by Parent	✓
Case	Public Read/Write/Transfer	Private	✓
Campaign	Public Full Access	Public Full Access	✓
Campaign Member	Controlled by Campaign	Controlled by Campaign	✓
User	Public Read Only	Private	✓
Activity	Private	Private	✓
Calendar	Hide Details and Add Events	Hide Details and Add Events	✓
Price Book	Use	Use	<input checked="" type="checkbox"/>

Price Book Access Levels

FIELD	DESCRIPTION
Use	All users can view price books and add them to opportunities. Users can add any product within that price book to an opportunity.
View Only	All users can view and report on price books but only users with the "Edit" permission on opportunities or users that have been manually granted use access to the price book can add them to opportunities.
No Access	Users cannot see price books or add them to opportunities. Use this access level in your organization-wide default if you want only selected users to access selected price books. Then, manually share the appropriate price books with the appropriate users.

How to use Price Books to sell products in multiple currencies?



[Salesforce CPQ by Smriti Sharan](#)

If a company has a single price for a product regardless the currency, it can enable the multicurrency feature in Salesforce. Then, the conversion rates entered in Salesforce will automatically apply to product prices added to an Opportunity.

Can we delete Standard Price book?

A company can deactivate Standard Price Book, but not delete or archive it. In case a company changes prices in Standard Price Book, it may affect open and even closed Opportunities.

Can we delete Custom Price book in Salesforce?

It's not possible to delete Price Book while it's in use on Opportunities or Quotes.

What is the means to archive a price book?



[Salesforce CPQ by Smriti Sharan](#)

If a company decides that it **won't need a certain Price Book in the future**, it can archive a Price Book even if it's being used in Opportunities or Quotes.

Once archived Price Books can't be recovered.

How can we determine which price books are assigned to a user?

By default, the org-wide default setting for price book is "Use". This means **that any user with read access to both the "Product" and "Price Book"** object will be able to select any active price book.

How to control access to Salesforce Price Books?

A company can control access to Price Books by setting the **Organization-wide sharing rule on Price Books to "View only" or "No Access"** and then giving access to a particular Price Book based on the Role, Public Group.

How to set access Level of Product?



[Salesforce CPQ by Smriti Sharan](#)

The products object does not use the standard record sharing model. Access to manipulated products is controlled by object-level security (profile & permission sets).



[Salesforce CPQ by Smriti Sharan](#)

How to modify prices in Price Books?

User can modify prices in both Standard and Custom Price Books. This can be achieved in two ways:

- Manually
- Using Apex Data Loader



[Salesforce CPQ by Smriti Sharan](#)

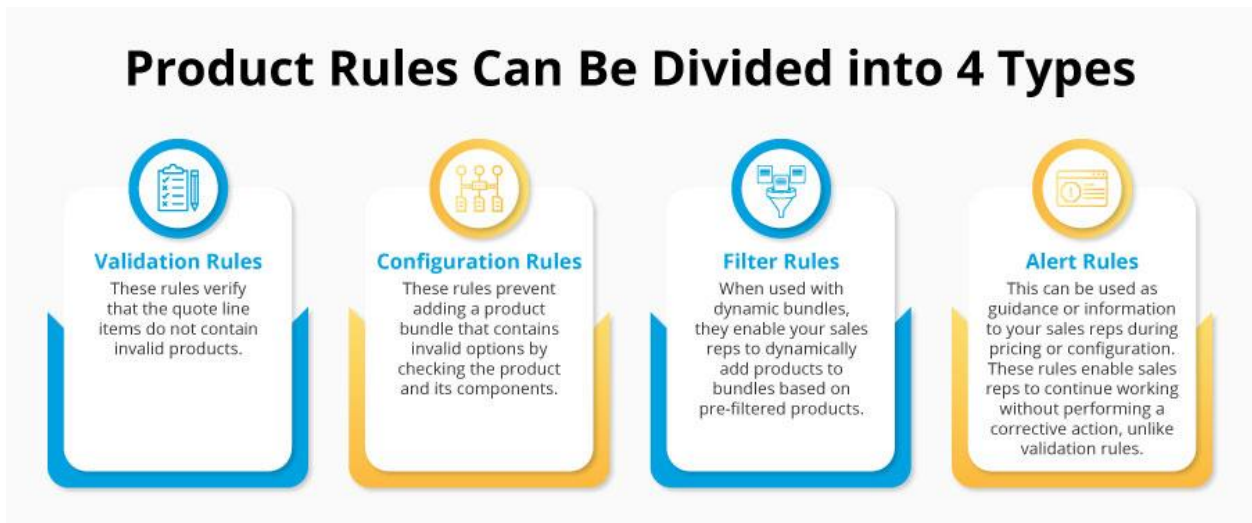
Product Rule in CPQ



[Salesforce CPQ by Smriti Sharan](#)

What are product rules in Salesforce CPQ?

There are four types in product rules:



- Validation
- Alert
- Selection
- Filter

1. Validation

Validation is an option in product rules which **shows an error message and does not allow a save until we have taken action to correct the mistake** according to the error message which is shown by the validation rule, we created. The validation rule checks the error condition if it returns true then gives an error message. We can only save it after correcting that error. To



[Salesforce CPQ by Smriti Sharan](#)

create a validation rule we use summary variable. We'll discuss further what summary variables are in CPQ.

2. Alert

When we select Alert type while creating a Product Rule and when we select that product and try to save it if the error condition returns true then it shows an alert message to suggest the best option. Whenever we try to save, [it shows two options that are canceled and continue](#). We can change our product according to alert suggestions or we can also continue with the selected product based on the requirement. [It just suggests to us through an alert message when the error condition returns true.](#)

3. Selection

[It automatically performs actions on Product Options without showing any kind of message to users.](#) In other words, we can also say that it sets rules to automatically add, delete, hide, show, enable, or disable product options.



[Salesforce CPQ by Smriti Sharan](#)

4. Filter

The filter rule is used to filter the product and shows only available products to add to the bundle. To create a filter rule we select the type field as Filter under the Product Rule. This rule is used to configure a bundle. This rule dynamically updates the bundle of which products we are offering to the customer.

What is the difference between option constraints and product rules?

Option Constraints and Product Rules have different use cases.

Option Constraints cannot select products automatically. (although they can deselect products when dependency applies). They are good at two scenarios:

- dependency
- exclusion



[Salesforce CPQ by Smriti Sharan](#)

option constraints allow **for real-time processing**. Whenever the user selects or deselects an option, the dependency, or the exclusion fires immediately.

Product Rules has more features than option constraint, though it is required to choose an evaluation event to specify when Salesforce CPQ should consider and run a product rule: [Load](#), [Edit](#), [Save](#), or [Always](#).

Explain lookup object field?

Product rules and price rules in Salesforce CPQ contain the Lookup Object field. This field defines the object that stores data you can evaluate through lookup queries to compare against your quote, quote line, or product option fields.

How to use option constraints?

There are lots of scenarios when your sales reps must enable or disable a product in a Bundle based on another product option.



[Salesforce CPQ by Smriti Sharan](#)

scenarios include:

1. A. Product A becomes AVAILABLE if Product B is selected.
2. B. Product A is NOT AVAILABLE if Product B is selected.

This can be achieved by creating option constraints from the Option Constraints related list on the bundle parent product record.

Create an Option Constraint so that Product A may be selected only when Product B is selected.

You have a Scenario where sales representative must sell a bundle in which there are at least as many Product B's as Product A's. How to setup for this scenario?

Step 1: Create a Summary Variable to determine the aggregated quantity of Product A in configuration.

Step 2: Create a Summary Variable to determine the aggregated quantity of Product B in a configuration.



[Salesforce CPQ by Smriti Sharan](#)

Step 3: Create a Product Rule that confirms how the validation rule will function during product configuration.

Step 4: Create an Error Condition that controls when the rule fires, that is only when Product B Summary Variable is less than Product A Summary Variable.

Step 5: Create a Configuration Rule to show that the validation rule will apply to the Product bundle.

Step 6: Check whether the rule behaves as expected.

What are Summary Variables in Product Rules?

Summary variables are used to determine the [aggregations of products](#) on quote. It is also known as the tested variable or filter variable.

There are five aggregate functions we can perform for the aggregation while creating a summary variable.

Sum: It returns the sum of all aggregate field's values.



[Salesforce CPQ by Smriti Sharan](#)

Average: Returns the average of all aggregate field's values.

Min: Returns the minimum of among all the aggregate field's values.

Max: Returns the maximum of among all the aggregate field's values.

Count: It counts the number of records that satisfy the condition of filter criteria.



[Salesforce CPQ by Smriti Sharan](#)

Price Rule in CPQ



[Salesforce CPQ by Smriti Sharan](#)

What is the price rule?



Imagine [you have a candy store](#) with different types of candies like lollipops, gummy bears, and chocolate bars. Each candy has a different price. Now, let's talk about the Price Rule:

[The Price Rule is like a special rule you create for your candy store.](#) It helps you decide how to set the prices or apply discounts to different candies based on certain conditions.

For example, let's say you want to have a "Buy One, Get One Free" offer for gummy bears candy. You can create a Price Rule that says whenever



[Salesforce CPQ by Smriti Sharan](#)

someone buys one packet of gummy bears, they get another packet for free. It's like a special deal just for gummy bear lovers!

What is the difference between product rule and price rule?

The overall structure of price rules and product rules is the same:

- Rule Record – Which contains general properties.
- Conditions – Both support Advanced Condition Logic
- Actions

Product Rules

- Product Rules are used to enforce business logic, help automate feedback to sales reps about product selections and they also accelerate the quoting process.

There are 4 types of Product Rules:



[Salesforce CPQ by Smriti Sharan](#)

- **Validation Rules:** When the conditions are not met, an error message is displayed, and the users can't continue until the situation is fixed. So, they act as a **hard stop**.
- **Alert Rules** are a **soft stop** as compared to the Validation Rules' hard stop. They also display a message, but it's more as a suggestion because the user can proceed without meeting the specified business requirements.
- **Selection Rules:** can **add, remove, show, hide, enable, or disable** or perform some combination of these actions on product options within a bundle or they can add a product to a quote. This rule can be a great way to leverage Guided Selling to guide your reps to the appropriate or most desired options by automatically selecting the right products.
- **Filter Rules** are used in bundles to pull in product options from the product catalog using a **filter** rule. This is often referred to as creating a dynamic bundle.



[Salesforce CPQ by Smriti Sharan](#)

Price Rules

- **Price Rules** can be considered to **make field updates** within the calculation that will affect the pricing. you can use them to set discount and quantities.

What is the structure of the Price Rule?

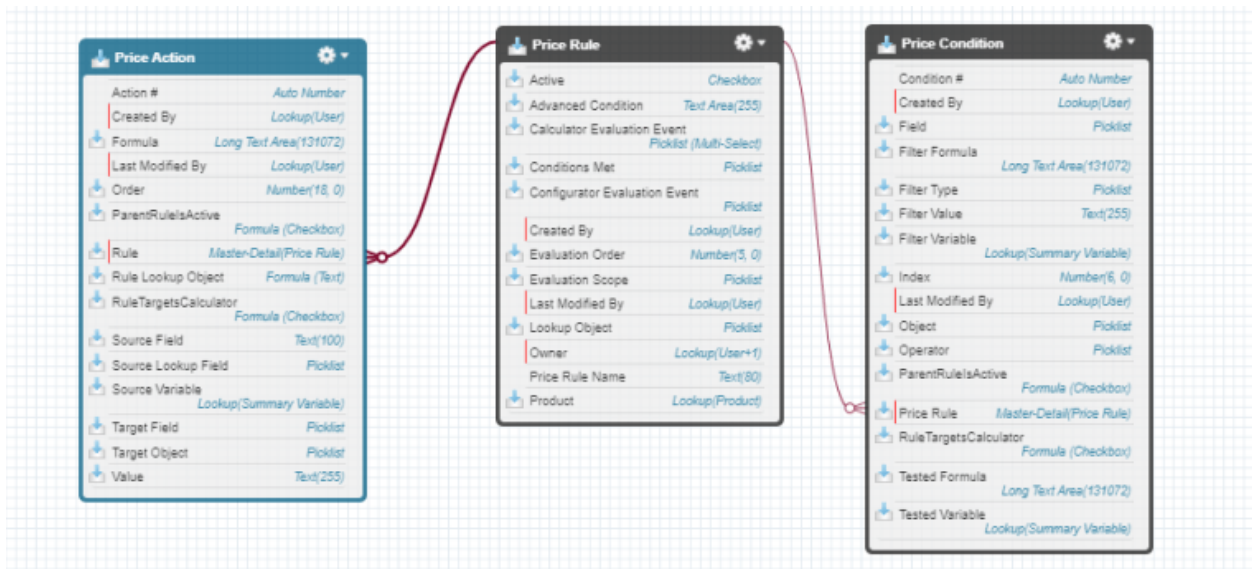
The price rule included three related records:

1. Price Rule
2. Price Condition
3. Price Action



[Salesforce CPQ by Smriti Sharan](#)

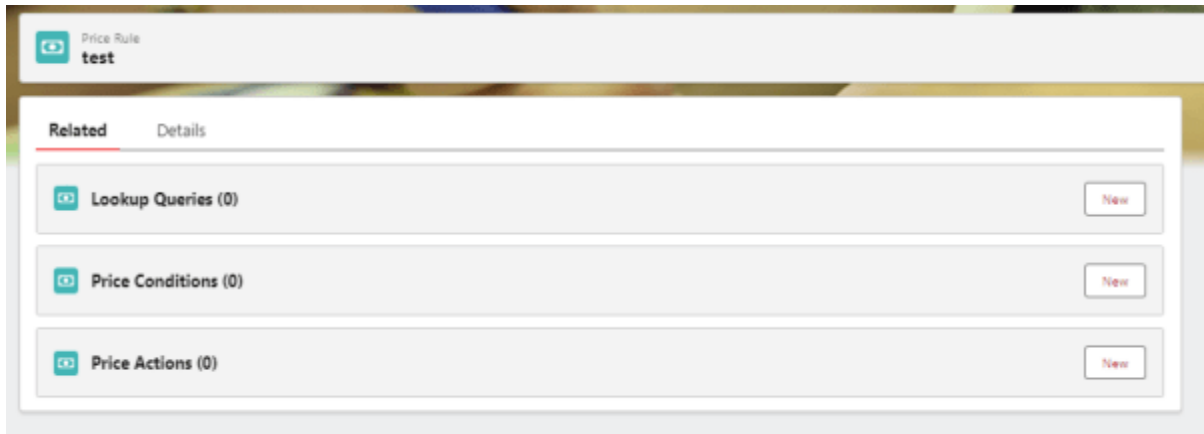
The price Rule has a Master-Detail relationship with Price Condition and Price Action.



By connecting the Price Rule with the relevant child Price Conditions, the pricing system ensures that the appropriate rules are applied based on specific conditions. Price Action defines the actions that need to be taken when the associated Price Conditions are met. It specifies how the pricing or other field values should be modified based on the rule criteria



[Salesforce CPQ by Smriti Sharan](#)



1. Price Rule: The price rule contains all the specific rules for pricing in the candy store. It tells the store how to handle different situations when setting candy prices.
2. Price Condition: There's someone in the store who checks the type of candy you want to buy. They make sure it meets certain conditions, like being a "Gummy Bear" or having a specific flavor.
3. Price Action: Once the candy passes the inspection, there's someone who adjusts the price based on the price rules. They might give a discount, change the quantity, or do something special to make the price more appealing.



[Salesforce CPQ by Smriti Sharan](#)

Fields in Price Rule

New Price Rule

* = Required Information

Information

* Price Rule Name
Complete this field.

Active

* Evaluation Scope

Evaluation Order

* Conditions Met

Advanced Condition

Calculator Evaluation Event

Available	Chosen
On Initialization	On Calculate
Before Calculate	
After Calculate	

Configurator Evaluation Event

Product

Lookup Object



[Salesforce CPQ by Smriti Sharan](#)

Difference between Calculator and Configurator Evaluation Scope in price rule in Salesforce CPQ?

*Evaluation Scope ⓘ

Calculator ▼

--None--

Configurator

✓ Calculator

The **Evaluation Scope** field is like using that magical calculator.





[Salesforce CPQ by Smriti Sharan](#)

When you have price rules with the Calculator Evaluation Scope, they are used by the calculator to decide how much something should cost. It has two picklist values:

- **Calculator Evaluation Scope** – Scope is about using special rules to calculate prices.
- **Configurator Evaluation Scope** – Following rules when you're picking different options.

Let's explain the difference between Calculator and Configurator Evaluation Scopes in Salesforce CPQ with a **real-time example**:

Imagine you're going to a candy store to buy some delicious candies. You have a budget of \$10, and you want to buy different types of candies based on their flavors.



[Salesforce CPQ by Smriti Sharan](#)



Let's explore how Calculator and Configurator Evaluation Scopes come into play:

1. Calculator Evaluation Scope:

- Imagine there's a special offer where **if you buy more than 5 candies, you get a 20% discount on the total price.**
- This offer is an example of a price rule with the Calculator Evaluation Scope. The rule calculates the **discounted price based on the quantity** of candies you're buying.
- So, if you add 7 candies to your cart, the Calculator evaluates the price rule and applies the discount, reducing the total cost.



[Salesforce CPQ by Smriti Sharan](#)

2. Configurator Evaluation Scope:

– Now, let's consider the flavors of candies available in the store:

strawberry, chocolate, and lemon.

– There's a rule in place that **says if you choose strawberry-flavored candies, you cannot select lemon-flavored candies because they don't go well together.**

– This rule is an example of a price rule with the Configurator Evaluation Scope. It **controls the availability of certain options** based on your previous choices.

– If you already have strawberry candies in your cart, the Configurator evaluates the rule and **prevents you from selecting** lemon candies, ensuring you don't end up with an incompatible combination.

In this above example

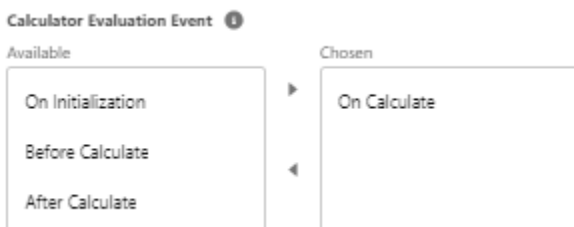


[Salesforce CPQ by Smriti Sharan](#)

The Calculator Evaluation Scope helps calculate the total price of your candies based on special offers, discounts, and quantity.

The Configurator Evaluation Scope ensures that you follow certain rules when selecting different options, such as flavors, to avoid incompatible choices.

Explain calculator evaluation event?



Let's explain the Calculator Evaluation Events of "On Initiate," "On Calculate," "Before Calculate," and "After Calculate" with a real-time example using the candy store scenario:



[Salesforce CPQ by Smriti Sharan](#)

In Brief:

“On Initiate” marks the **beginning of the candy calculation process**.

“On Calculate” performs the **actual price calculation of the candies**.

“Before Calculate” applies **any final checks** or rules just before the calculation.

“After Calculate” allows for **additional actions** or surprises after the prices have been determined.

Let's understand in more detail:

1. On Initiate:

– Suppose you have a price rule that states if the customer is a loyal member, they get a 5% discount on all candy purchases.

– This price rule would have the evaluation event set to “On Initiate.”



[Salesforce CPQ by Smriti Sharan](#)

– When the customer initiates the price calculation process, such as by starting a new quote, the system checks if they are a loyal member and applies the 5% discount to all the candies in their cart from the beginning.

2. On Calculate:

– Imagine you have a price rule that offers a bulk discount if the customer buys more than 10 candies of any flavor.

– This price rule would have the evaluation event set to “On Calculate.”

– As the customer adds candies to their cart, the system continuously calculates the total quantity of candies. When it reaches or exceeds 10, the price rule is triggered, and a bulk discount is applied to all the candies in the cart.

3. Before Calculate:



[Salesforce CPQ by Smriti Sharan](#)

– Let's say you have a price rule that states if the customer has selected candies with a total price over \$30, they receive a complimentary candy bag.

– This price rule would have the evaluation event set to “Before Calculate.”

– Just before the final calculation is performed, the system checks if the total price of the candies exceeds \$30. If it does, the price rule is triggered, and a complimentary candy bag is added to the cart.

4. After Calculate:

– Suppose you have a price rule that gives a special promotional offer of “Buy 2, Get 1 Free” for a specific candy flavor.

– This price rule would have the evaluation event set to “After Calculate.”

– After the final calculation is completed, the system reviews the customer's selected candies and identifies if they qualify for the “Buy 2, Get

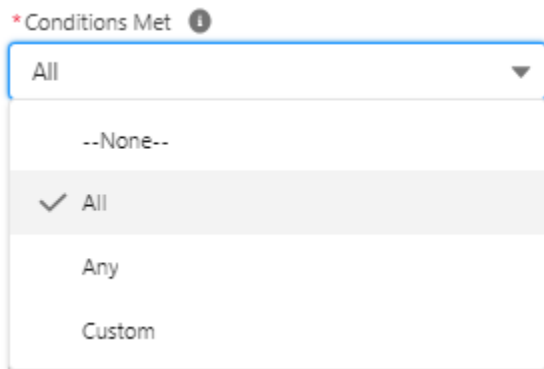


[Salesforce CPQ by Smriti Sharan](#)

1 Free” offer. If they do, the price rule is triggered, and the system adds the free candy to the cart.

By using these different Calculator Evaluation Events, you can apply specific discounts, offers, or modifications at different stages of the calculation process in Salesforce CPQ.

What is the significance of the “Condition Met” field in a Price Rule?



Let’s explain the concepts of “Condition Met – All” and “Condition Met – Any” in using our candy store example:

1. Condition Met – All:



[Salesforce CPQ by Smriti Sharan](#)

– Imagine you have a candy store with a special promotion: If a customer buys at least 5 candies and the total price exceeds \$10, they receive a free candy bag.

– This price rule would have the condition met set to “All.”

– When a customer adds 5 or more candies to their cart and the total price of those candies exceeds \$10, all the conditions are met!

– As a result, the system recognizes that the customer qualifies for the promotion, and a free candy bag is automatically added to their order at checkout.

2. Condition Met – Any:

– Now, consider a different price rule: If a customer purchases any candy with a specific flavor (e.g., “Cherry”), or if they buy more than 10 candies in total, they receive a discount of 20%.

– This price rule would have the condition met set to “Any.”



[Salesforce CPQ by Smriti Sharan](#)

– When a customer either selects a candy with the “Cherry” flavor or adds more than 10 candies to their cart, any one of the conditions is met!

– As a result, the system applies the 20% discount to the qualifying candies, regardless of which condition triggered the discount.

In both scenarios, the “Condition Met – All” requires all specified conditions to be satisfied, while the “Condition Met – Any” only requires at least one condition to be met. These different approaches allow you to create customized pricing strategies based on specific combinations or individual criteria.

Can you explain Lookup Object field in a Price Rule with help of scenario?

Lookup Object ⓘ

In your candy store, you have different types of candies with unique features. The Lookup Object field in the Price Rule object is like a special



[Salesforce CPQ by Smriti Sharan](#)

connection that helps you **apply specific rules to certain types of candies.**

Imagine you have a candy called “Rainbow Lollipop,” and you want to create a special price rule for it. Here’s how it works:

1. Lookup Object Relationship:

– The Lookup Object field is like a bridge that connects the Price Rule to the “Rainbow Lollipop” candy.

– It helps you establish a special relationship between the Price Rule and this specific candy type.

2. Accessing Candy Information:

– Once the lookup relationship is established, you can access information about the “Rainbow Lollipop” candy within the Price Rule.

– You can find out details about the candy, such as its flavor, color



[Salesforce CPQ by Smriti Sharan](#)

3. Applying Special Rules:

- By using the Lookup Object field, you can create special rules that only apply to the “Rainbow Lollipop” candy.
- For example, you can create a rule that offers a discount of 20% for every “Rainbow Lollipop” purchased.

By having this special connection between the Price Rule and the “Rainbow Lollipop” candy through the Lookup Object field, you can create unique and customized rules just for that candy type. It allows you to apply special offers, discounts, or other actions specific to the “Rainbow Lollipop” that make it stand out among other candies in your store.

Let understand in more in-depth:

The Lookup Object field in the Price Rule object is a feature that allows you to establish a relationship between the Price Rule and another object in your Salesforce org.



[Salesforce CPQ by Smriti Sharan](#)

If we don't put any object name in the lookup object field, then does it mean it won't fetch data from that object and we cannot set condition based on a particular record?

That's correct. If **you don't specify an object name in the Lookup Object field** of the Price Rule, it means that the price rule **won't fetch data from any specific object**. Without a lookup object specified, you won't be able to set conditions or perform actions based on a particular record or data from another object.

The **Lookup Object field** is crucial for establishing a relationship between the **Price Rule and a specific object**. It enables you to access and leverage data from that object in your price rule logic. If you leave the **Lookup Object field blank**, the price rule will not have a connection to any object, limiting its ability to reference data from other records or apply conditions based on specific values.



[Salesforce CPQ by Smriti Sharan](#)

What is the significance of Active checkbox in Price Rule?

Active ⓘ

When the “Active” checkbox is checked, it indicates that the Price Rule is active and only then it will be evaluated during the pricing calculation process. Inactive price rules are not evaluated, and their conditions and actions are ignored.

How does the Evaluation Order field impact the application of Price Rules?

Evaluation Order ⓘ

The “Evaluation Order” in a Price Rule is essential for defining the sequence in which the rules are evaluated during the pricing calculation process. It helps establish the priority and hierarchy of rules when multiple



[Salesforce CPQ by Smriti Sharan](#)

rules are applicable. Rules with lower Evaluation Order values are evaluated first.

Example of Evaluation Order:

Rule 1: Discount for Chocolate Bars

Evaluation Order: 1

Condition: Candy Type equals "Chocolate Bar"

Action: Apply a 10% discount on the price of the chocolate bar.

Rule 2: Buy One, Get One Free

Evaluation Order: 2

Condition: Quantity is greater than or equal to 2

Action: Provide a free candy for every two candies purchased.

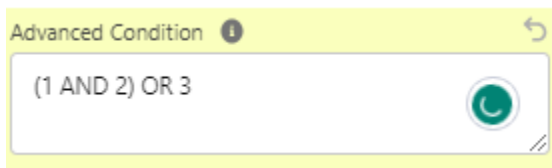


[Salesforce CPQ by Smriti Sharan](#)

In this example, Rule 1 has an Evaluation Order of 1, while Rule 2 has an Evaluation Order of 2.

Rule 1 will be evaluated first, followed by Rule 2.

How can you define complex conditions using the Advanced Condition feature in a Price Rule?



Let's explain the Advanced Condition in a Price Rule using a candy example:

Rule Scenario:

– You want to create a price rule for candies that are of type “Gummy Bear,” have a flavor of “Strawberry,” or fall within a price range of \$1 to \$5.

Creating the Advanced Condition:



[Salesforce CPQ by Smriti Sharan](#)

– With the Advanced Condition feature, you can define a complex condition that involves multiple fields and comparison values.

– In this case, you would set up Price Condition associated with price rule with the following components:

– Condition 1: Candy Type equals “Gummy Bear”

– Condition 2: Flavor equals “Strawberry”

– Condition 3: Price is between \$1 and \$5

```
(Candy_Type__c = 'Gummy Bear' && Flavor__c = 'Strawberry') || (Price__c >= 1 && Price__c <= 5)
```

As it is (1 AND 2) OR 3 so we will implement this logic by using advanced condition

What is the Product field in Price Rule?



[Salesforce CPQ by Smriti Sharan](#)

Product ⓘ

Let's explain the "Product" field in a Price Rule using a candy example in a simple and easy-to-understand way:

Imagine you have a candy store with a variety of candies, such as lollipops, chocolate bars, and gummy bears. Each candy has its own price. Now, let's talk about the "Product" field in a Price Rule:

1. Selecting a Candy Product:

– The "Product" field in a Price Rule allows you to choose **a specific candy from your store's inventory.**

– For example, you can select the "Gummy Bears" candy as the target product for the price rule.

2. Applying Pricing Actions to the Selected Candy:



[Salesforce CPQ by Smriti Sharan](#)

- Once you have selected the “Gummy Bears” candy in the “Product” field, you can define pricing actions that will be applied specifically to that candy.
- Pricing actions can include things like discounts or special offers.
- For instance, you can create a price rule that provides a 10% discount on the price of the “Gummy Bears” candy.

3. Creating Fun Conditions (Optional):

You can also set fun conditions based on the characteristics of the selected candy.

For example, you can create a condition that applies the price rule only if the “Gummy Bears” candy is red in color.

We can add the product in the lookup object field instead of product field in salesforce so what is the difference?

You’re correct! In addition to selecting the product directly in the “Product” field of a Price Rule, you also have the option to reference the product



[Salesforce CPQ by Smriti Sharan](#)

through a lookup relationship field using a lookup object. Let's explore the difference between the two approaches:

1. Product Field:

- When you select the product directly in the “Product” field of the Price Rule, you are associating the **price rule directly with a specific product**.
- This approach is useful when you want to define pricing actions or conditions that apply to a particular product or group of products.
- It is a straightforward way to target specific products within your catalog and define rules exclusively for those products.

2. Lookup Object:

- Using a lookup object, you can establish a relationship between **the Price Rule and another object** in Salesforce, such as a custom object or a standard object like an Account or Opportunity.



[Salesforce CPQ by Smriti Sharan](#)

– This allows you to reference the product indirectly through the related object.

– By using a lookup relationship, you can define pricing actions or conditions based on data from both the related object and the associated product.

– This approach provides more flexibility in creating rules that depend on information from other objects in addition to the product itself.

The choice between using the direct “Product” field or a lookup object depends on your specific use case and requirements. If you only need to target and define rules for specific products, using the “Product” field is typically sufficient. However, if your pricing logic involves data from related objects or requires more complex relationships, utilizing a lookup object can be beneficial.

What are some best practices for designing and configuring Price Rules in Salesforce CPQ?



[Salesforce CPQ by Smriti Sharan](#)

1. Define Clear Pricing Requirements:

- Clearly define your pricing requirements and understand the desired pricing behavior.
- Identify the pricing strategies, discounting rules, and custom calculations needed.

2. Plan Price Rules Structure:

- Plan the structure of your Price Rules in a logical and organized manner.
- Break down complex pricing scenarios into smaller, manageable Price Rules.
- This improves maintainability and makes troubleshooting and modifications easier.

3. Use Descriptive Rule Names:

- Give meaningful and descriptive names to your Price Rules.



[Salesforce CPQ by Smriti Sharan](#)

- This helps identify and understand the purpose of each rule, especially when dealing with many Price Rules.

4. Set Logical Evaluation Order:

- Set the Evaluation Order field on the Price Rule object appropriately.
- Control the sequence in which Price Rules are evaluated and applied.
- Ensure the evaluation order aligns with your intended pricing logic and requirements.

5. Test and Validate Price Rules:

- Thoroughly test and validate your Price Rules with various test scenarios.
- Verify that the pricing results align with the expected outcomes.
- Test different combinations of conditions, product configurations, and pricing scenarios.

6. Leverage Advanced Conditions:



[Salesforce CPQ by Smriti Sharan](#)

– Utilize the Advanced Conditions feature to implement complex rule criteria.

– Combine multiple conditions using logical operators (AND, OR) and functions.

– This allows for more intricate rule logic and fine-grained control.

7. Regularly Review and Update Rules:

– Regularly review your Price Rules to ensure they remain aligned with evolving pricing strategies and business requirements.

– Update rules as needed to reflect changes in products, pricing models, or discounting strategies.

8. Document Price Rule Logic:

– Document the logic and reasoning behind your Price Rules.



[Salesforce CPQ by Smriti Sharan](#)

– This serves as a reference for future modifications and helps with knowledge transfer within the team.

Can Price Rules be used to update field values on a record? Explain.

No, Price Rules in Salesforce CPQ are primarily used for pricing calculations and adjustments rather than directly updating field values on a record.

While Price Rules can affect the pricing outcomes and results, they are not designed to directly update field values on a record outside of the pricing context.

What is the difference between a Price Rule and a Discount Schedule?

Price Rules is used to define dynamic and conditional pricing logic, allowing for complex calculations and actions based on specific conditions. On the other hand, Discount Schedules provide a standardized approach to apply predefined discounts uniformly across multiple products or pricing



[Salesforce CPQ by Smriti Sharan](#)

scenarios. Price Rules offers more flexibility and customization options, while Discount Schedules offer a consistent and streamlined way to manage discounts.

Let's explain the difference between a Price Rule and a Discount Schedule using a candy example:

Price Rule:

- A Price Rule allows you to define specific pricing actions or adjustments based on certain conditions for individual candies or groups of candies.
- For example, you can create a Price Rule that applies a 10% discount to all gummy bear candies during a promotional period.
- Price Rules provide flexibility to apply different pricing logic, discounts, or adjustments based on various conditions for specific candies or groups of candies.



[Salesforce CPQ by Smriti Sharan](#)

Discount Schedule:

- In the same candy store scenario, a Discount Schedule would be used to manage standardized discounts that apply across multiple candies.
- A Discount Schedule provides a consistent approach to apply predefined discounts uniformly.
- For example, you can create a Discount Schedule called “10% Off” that offers a 10% discount on all candies storewide.
- Discount Schedules help maintain consistent pricing across different candies or pricing tiers, ensuring that the same discount is applied to multiple products without needing individual Price Rules.

What is Price Condition?

Let’s explore the use of the Price Condition object with the help of use cases.

1. Candy Type Discount:



[Salesforce CPQ by Smriti Sharan](#)

– Condition: Set the Field to “Candy Type” and the Operator to “Equals.”

Specify the Value as “Gummy Bears.”

– Result: When a customer adds gummy bear candies to their quote, the Price Condition will evaluate the candy type and apply the defined pricing action, such as a discount or adjusted price.

2. Quantity Threshold:

– Condition: Define the Field as “Quantity” and the Operator as “Greater Than” or “Equals.” Set the Value as a specific quantity threshold, like 10.

– Result: When a customer adds candies to their quote and the quantity exceeds the specified threshold, the Price Condition will trigger a pricing action, such as a bulk discount or a different pricing tier.

3. Custom Attribute Condition:

– Condition: Define a custom field in the Field section, such as “Organic Candy.” Set the Operator to “Equals” and the Value as “True.”



[Salesforce CPQ by Smriti Sharan](#)

– Result: When a customer adds organic candies to their quote, the Price Condition will check the custom attribute and apply a specific pricing action, such as a premium price or a discount for organic products.

4. Date-Based Promotion:

– Condition: Specify the Field as “Quote Date” Set the Operator to “Within Range” and provide the desired date range for the promotion.

– Result: When the quote or line-item falls within the specified date range, the Price Condition will trigger a pricing action, such as a limited-time discount or a special offer.

Price Rule and Price Condition object are having Master – Detail Relationship.



[Salesforce CPQ by Smriti Sharan](#)

SETUP > OBJECT MANAGER

Price Condition

Price Condition Custom Field

Price Rule (Managed)

[Back to Price Condition](#) [Help for this](#)

[This Custom Field Definition is managed, meaning that you may only edit certain attributes. \[Display More Information\]\(#\)](#)

[Validation Rules \(0\)](#)

Custom Field Definition Detail [Edit](#) [Set Field-Level Security](#) [View Field Accessibility](#) [Where is this used?](#)

Field Information		Object Name	Price Condition
Field Label	Price Rule		
Field Name	Rule		
Namespace Prefix	SBQQ	Data Type	Master-Detail
API Name	SBQQ__Rule__c		
Description	Price rule running this condition.		
Help Text	Price rule running this condition.		
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	Salesforce_Integration_User		6/7/2017 8:11 AM
Modified By	Salesforce_Integration_User		6/7/2017 8:11 AM

Understanding Fields in Price Condition Object



[Salesforce CPQ by Smriti Sharan](#)

A screenshot of the Salesforce CPQ configuration interface for a Price Condition. The interface is divided into three main sections: Information, Filter Information, and System Information. The Information section includes fields for Price Rule (set to 'test'), Object (set to 'None'), Field (set to 'None'), Tested Variable (with a search box for 'Search Summary Variables...'), and Tested Formula. The Filter Information section includes fields for Operator (set to 'None'), Filter Type (set to 'None'), Filter Value, Filter Variable (with a search box for 'Search Summary Variables...'), and Filter Formula. The System Information section is currently empty. At the bottom, there are buttons for 'Cancel', 'Save As New', and 'Save'.

What is the use of Price Rule Field in Price Condition object?

Connects Price Condition with Parent Price Rule. It signifies which Price Rule the Price Condition is associated with and determines under which rule the condition should be evaluated.

What is the significance of Object and Field in Price Condition?



[Salesforce CPQ by Smriti Sharan](#)

A screenshot of a Salesforce CPQ form. The 'Object' field is highlighted in yellow and contains the text 'Quote Line'. Below it, the 'Field' field is also highlighted in yellow and contains the text 'Type'. Both fields have a dropdown arrow on the right side.

When creating a Price Condition, you select the appropriate object from the available options in the “Object” field. [This specifies the related object that you want to reference in the condition.](#) Once you select the object, you can then choose the specific field from that object in the “Object Field” field.

For example, let’s say you have a Price Condition that checks if the “Candy Flavor” field on the related Quote Line object is equal to “Lemon”. In this case, you would select the “Quote Line” object in the “Object” field and then choose the “Candy Flavor” field.

Scenario: Set up a Price Condition to check if the “Type” field of the Candy associated with the Quote Line is equal to “Gummy Bear.”

Here’s how it works:



[Salesforce CPQ by Smriti Sharan](#)

1. In the Price Condition record, you would select the “Quote Line” object in the “Object” field. This indicates that you want to evaluate a field from the Quote Line object.
2. Next, in the “Object Field” field, you would choose the “Candy.Type” field. This specifies that you want to reference the “Type” field from the related Candy object associated with the Quote Line.
3. To complete the condition, you would set the operator as “Equals” and provide the value “Gummy Bear.” This condition will check if the Candy type for the associated Quote Line is “Gummy Bear.”



[Salesforce CPQ by Smriti Sharan](#)

A screenshot of the Salesforce CPQ Price Condition configuration interface. The interface is divided into several sections. The top section is for defining the condition: 'Object' is set to 'Quote Line', 'Field' is set to 'Type', 'Tested Variable' is a search box for 'Search Summary Variables...', and 'Tested Formula' is an empty text area. Below this is a 'Filter Information' section. The 'Operator' is set to 'equals', the 'Filter Type' is set to 'Value', and the 'Filter Value' is set to 'Gummy Bear'. The interface has a clean, modern design with yellow highlights on the dropdown menus.

When multiple price rules are here what challenges you see?

What is Tested Variable Field in Price Condition?

The “Tested Variables” field in the Price Condition object allows you to store and reference intermediate values during the evaluation of the condition. Let’s understand its use with a candy example.



[Salesforce CPQ by Smriti Sharan](#)

Scenario: You want to apply a discount on a specific type of candy, but only if the total quantity of all candy items in the quote exceeds a certain threshold.

1. Price Condition:

- Object: Quote Line
- Object Field: Product.Type (to check the candy type)
- Condition: Candy Type equals “Chocolate Bar”

2. Tested Variables:

- Create a tested variable called “TotalCandyQty” to store the total quantity of all candy items in the quote.

3. Price Action:

- Object: Quote
- Object Field: Discount Percentage



[Salesforce CPQ by Smriti Sharan](#)

– Action: Update Field

– **Value: If TotalCandyQty is greater than 10, set the Discount Percentage to 10%; otherwise, set it to 5%.**

Solution

In this example, the Price Condition checks if the candy type is “Chocolate Bar” for each Quote Line. If it matches, you can utilize **the “Tested Variables” field to store the total quantity of all candy items in the quote.**

The Price Action then updates the Discount Percentage field on the Quote based on the value of the “TotalCandyQty” tested variable. If the total quantity of candy items in the quote exceeds 10, a discount of 10% is applied. Otherwise, a discount of 5% is applied.

By using the “Tested Variables” field, you can calculate and store the total quantity of candy items in the quote and use it in subsequent actions to determine the appropriate discount percentage. This enables you to apply



[Salesforce CPQ by Smriti Sharan](#)

different discounts based on the overall quantity of candy items in the quote.

What is the use of Tested Formula Field in Price Condition?

Tested Formula ⓘ

The “Tested Formula” field in a Price Condition allows you to define a formula expression that is evaluated to determine whether the condition is met or not. It allows you to create dynamic conditions that consider different factors.

Scenario: You want to apply a discount on a product only if the following conditions are met:

- The product is a “Chocolate Bar” or a “Gummy Bear.”
- The quantity of the product is greater than or equal to 10.
- The total price of the product exceeds \$50.



[Salesforce CPQ by Smriti Sharan](#)

To set up this condition using a tested formula:

1. Create a Price Condition with the following details:

– Object: Quote Line

– Tested Formula: Enter the formula expression that combines the conditions:

```
`(Product.Type = "Chocolate Bar" || Product.Type = "Gummy Bear") &&  
Quantity >= 10 && TotalPrice > 50`
```

In this example, the tested formula combines multiple conditions using logical operators (|| for OR, && for AND). It checks if the product type is either "Chocolate Bar" or "Gummy Bear," the quantity is greater than or equal to 10, and the total price is greater than \$50. Only if all these conditions are true, the Price Condition will be met.

What is the difference between "Tested Formula" in a Price Condition and the "Advanced Condition" in a Price Rule?



[Salesforce CPQ by Smriti Sharan](#)

The “Tested Formula” in a Price Condition and the “Advanced Condition” in a Price Rule allow you to define complex conditions for determining when a specific action or adjustment should be applied. However, there are some key differences between them:

Tested Formula in Price Condition:

- It is used within a Price Condition to evaluate whether the condition is met or **not for individual quote line items**.
- It is evaluated at the line-item level, considering the specific values of fields on the quote line.
- The formula can include field references, logical operators, comparisons, and functions to perform calculations and comparisons based on the values of fields on the quote line.
- The result of the formula determines if the condition is met, and if so, the associated Price Action(s) will be applied to the quote line.



[Salesforce CPQ by Smriti Sharan](#)

Advanced Condition in Price Rule:

- It is used within a Price Rule to define more complex conditions that may involve **multiple quote line items or related objects**.
- It is evaluated at the rule level, considering the values across multiple quote line items or related objects.
- The advanced condition is typically written using Apex or SOQL (Salesforce Object Query Language) and can include custom logic, queries, and calculations.
- The result of the advanced condition determines if the entire rule is applied or skipped entirely for a particular quote.

In summary, the tested formula in a Price Condition is used to **evaluate conditions at the individual quote line item level**, while the advanced condition in a Price Rule allows for more advanced conditions involving multiple quote line items or related objects.



[Salesforce CPQ by Smriti Sharan](#)

Let's understand the difference between the tested formula in a Price Condition and the advanced condition in a Price Rule with the help of an example:

Scenario:

You have a Price Rule for applying a discount on products based on the following conditions:

- Condition 1: Apply a 10% discount if the product quantity is greater than or equal to 10.**
- Condition 2: Apply a 5% discount if the product belongs to a specific category, “Special Category”.**

Tested Formula in Price Condition:

In this approach, you will use a tested formula within each Price Condition to evaluate the condition at the individual quote line-item level.

Price Condition 1:



[Salesforce CPQ by Smriti Sharan](#)

– Object: Quote Line

– Tested Formula: `Quantity >= 10`

Price Condition 2:

– Object: Quote Line

– Tested Formula: `Product.Category = “Special Category”`

In this example, the tested formula is used to check if each quote line item meets the specific condition. If the formula evaluates to true for a quote line item, the associated Price Action will be applied to that item.

Advanced Condition in Price Rule:

In this approach, you will use an advanced condition within the Price Rule to evaluate the conditions at the rule level, considering multiple quote line items.

Price Rule:

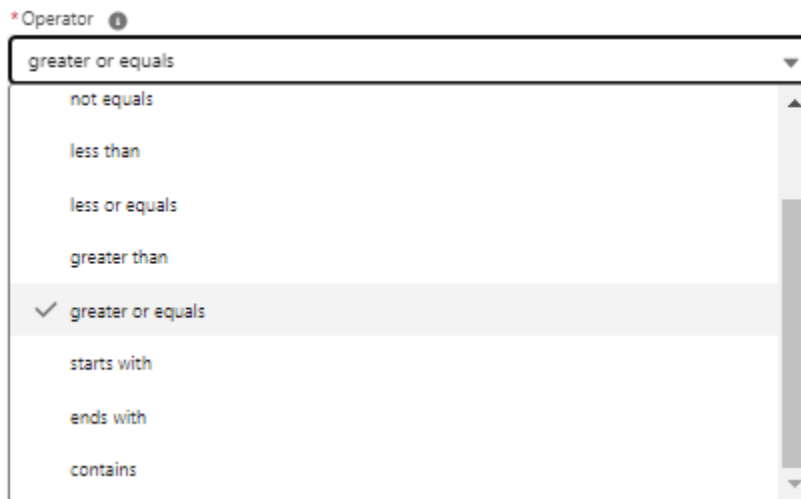


[Salesforce CPQ by Smriti Sharan](#)

– Advanced Condition: ``SUM(QuoteLine.Quantity) >= 10 && MAX(QuoteLine.Product.Category) = "Special Category"```

In this example, the advanced condition is used to evaluate the conditions across all the quote line items in the quote. If the advanced condition evaluates to true, the associated Price Actions within the Price Rule will be applied to all the eligible quote line items.

What is the use of the “Operator” field in a Price Condition?



This field determines the comparison or logical operation to be performed when evaluating the condition.



[Salesforce CPQ by Smriti Sharan](#)

The available operators in the “Operator” field may vary depending on the data type of the field being compared. Here are some common operators you may encounter:

- Equals (=): Checks if the field value is equal to the specified value.
- Not Equals (!=): Checks if the field value is not equal to the specified value.
- Greater Than (>): Checks if the field value is greater than the specified value.
- Less Than (<): Checks if the field value is less than the specified value.
- Greater Than or Equal To (>=): Checks if the field value is greater than or equal to the specified value.
- Less Than or Equal To (<=): Checks if the field value is less than or equal to the specified value.
- Contains: Checks if the field value contains a specific substring or value.



[Salesforce CPQ by Smriti Sharan](#)

For example, if you have a Price Condition to apply a discount when the quantity of a product is greater than 10, you will use the “Greater Than” operator (>) and specify the value 10 in the condition.

What is Use of Filter Type field in Price Condition Object?

A screenshot of a dropdown menu titled '* Filter Type' with an information icon. The menu is open, showing four options: '--None--', 'Value' (which is selected with a checkmark), 'Variable', and 'Formula'. The 'Value' option is highlighted in light blue.

1. Variable: Allows you to [select a summary variable that holds a specific value](#). Summary variables are used to store and reference calculated values based on formula logic or aggregations.
2. Formula: Allows you to specify a formula or expression that determines the [condition for the Price Rule to be applied](#). You can use formula functions, operators, and field references to create complex conditions based on the values of the quote line item or other related objects.



[Salesforce CPQ by Smriti Sharan](#)

3. Value: Specifies a specific value that the field on the quote line item will be compared against. It is typically used in conjunction with the “Field” and “Operator” fields to define the condition.

Together, the “Variable”, “Formula”, “Value” allow you to define the conditions that determine when the Price Rule should be applied to a specific quote line item.

What is Filter Value Field in Price Condition?

A screenshot of a software interface showing a text input field. The field is labeled 'Filter Value' in the top left corner, with a small information icon to its right. The input field itself is empty and has a blue border. In the top right corner of the field, there is a small grey arrow icon pointing to the right.

The filter value represents the expected value or threshold against which the field on the quote line or quote line group is compared. The filter value can be a numeric value, text value, date, or any other appropriate data type depending on the field being used in the condition.



[Salesforce CPQ by Smriti Sharan](#)

Scenario: Price Rule that offers a discount on gummy bear candies if the quantity exceeds a certain threshold. You want to apply for a 10% discount when the quantity is greater than 10.

Solution

In this scenario, the Price Condition will have the following configuration:

- Filter Type: Quote Line Field
- Field: Quantity
- Operator: Greater Than
- Filter Value: 10

Here, the filter value is set to 10, indicating that the condition is looking for a quantity greater than 10. When evaluating the quote lines, if the quantity of gummy bear candies exceeds 10, the condition is considered true, and the Price Rule with the associated action, such as applying the 10% discount, will be applied.



[Salesforce CPQ by Smriti Sharan](#)

What is the use of Filter Variable Field in Price Condition?



The Filter Variable Field in the Price Condition allows you to reference a custom variable specific to the a quote line or quote line group and use it in the condition evaluation.

Let's consider a candy example to understand the use of the Filter Variable Field in the Price Condition.

Scenario

Suppose you have a custom variable called " MinOrderQuantity ".The goal is to apply a discount when quantity exceeds the MinOrderQuantity value of 10

Solution:

1. Define a Custom Variable:



[Salesforce CPQ by Smriti Sharan](#)

– Create a custom variable called “MinOrderQuantity” to represent the minimum quantity of candy items required to qualify for a discount.

– Set the value of “MinOrderQuantity” to 10.

2. Create a Price Condition:

– Filter Type: Quote Line Field

– Field: Quantity

– Operator: Greater Than or Equal To

– Filter Variable Field: MinOrderQuantity

This Price Condition checks if the Quantity of the candy items in the quote line is greater than or equal to the value stored in the “MinOrderQuantity” variable.

3. Create a Price Action:



[Salesforce CPQ by Smriti Sharan](#)

– Specify the action to be taken when the Price Condition is met. For example, you can apply a discount of 10% to the Unit Price of the candy items.

4. Apply the Price Rule:

– Connect the Price Condition and Price Action to the Price Rule.

– Ensure that the Price Rule is active and set up to be triggered appropriately in your pricing configuration.

Now, let's see how this example works:

– Scenario 1: Quantity is less than 10

– If the Quantity of the candy items in the quote line is less than 10, the Price Condition evaluates to false.

– The Price Rule is not triggered, and no discount is applied.

– Scenario 2: Quantity is 10 or more



[Salesforce CPQ by Smriti Sharan](#)

– If the Quantity of the candy items in the quote line is equal to or greater than 10, the Price Condition evaluates to true.

– The Price Rule is triggered, and the specified action, such as applying a 10% discount to the Unit Price, is executed on those candy items.

What is the difference between Filter Variable and Tested Variable field in Price Condition?

The main difference between a filter variable and a tested variable in Salesforce CPQ is the context in which they are used.

Filter Variable:

– A filter variable is used in the Price Condition to define a dynamic condition **based on a specific value stored in a custom variable.**

– It allows you to reference a custom variable defined within the context of the quote line or quote line group.



[Salesforce CPQ by Smriti Sharan](#)

– The filter variable is used as part of the condition evaluation to compare against other fields or values.

– It helps determine if the conditions specified in the Price Condition is met or not.

Tested Variable:

– A tested variable is used in the Advanced Condition of a Price Rule to evaluate a formula or logic based on specific values stored in custom variables.

– It allows you to create complex conditions or calculations based on the values of custom variables.

– The tested variable is used in combination with other variables, fields, or formulas to determine the outcome of the Advanced Condition.

Let's differentiate between Filter Variable and Tested Variable field in the context of a candy example:



[Salesforce CPQ by Smriti Sharan](#)

Example: Seasonal Discount on Candy Store

Scenario: Filter Variable Usage

If the customer orders 25 candy items, the Price Condition with the filter variable “MinOrderQuantity” will evaluate if the quantity is greater than or equal to 20. If the condition is met, the specified action, such as applying a discount, will be triggered.

Solution

1. Create a custom variable called “MinOrderQuantity”
2. Set the value of “MinOrderQuantity” to 20, indicating that customers need to order at least 20 candy items to be eligible for the discount.

Price Condition:

3. Create a Price Condition in the Price Rule.
4. Select the “Quantity” field as the filter field.



[Salesforce CPQ by Smriti Sharan](#)

5. Choose the “Greater than or equal to” operator.
6. Enter the filter value as the “MinOrderQuantity” variable.
7. Specify the desired action to be taken, such as applying a percentage discount, when the condition is met.

Scenario Tested Variable Usage

Example: If the customer has a loyalty level of “Gold” and the tested variable “DiscountPercentage” is set to 15%, the Advanced Condition will evaluate if the customer’s loyalty level is “Gold” .If the condition is true, discount percentage will be executed.

Solution

1. Create a custom variable called “DiscountPercentage” in Salesforce CPQ.
2. Set the value of “DiscountPercentage” to 15, indicating a 15% discount for loyal customers.



[Salesforce CPQ by Smriti Sharan](#)

Advanced Condition:

3. Create an Advanced Condition in the Price Rule.
4. Define the logic using tested variables, such as “DiscountPercentage”, customer loyalty level, and other criteria.
5. Specify the desired action to be taken, such as applying the discount percentage, when the Advanced Condition evaluates to true.

In summary, **filter variables are used in Price Conditions to check if certain criteria or thresholds are met**, based on custom variables, fields, or values. **Tested variables, on the other hand, are used in Advanced Conditions to perform calculations or logic based on custom variables**, customer attributes, or other factors to determine the outcome of the Price Rule.



[Salesforce CPQ by Smriti Sharan](#)

How do we use Filter Formula field in price condition?



The filter formula field in the Price Condition allows you to define a formula-based condition to determine if the rule should be applied. Let's consider an example in the context of a candy store:

Scenario: Buy One, Get One Free Promotion

If the customer is purchasing at least two candy items apply a discount to make one of the items free as part of a “Buy One, Get One Free” promotion.

Solution

Here's a step-by-step example of using the filter formula field in the Price Condition to implement a “Buy One, Get One Free” promotion in a candy store scenario:



[Salesforce CPQ by Smriti Sharan](#)

1. Configure the Price Condition:

- Set the filter type to “Formula” in the Price Condition record.
- Enter a formula in the filter formula field to check if the condition is met.

Quantity >= 1.

- This formula checks if the quantity of the candy item in the quote line is greater than or equal to 1, indicating that the customer is purchasing at least two items.

2. Specify the desired action:

- Determine the action you want to take when the filter formula evaluates to true. In this case, it is a “Buy One, Get One Free” promotion.

What is the use of Index Field in Price Condition?

Index ⓘ



[Salesforce CPQ by Smriti Sharan](#)

The “Index” field in the Price Condition object can be used to define the order in which the conditions are evaluated for applying pricing rules. Let’s consider an example to understand its use:

What is Price Action?

Let’s explore the use of the Price Action object with various examples.

1. Applying a Discount:

- Price Action: Set a Price Action object with the Target Object as “Quote Line”
- Action: Specify a discount percentage, like 20% off, in the Price Action.
- Result: When a customer adds a specific candy to their quote, the Price Action will automatically apply the discount to the Unit Price of that candy, resulting in a reduced price.

2. Volume Pricing:



[Salesforce CPQ by Smriti Sharan](#)

– Price Action: Create a Price Action object with the Target Object set as “Quote Line.”

– Action: Define a price tier structure in the Price Action based on quantity ranges. For example:

– 1-10 candies: \$1 per candy

– 11-20 candies: \$0.90 per candy

– 21+ candies: \$0.80 per candy

– Result: When a customer adds candies to their quote, the Price Action will adjust the Unit Price based on the quantity, providing volume-based pricing discounts.

3. Gift with Purchase:

– Price Action: Set the Target Object as “Quote” to apply the action at the quote level.



[Salesforce CPQ by Smriti Sharan](#)

– Action: Specify a condition that if the customer purchases a certain quantity of a specific candy, they will receive a gift.

– Result: When the specified condition is met, the Price Action will automatically add the free gift to the quote, enhancing the customer experience.

4. Special Pricing for VIP Customers:

– Price Action: Create a Price Action object with the Target Object as “Quote” or “Quote Line” based on your requirement.

– Action: Define a custom field that identifies VIP customers. Set the Price Action to apply a specific discount for those customers.

– Result: When a VIP customer adds candies to their quote, the Price Action will recognize their status and apply the exclusive discount automatically.



[Salesforce CPQ by Smriti Sharan](#)

These examples demonstrate how the Price Action object allows you to define specific actions, such as applying discounts, adjusting pricing based on quantity, offering free gifts, or providing special pricing for specific customer segments. The Price Action object enables you to automate these actions.

Price Rule and Price Action object have master detail relationship.

Price Action Custom Field
Rule (Managed)
[Back to Price Action](#)

This Custom Field Definition is managed, meaning that you may only edit certain attributes. [Display More Information](#)

[Validation Rules](#)

Custom Field Definition Detail [Edit](#) [Set Field-Level Security](#) [View Field Accessibility](#) [Where is this used?](#)

Field Information		Object Name	Detail Type
Field Label	Rule	Price Action	Master-Detail
Field Name	Rule		
Namespace Prefix	SBOO		
API Name	SBOO__Rule__c		
Description	Price rule running this action.		
Help Text	Price rule running this action.		
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	Salesforce Integration User	0/7/2017 8:11 AM	
Modified By	Salesforce Integration User	0/7/2017 8:11 AM	

Understanding Fields in Price Action Object



[Salesforce CPQ by Smriti Sharan](#)

* = Required information

Information

Action #

* Target Object ⓘ
--None--

* Target Field ⓘ
--None--

Value ⓘ

Formula ⓘ

* Rule ⓘ
Check WDH Exist

Order ⓘ

Price Action Sources

Source Field ⓘ

Source Variable ⓘ
Search Summary Variables...

Source Lookup Field ⓘ
--None--

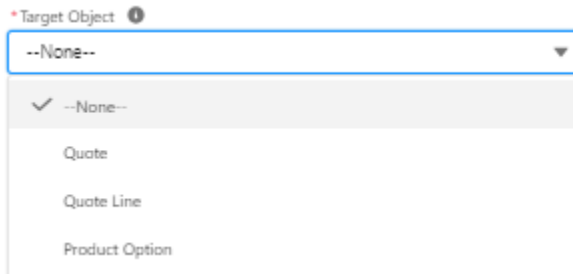
What is the use of Price Rule Field in Price Action object?

Connects Price Action with Parent Price Rule. It signifies which Price Rule the Price Action is associated with and determines under which rule the condition should be evaluated.

What is the significance of Target Object in Price Action?



[Salesforce CPQ by Smriti Sharan](#)



Target Object in Price Action determines where the pricing changes specified in the Price Action will be applied.

Suppose you have a Price Action that applies a discount of 10% to the Unit Price of gummy bear candies when the quantity exceeds 100. The Target Object field in the Price Action determines which records will receive this discount.

In this case, you would select the “Quote Line” object as the Target Object in the Price Action. This means that the discount will be applied to the Quote Line records that represent the gummy bear candies in the quote.

Significance of Value field in Price Action?



[Salesforce CPQ by Smriti Sharan](#)

A screenshot of a Salesforce 'Value' field. The field is highlighted in yellow. It contains the number '2.5'. There is a small 'i' icon in the top left corner and a refresh icon in the top right corner of the field.

The Value field in the Price Action object is used to specify the specific value or data that needs to be set field in the target object. It defines the outcome of the Price Action, indicating the desired result or change that should occur.

Scenario: When a customer adds a strawberry-flavored gummy bear to their quote, the Price Action will update the Price field of that candy to the specified value of \$2.50.

Solution:

- Price Action: Create a Price Action object with the Target Object set as “Quote Line.”
- Field: Choose the “Price” field in the target object that you want to update with a new value



[Salesforce CPQ by Smriti Sharan](#)

– Value: Specify the desired price for the candy flavor in the Value field. If the candy is a strawberry-flavored gummy bear, set the Value field as \$2.50.

What is the use of the Formula field in the Price Action object and the Formula field in the Price Condition object?

Let's understand when to use the Formula field in each object.

- The Formula field in the Price Action is used to calculate and directly apply the discount amount or field value based on certain criteria.
- The Formula field in the Price Condition sets conditions or calculates values that determine when the Price Action should be executed.

Suppose you want to offer a 10% discount on a “chocolate” candy flavor, but the discount amount varies based on the quantity ordered of the chocolate candy flavor in the quote.

Using Formula in Price Action:



[Salesforce CPQ by Smriti Sharan](#)

In this scenario, you can add a Formula field directly to the Price Action object. The formula would calculate the discount amount based on the quantity of the “chocolate” candy flavor.

Using Formula in Price Condition:

In some cases, you may want to apply the discount only if the quantity of the specific candy flavor exceeds a certain threshold.

“IF (Quantity > 10, 0.2, 0.1)” which sets the discount percentage to 20% if the quantity is greater than 10, and 10% otherwise. The Price Condition checks this formula and evaluates whether the condition is met. If the condition is met, it triggers the associated Price Action with the defined discount percentage.

Reasons for Using Formulas in Price Condition:

1. **Conditional Logic:** The Formula field in the Price Condition allows you to define conditional logic based on specific criteria. You can **set up complex**



[Salesforce CPQ by Smriti Sharan](#)

conditions involving multiple fields or calculations to determine if the Price Action should be triggered.

2. Advanced Filtering: The Price Condition can **evaluate various factors such as quantity, price, date range**, or any other relevant fields before deciding to apply the Price Action.

Reasons for Using Formulas in Price Action:

1. Direct Field Updates: If the discount calculation is straightforward and does not require complex conditions or filtering, you can directly calculate and update the field value in the Price Action.

What is the Significance of Source Field in Price Action?

The “Source Field” in Price Action specifies the field from which the value will be fetched and used in the Price Action.



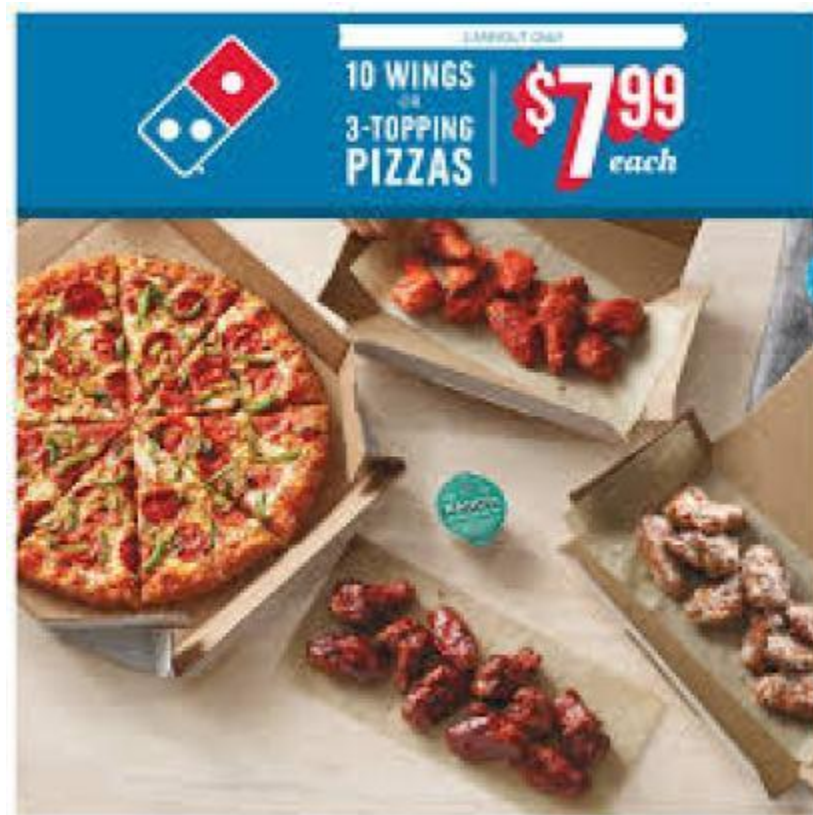
[Salesforce CPQ by Smriti Sharan](#)

Pricing Methods, Discounting and Price Waterfall



[Salesforce CPQ by Smriti Sharan](#)

Ready to deep dive into Salesforce CPQ pricing methods, Price Waterfall Model, CPQ Discounting? We will understand Salesforce CPQ Pricing tackle interview questions with the help of mouthwatering Domino's offerings. 🍕📊📶





[Salesforce CPQ by Smriti Sharan](#)

What are the pricing methods in Salesforce CPQ?

Pricing Method	Function
Block Pricing	The method is used if we are selling a product in some specific volumes. For example, pencils in boxes: small (10 pencils), medium (11-50 pencils), large (51-150 pencils).
Percent of total	It is used when we need to set the price for some product in Quote as a % of the total cost of products in Quote. The price for this product is % of the total Quote.
Cost plus markup pricing	Allows you to use not the automatic CPQ price substitution, but the base price of the product + surcharges.
Option Pricing	It is used when the price of the product included in the Bundle should differ from the standard price of this product in the Price List.
Contracted Pricing	Allows setting a special price for some clients (for example, lower than in the Price List).
Manual Override	Allows editing prices manually.



[Salesforce CPQ by Smriti Sharan](#)

What is the list price in Salesforce CPQ?

Salesforce CPQ Pricing

<p>Pricing Method ⓘ</p> <p>List</p> <p>--None--</p> <p>✓ List</p> <p>Cost</p> <p>Block</p> <p>Percent Of Total</p>	<p>Optional ⓘ</p> <p><input type="checkbox"/></p> <p>Hide Price in Search Results ⓘ</p> <p><input type="checkbox"/></p> <p>Percent Of Total (%) ⓘ</p> <p></p> <p>Percent Of Total Base ⓘ</p> <p>List</p> <p>Percent Of Total Category ⓘ</p> <p>--None--</p> <p>Percent Of Total Constraint ⓘ</p> <p>--None--</p> <p>Percent Of Total Target ⓘ</p> <p>Search Products...</p> <p>Include In Percent Of Total ⓘ</p> <p><input type="checkbox"/></p>
<p>Subscription Term ⓘ</p> <p></p> <p>Default Pricing Table</p> <p>--None--</p> <p>Block Pricing Field ⓘ</p> <p>Quantity</p> <p>Cost Editable ⓘ</p> <p><input type="checkbox"/></p>	

Cancel Save & New Save



[Salesforce CPQ by Smriti Sharan](#)

List price is the price that is **retrieved from the price book entry**. A major part of the list price is price book entry as the prices are invoked from the price book. Each product must have a price book entry or else the product will not get listed on the selection page of the quote line editor.

For example, Domino's has their new "AI Extravaganza" pizza, priced at \$24 in their CPQ price book. That's our list price. Now, on "Techie Night," they offer a 20% discount for all tech enthusiasts. Salesforce CPQ takes the \$24 list price, calculates the 20% discount, and the customer is offered the pizza at \$19.20. But without the list price book entry establishing the list price.

What is the Cost Price?

Cost price is used when a product costs a specific amount to obtain, then you can add cost price and add markup price on the product. Markup can be negotiated between the sales reps and the customer. Cost price is the price which is filled in special price. field.



[Salesforce CPQ by Smriti Sharan](#)

Imagine Domino's new "AI Extravaganza" pizza. The ingredients, labor, and overhead costs total up to \$15 per pizza — that's our Cost Price. Now, Domino's wants a slice of profit, so they add a markup. Let's say they mark it up by \$9, making the List Price \$24 (the price you see on the menu).

What is Block pricing?

Block pricing is the pricings that are updated based on the quantities of the product purchased. For e.g., a pack of 1-10 units will cost 10\$, 10-20 units will cost 18\$ and so on.

To set up the block pricing we need to update the 4 pieces of information.

Name: Set the name to block the pricing method.

Lower bound: The lowest quantity for the quantity range.

Upper bound: The highest quantity for the quantity range.

Price: Price to be set up for the given quantity range.



[Salesforce CPQ by Smriti Sharan](#)

Block Prices Editor Toner Recycling Help for this Page

Block Prices Save Quick Save Cancel

Price Book: Standard Price Book Clone From: None Clone

Price Name	Lower Bound	Upper Bound	Price (\$) OverageRate (\$)
Small Pickup	1	21	25.00
Large Pickup	21	51	35.00
Overage	51		45.00

Save Quick Save Cancel

Note: A block price’s upper bound and the next block price’s lower bound must be the same.

Let’s understand with help of Dominos scenario, imagine block pricing as Domino’s special offer on their pizzas, but instead of “Buy 1, Get 1 Free,” it’s more like “Buy More, Save More!” This approach is used in Salesforce CPQ to encourage customers to purchase in higher volumes by reducing the unit price as the quantity increases.

Let’s say Domino’s is selling their “AI Extravaganza” pizza, and they introduce a block pricing strategy to boost sales:

- **Name:** “AI Extravaganza Block Deal”
- **Lower Bound:** 1 (minimum number of pizzas)



[Salesforce CPQ by Smriti Sharan](#)

- **Upper Bound**: 10 (maximum number of pizzas before the next block)
- **Price**: \$24 per pizza

So, if you order between 1 and 10 “AI Extravaganza” pizzas, each cost \$24. But Domino’s wants to incentivize you to order more for your tech meetups or gaming nights, so they introduce the following:

- **Next Block’s Lower Bound**: 11
- **Next Block’s Upper Bound**: 20
- **Next Block’s Price**: \$22 per pizza

With this deal, if you order 11 to 20 pizzas, the price drops to \$22 each.

Your attendees are happy because there’s plenty of pizza, and Domino’s is thrilled with the larger order. It’s a win-win!

So, block pricing is like unlocking volume discounts: the more you order, the cheaper each pizza becomes.



[Salesforce CPQ by Smriti Sharan](#)

Where do you set block price, cost price?

What is Percent of Total?

Percent of the total is calculated based on the percent of the sum of all the other product that is purchased. For example, the tip for a bill at a restaurant would be calculated in this category.

For implementing the percent of the total, we need to update the following fields on products:

Pricing Method: Set this to “Percent of Total”

Percent of Total Base: This field says on which price field should we implement the percent of the total.

Option Pricing Override: This pricing will work only on the bundle product.

The product options are the option on the bundle. You can override the price of the product if sold standalone and along with the bundle.

Let's dive into the “Percent of Total” concept with a mouth-watering Domino's scenario!



[Salesforce CPQ by Smriti Sharan](#)

Imagine you're hosting a big game night. So, you order a variety of pizzas, sides, and desserts, enough to satisfy the diverse cravings of your friends.

Here's a breakdown of your order:

5 pizzas at \$15 each	\$75
4 orders of garlic bread at \$5 each	\$20
3 lava cakes at \$6 each	\$18
Total	\$118
Drinks are a separate entity	
6 bottles of soda at \$2.50 each	%15

Here's how it's integrated into Salesforce CPQ:

Pricing Method: They've selected "Percent of Total." It's dynamic, changing with your food choices and portions.



[Salesforce CPQ by Smriti Sharan](#)

Percent of Total Base: It's defined as the "Total Food Bill," which is your cost for pizzas, sides, and desserts, totaling \$113. Your sodas are out of this.

Option Pricing Override: Normally, a bottle of Domino's cola is \$2.50 but you ordered the "Combo!" This combo privilege drops your soda cost to \$1.80 per bottle, saving you \$4.20 on drinks!

Total

"Party Tax" is 10% of \$113, so that's \$11.30. Your total food cost was \$113, drinks were \$15, and with your drink discount due to "Combo" offer, you subtract \$4.20, then add your "Party Tax" of \$11.30. So, your grand total for the night is \$135.10!

Explain the concept of "Option Pricing"?

Imagine you're throwing a pizza party and Domino's has this deal called the "Pizza Party Pack."- which includes pizzas, sides, and drinks.



[Salesforce CPQ by Smriti Sharan](#)

Here's how it works:

Option Pricing: In the "Pizza Party Pack," garlic bread doesn't cost \$5; it's just \$3.75! It gets a price cut because it's in a bundle.

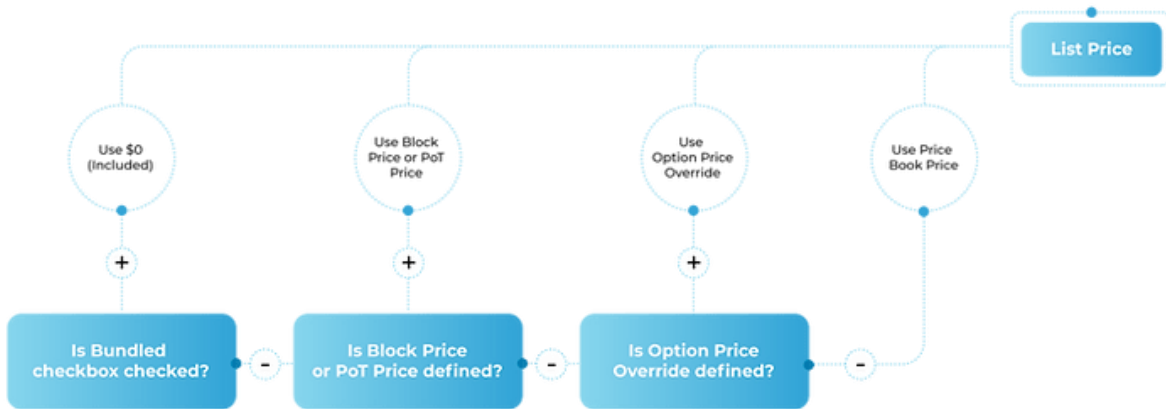
Steps:

1. First, you add garlic bread to your cart – it's still \$5.
2. Then, you choose the "Pizza Party Pack" and the garlic bread in your bundle is now priced at \$3.75.

What's happening here, due to the concept of Option Pricing, is that the system checks if garlic bread is part of the "Pizza Party Pack." the discounted price of \$3.75, as specified in the bundle option, kicks in.



[Salesforce CPQ by Smriti Sharan](#)



Explain Limitations in Option pricing?

Option pricing does not support multicurrency, so it's only possible to define the override price in a single currency.

Explain Contracted Pricing in Salesforce CPQ?



[Salesforce CPQ by Smriti Sharan](#)

Imagine Domino's has a loyal customer company, "Pizza Lovers Texas Inc.," a local business that orders pizzas for its staff every Friday. Since they order so frequently and in bulk, Domino's wants to offer them a special deal.

Here's how it works:

1. Normally, a large veggie pizza costs \$15. But for Pizza Lovers Texas, Domino's sets a contracted price of \$12 per pizza, a loyalty gesture.
 1. On Domino's Salesforce CPQ system, they go to Pizza Lovers Texas's account.
 2. Navigate to the "Related" tab, and in the "Contracted Prices" section, they click "New."
 3. They select the veggie pizza in the "Product" field and enter \$12 in the "Price" field- the contracted price is set.

Contracted Pricing pro tips! 📺 ✨



[Salesforce CPQ by Smriti Sharan](#)

1. No Double Deals: if Pizza Lovers Texas Inc. had two different contracted prices for the same Veggie Pizza — one contract offers it at \$12, and an older one at \$14. Domino's Salesforce system would not know which one I should pick. To avoid this, CPQ makes sure they have only ONE contracted price per product for each account.

2. Special Pricing Exceptions: Products that use Pricing Methods Block Pricing or Percent of Total Pricing do not work with Contracted Price.

3 Family Matters: If the Accounts are linked by a parent-child relationship, then by default the Contracted Price created on the Parent Account is inherited by all children.

Say, PizzaLovers Texas Inc. is part of a bigger empire, "PizzaLovers Universe," with subsidiaries like "PizzaLovers Chicago" and "PizzaLovers Bay" setup in different locations. Domino's sets up a contracted price with the parent company, "PizzaLovers Universe." Salesforce CPQ



[Salesforce CPQ by Smriti Sharan](#)

automatically applies this contracted price to all the PizzaLovers' subsidiaries because, in the pizza world, family sticks together.

Explain the use case for 'Ignore Parent Contracted Prices'?

What if "PizzaLovers Chicago" negotiates a different, exclusive deal because they order twice as much every Friday? Domino's simply toggles the "Ignore Parent Contracted Prices" to true on that account, ensuring "PizzaLovers Chicago" gets their unique deal, while the rest still benefit from the parent company's contracted price.

Explain Price "Manual Override" in Salesforce CPQ Pricing? 📶 💧

1. Create Your Own Pizza:



[Salesforce CPQ by Smriti Sharan](#)

DOMINO'S PIZZA BUILDER



MY PIZZA

Large (14") Hand Tossed Pizza
Whole: Pepperoni, Italian Sausage, Black Olives, Mushrooms, Onions

Quantity: 1

ADD TO ORDER

← BACK

CHOOSE TOPPINGS

CHOOSE MEATS

- Pepperoni Salami Premium Chicken
- Italian Sausage Premium Chicken
- Sliced Italian Sausage
- Beef
- Philly Steak
- Ham
- Bacon

CHOOSE NON-MEATS

- Cheddar Cheese Green Peppers
- Feta Cheese Jalapeno Peppers
- Shredded Parmesan Asiago Mushrooms
- Shredded Provolone Cheese Banana Peppers Black Olives Pineapple
- Garlic Onions
- Roasted Red Peppers
- Spinach
- Diced Tomatoes
- Hot Sauce

Dominos knows that sometimes you want to create your own pizza. So, they set the "Price Editable" field to true on the 'Create Your Own Pizza'



[Salesforce CPQ by Smriti Sharan](#)

product. You get into the Quote Line Editor, and, you can adjust the List Price based on the number and type of toppings you choose on that pizza!

Quote Information							
	Start Date		Subscription Term		Additional Disc. (%)		
#	PRODUCT CODE	PRODUCT NAME	QUANTITY	LIST UNIT PRICE	ADDITIONAL DISC.	NET UNIT PRICE	NET TOTAL
1	COPYMACHINE	Copy Machine	1.00	\$2,100.00		\$2,100.00	\$2,100.00
PRICING METHOD			ORIGINAL PRICE	SPECIAL PRICE			
List			\$2,000.00	\$2,100.00			
OPTIONAL			UNIT COST	REGULAR UNIT PRICE			
<input type="checkbox"/>			\$1,600.00	\$2,100.00			
PACKAGE PRODUCT CODE			MARKUP	CUSTOMER UNIT PRICE			
				\$2,100.00			
SUBTOTAL:						\$2,100.00	

2. Flexibility:

Domino’s offers free Wi-Fi in their dine-in locations. The Wi-Fi is typically ‘sold’ using ‘Pricing Method = List,’ but for large party bookings, they offer it as a perk, part of ‘Pricing Method = Percent of Total’ with the total bill. To switch between these methods, they make the ‘Pricing Method’ field editable in the Quote Line Editor by setting ‘Pricing Method Editable’ to true for the product (in this case, Wi-Fi).



[Salesforce CPQ by Smriti Sharan](#)

Quote Information

Start Date Subscription Term Additional Disc. (%)

#	PRODUCT CODE	PRODUCT NAME	QUANTITY	LIST UNIT PRICE	ADDITIONAL DISC.	NET UNIT PRICE	NET TOTAL	
<input type="checkbox"/>	1	WIFIINSTALL	WiFi Installation	1.00	\$300.00	\$300.00	\$300.00	
PRICING METHOD <input checked="" type="checkbox"/> List <input type="checkbox"/> Cost <input type="checkbox"/> Block <input type="checkbox"/> Percent Of Total <input type="checkbox"/> Custom				ORIGINAL PRICE	\$300.00	SPECIAL PRICE	\$300.00	
PACKAGE PRODUCT CODE				UNIT COST	\$175.00	REGULAR UNIT PRICE	\$300.00	
				MARKUP		CUSTOMER UNIT PRICE	\$300.00	
<input type="checkbox"/>	2	WIFIAP	WiFi Access Point	1.00	\$60.00	\$60.00	\$60.00	
							SUBTOTAL:	\$360.00

3. Custom Deals for Custom Meals:

Sometimes, a customer negotiates a deal that doesn't fit the usual pricing methods. Domino's responds by setting 'Pricing Method = Custom' in the Quote Line Editor. The staff can then manually set a custom price.

Quote Information

Start Date Subscription Term Additional Disc. (%)

#	PRODUCT CODE	PRODUCT NAME	QUANTITY	LIST UNIT PRICE	ADDITIONAL DISC.	NET UNIT PRICE	NET TOTAL	
<input type="checkbox"/>	1	TONERRECYCLE	Toner Recycling	40.00	\$35.00	\$0.75	\$30.00	
PRICING METHOD <input type="checkbox"/> List <input type="checkbox"/> Cost <input type="checkbox"/> Block <input type="checkbox"/> Percent Of Total <input checked="" type="checkbox"/> Custom				ORIGINAL PRICE	\$0.00	SPECIAL PRICE	\$35.00	
OPTIONAL				UNIT COST		REGULAR UNIT PRICE	\$35.00	
PACKAGE PRODUCT CODE				MARKUP		CUSTOMER UNIT PRICE	\$0.75	
							SUBTOTAL:	\$30.00

4. The Surprise Side:



[Salesforce CPQ by Smriti Sharan](#)

Ever had a Domino's staff recommend cheesy garlic bread to you to order? Sometimes, they add it to your order as an 'Optional' item by setting the 'Optional' field to true in the Quote Line Editor. It won't add up to your total but it's there for you to consider.

Explain Price Waterfall?



Let's dive into the concept of the pricing waterfall in Salesforce CPQ! 🌊

The CPQ Pricing Waterfall is a multi-step process/sequence that utilizes different prices and discounts to get us to the Net Price (this is the price field for the total of the Quote Line/Quote).

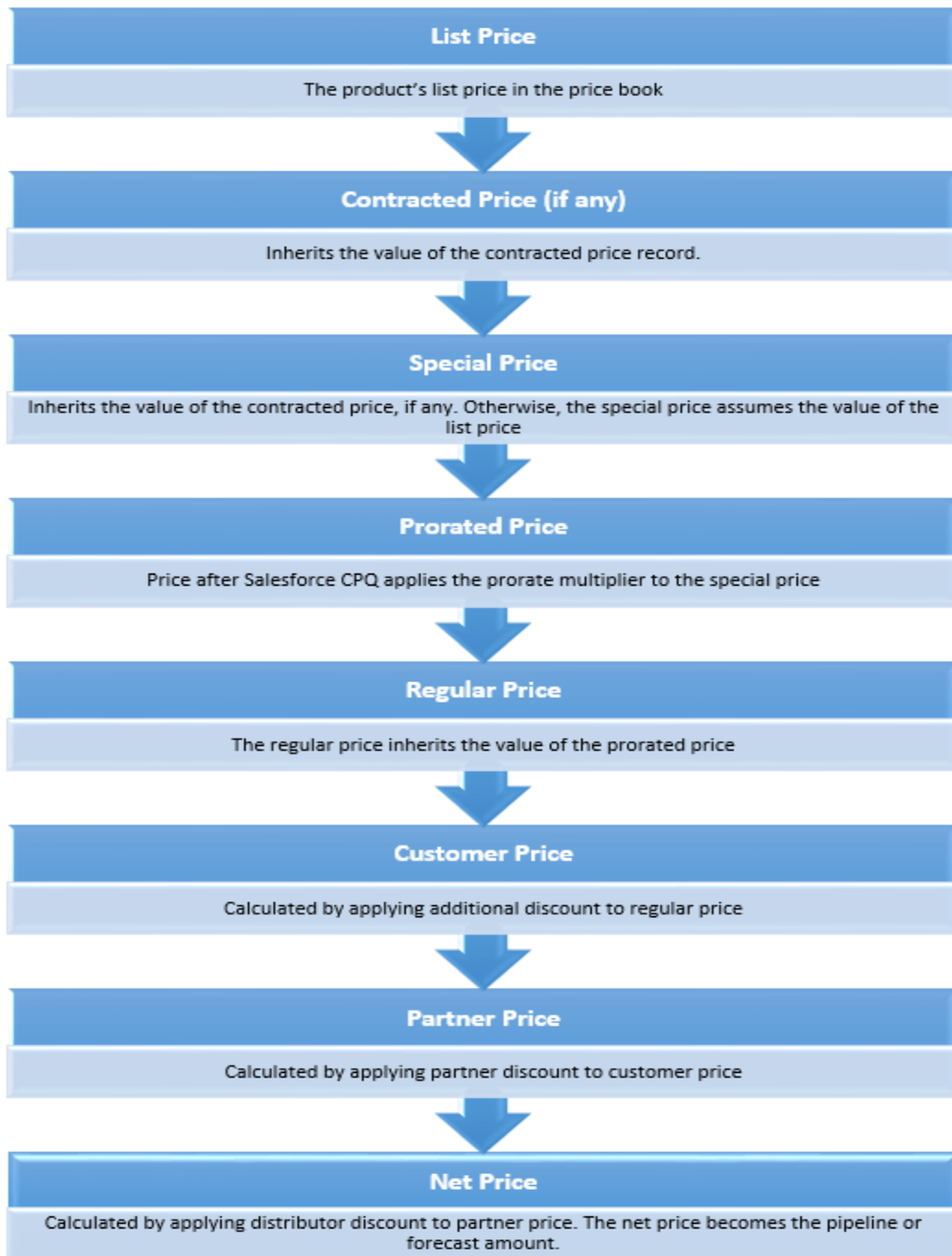


[Salesforce CPQ by Smriti Sharan](#)

It is described as a waterfall as it works its way from top to bottom through the different prices and discounts/adjustments.



[Salesforce CPQ by Smriti Sharan](#)





[Salesforce CPQ by Smriti Sharan](#)

1. List Price:

This is the starting point – the menu price. Let's say the List Price for Domino's "ExtravaganZZa" large pizza is \$20, as listed in their Price Book.

2. Contracted Price:

Domino's has loyal customers, like the "Pizza Club," who dine-in every Tuesday. They've negotiated a 5% discount on all pizzas through a loyalty program. So, for them, the Contracted Price for the "ExtravaganZZa" isn't \$20 but drops to \$19 (\$20 minus 5%).

3. Special Price:

On occasions, Domino's offers a special one-day promotion, like "Cheesy Thursday," where all pizzas have an additional 10% off. This promo is applied on the Contracted Price or the List Price. For the loyal "Pizza Club," this takes their \$19 pizza down to \$17.10 for that day.



[Salesforce CPQ by Smriti Sharan](#)

4. Prorated Price:

Suppose one member of the club wants to buy half of an “ExtravaganZZa” and half of a “Veggie Delight” (also priced at \$20). Dominos calculates a Prorated Price, where they take the Special Price (\$17.10) and multiply it by a factor considering the half-and-half scenario, let’s say 0.5 for each half, equating to \$8.55 for each half-pizza portion.

5. Regular Price:

If there’s no special scenario, the Regular Price kicks in. It’s either the Prorated Price or goes back to the original List Price. So \$19 per “ExtravaganZZa.”

In the context of Domino’s pizza scenario with Salesforce CPQ, let’s elaborate on the Customer Price, Partner Price, and Net Price.

6. Customer Price:



[Salesforce CPQ by Smriti Sharan](#)

It is the final price that a direct consumer pays after all applicable discounts and special prices are considered. For the “Pizza Club” members dining on a “Cheesy Thursday,” their Customer Price for an “ExtravaganZZa” would be the special rate of \$17.10.

7. Partner Price:

The Partner Price is a special rate offered to business partners, usually lower than the standard rate, to accommodate the partners’ profit margins. For instance, a food delivery app ‘Uber’ might receive the “ExtravaganZZa” at a Partner Price of \$16, allowing them to mark it up for their customers.

8. Net Price:

It is the revenue that the seller keeps from the sale after all discounts, allowances, or rebates are deducted. The Net Price reflects the actual revenue the seller makes on the product before their internal costs are subtracted.



[Salesforce CPQ by Smriti Sharan](#)

In the Domino's case, the Net Price is the amount Domino's records as revenue for the "ExtravaganZZa" pizza sold to the club member on "Cheesy Thursday".

How can Salesforce CPQ help a company manage pricing complexity more efficiently?

With Salesforce CPQ, instead of creating multiple price books, company can have a standard price book and layer on pricing exceptions and adjustments as needed. For example, if there's a nationwide 20% discount on "Veg Extravaganza" pizza this can be applied on top of the standard price book without creating a new one.

Can sales reps offer product as part of a bundle at a different price with Salesforce CPQ, even though it's the same product listed in the price book?



[Salesforce CPQ by Smriti Sharan](#)



Yes! Salesforce CPQ allows for advanced pricing configurations. For instance, a bottle of “Coke” might be priced at \$2 when sold individually. Still, when part of the “Family Feast” bundle (which includes pizzas, sides, and a drink), the effective price of the “Coke” within the bundle might be just \$1.50. This differential pricing is possible within the same price book due to the flexibility of Salesforce CPQ.

Do sales reps still need price books if they’re using Salesforce CPQ?

Yes, price books remain a foundational element in Salesforce CPQ. They provide the baseline prices from which all adjustments, discounts, or special pricing are derived. If a product doesn’t have a price book entry, it



[Salesforce CPQ by Smriti Sharan](#)

won't appear in the product selection page and can't be added to a quote. Even a \$0 entry can be significant, as it may indicate a complimentary item or a product whose final price will be determined through Salesforce CPQ's pricing methods.

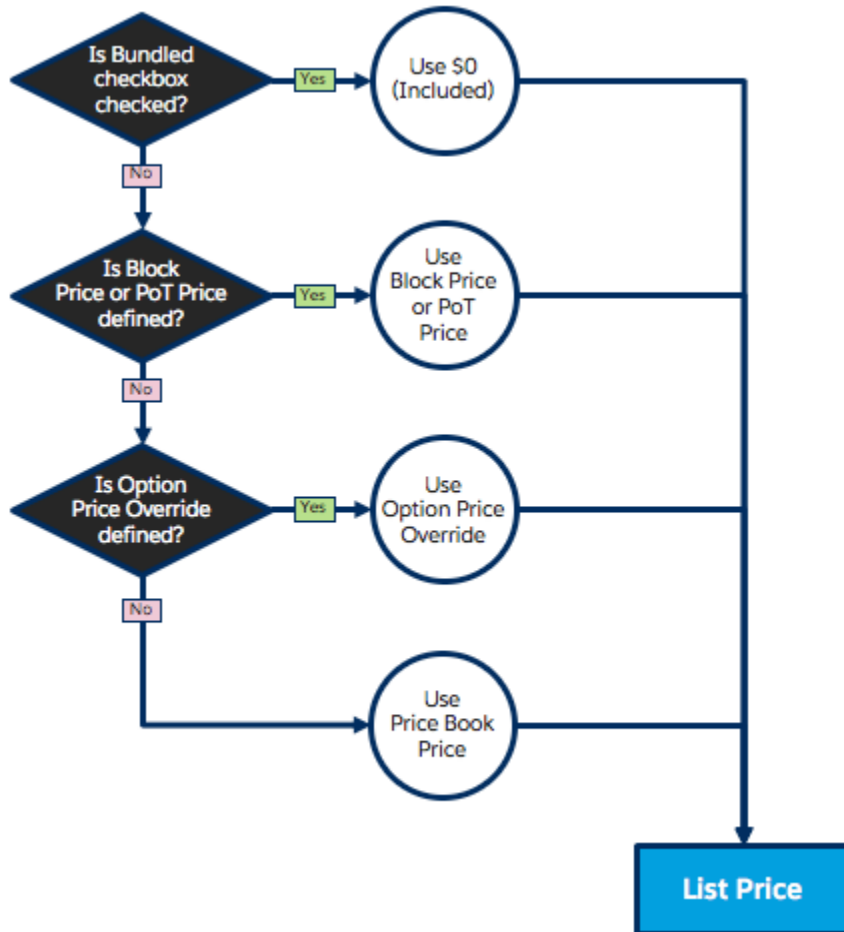
What should sales reps remember when creating new products in their system?

Domino's should always ensure that any product they introduce has an associated price book entry. This inclusion is crucial because, without it, the product won't be available for selection when creating quotes, even if it's meant to be a promotional item at a zero-dollar value.

What is the use of Bundled Checkbox?



[Salesforce CPQ by Smriti Sharan](#)



If Domino's decides that the "Party Combo" is a special promotion and the individual items in it (the options) should not be priced as they usually are when sold separately, they will check the "Bundled" checkbox in Salesforce CPQ for this combo.



[Salesforce CPQ by Smriti Sharan](#)

This means that even if a “Veggie Pizza” usually costs \$18 on its own, within the “Party Combo,” it’s priced at \$0.00, as are the other items in the combo.

Discounting in Salesforce CPQ

Discounting method	Purpose
Optional Discount	Discount on a product sold as part of a bundle
Volume-Based Discount Schedule	A discount depends on the volume of purchase in % or fixed
Manual Discount	Manual setting of discount
Partner and Distributor Discount	Discount for partners

What is optional Discount in CPQ?



[Salesforce CPQ by Smriti Sharan](#)

Domino's offer a Product Bundle, similar to a combo deal, called the "Pizza Fiesta." This bundle includes a large "Veggie Extravaganza" pizza, "Cheesy Bread," and a soda.

How to set up Optional Discount in CPQ?

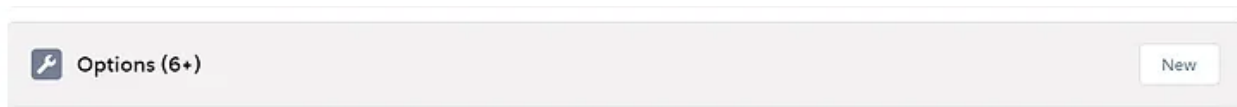
1. Product Bundle Creation: Domino's first establishes a Product Bundle called "Pizza Fiesta" which includes three items:
 - "Veggie Extravaganza" pizza
 - "Cheesy Bread"
 - soda
2. Identifying the Discounted Item: Domino's decides that the "Cheesy Bread" will be 10% cheaper if ordered as part of the "Pizza Fiesta" bundle, compared to its regular à la carte price.
3. Setting Up the Discount in Salesforce CPQ: Domino's admin goes into Salesforce CPQ and navigates to the main "Pizza Fiesta" bundle product in



[Salesforce CPQ by Smriti Sharan](#)

the system. Under the Related tab, they would find all the products that are part of this bundle listed as Product Options.

4. Applying the Optional Discount: The admin locates the “Cheesy Bread” within the list of Product Options for the “Pizza Fiesta” bundle. They then set up a specific pricing indicating that when the “Cheesy Bread” is part of the “Pizza Fiesta” bundle, it triggers a 10% discount on this particular item.



Explain Manual Discount feature in Salesforce CPQ?

Imagine a regular customer places a bulk order for an upcoming party.

They ordered 20 “Veggie” pizzas, each originally priced at \$10. To reward this bulk purchase, the sales manager has decided to offer a special discount. Instead of the regular price, they want the total cost for the 20 pizzas to be \$150, instead of \$200.



[Salesforce CPQ by Smriti Sharan](#)

What is a Non-Discountable Field in Salesforce CPQ?

Domino's is running a special promotion on "Veggie Delight" pizzas, and these are not eligible for further discounts. The 'Non-Discountable' field would be checked for this product in Salesforce CPQ, preventing any additional manual discounts on this item, regardless of the order size.

How to add Unit and Total values to picklist?

We go to the Quote Line object and create a new field of picklist type with the name AdditionalDiscountUnit and the values shown below.



[Salesforce CPQ by Smriti Sharan](#)

Field Label i

Values

Use global picklist value set

Enter values, with each value separated by a new line

Percent
Amount
Unit Override
Total Override

Display values alphabetically, not in the order entered

Use first value as default value i

Restrict picklist to the values defined in the value set i

Field Name i

Now we can choose additional values.

Quote Information

Start Date i End Date i

Additional Disc. (%) i Delivery Method i

#	PRODUCT CODE	PRODUCT NAME	QUANTITY	LIST UNIT PRICE	ADDITIONAL DISC.	NET U
1	LAPTOP15	15" Laptop	1.00	€1,500.00	<input type="text"/> %	
2	CPU22GHZI7	CPU 2.2GHz i7	1.00	Included	<input type="text"/> %	
3	RAM8GB	RAM 8GB	1.00	Included	<input type="text"/> EUR	
4	SSD256	SSD Hard Drive 256GB	1.00	Included	<input type="text"/> Unit	
					<input type="text"/> Total	



[Salesforce CPQ by Smriti Sharan](#)

Explain Partner and Distributor Discount in Salesforce CPQ?

This method is used when we want to give an additional partner discount to the client. The values for these discounts are displayed in the Distributor Discount and Net Unit Price fields on the Quote Line object. They are calculated by the following formulas:

Partner Price = Customer Price – Partner Discount

Net Price = Partner Price – Distributor Discount

Explain Volume Based Discounting?

Domino's wants to give an incentive to customers to order more pizzas during a cricket world cup by offering volume-based discounts.



[Salesforce CPQ by Smriti Sharan](#)



1. Setting up the Discount Schedule:

Discount Schedule **Bulk Keys** Edit Edit Tiers New Contact

Schedule Name: Bulk Keys Aggregation Scope:

Type:

Range:

Discount Unit: Percent

Description:

Advanced Settings

Override Behavior:

User Defined:

Like Price For Amount:

Cross Products:

Cross Orders:

Include Bundled Quantities:

Constraint Field:

Price Scale:

Edit Tiers **Bulk Keys** Help for this Page

Edit Tiers Save Cancel

#	Tier Name	Lower Bound	Upper Bound	Discount (%)
1	First Level	100.00	200.00	10.0000000000
2	Second Level	200		20

Save Cancel



[Salesforce CPQ by Smriti Sharan](#)

The Domino's admin navigates to the "Discount Schedules" tab within Salesforce CPQ and creates a new schedule named "Game Day Special."

- They then click on "Edit Tiers" and set up the volume tiers as follows:
- 1-10 pizzas: 0% discount
- 11-20 pizzas: 10% discount
- 21-30 pizzas: 15% discount
- 31+ pizzas: 20% discount
- They save this tier structure, which will dictate the discount applied based on the quantity of pizzas ordered.

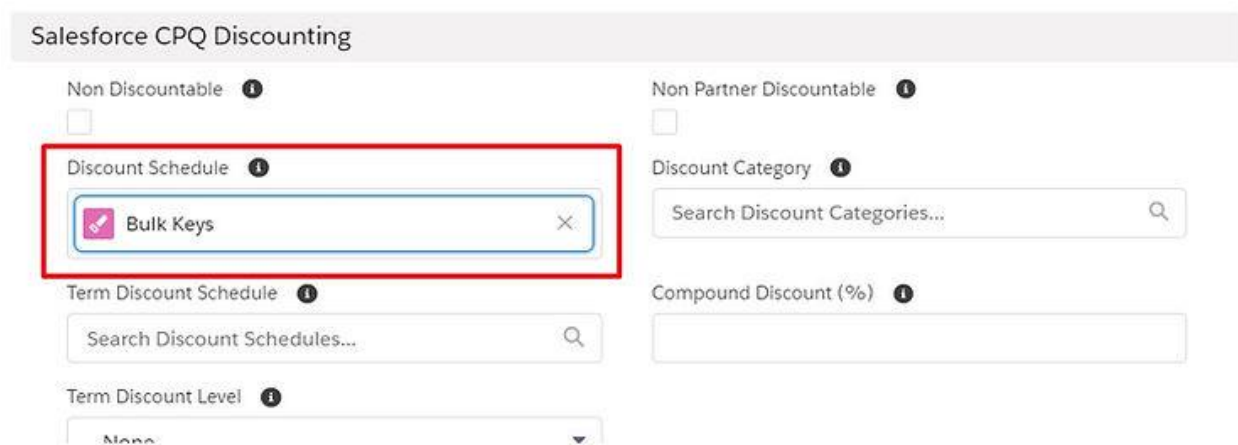
2. Applying the Discount Schedule to Products:

The admin then goes to the "Products" tab and selects the pizzas eligible for this promotion, like "Cheesy Extravaganza" and "Veggie Feast."



[Salesforce CPQ by Smriti Sharan](#)

– In the “Discount Schedule” field on each product, they select the “Game Day Special” schedule they previously created.



3. Placing Orders and Calculating Discounts:

– A customer places an order for 25 “Cheesy Extravaganza” pizzas during the cricket event weekend.

– When the order is entered into the system, the sales rep creates a new quote and adds the 25 pizzas to it.



[Salesforce CPQ by Smriti Sharan](#)

– Upon clicking “Calculate” in Salesforce CPQ, the system automatically recognizes that the quantity falls into the 21-30 tier, thus applying a 15% discount to the order.

– The sales rep sees the regular price and the discounted price side by side, confirming the volume discount was applied correctly.

Q-00019 Edit Quote

Quote Information

#	PRODUCT CODE	PRODUCT NAME	QUANTITY	LIST UNIT PRICE	ADDITIONAL DISC.	NET UNIT PRICE	NET TOTAL	ORIGINAL PRICE
1	RFIDKEYCARD	RFID Key Card	1.00	€1.00		€1.00	€1.00	€1.00

PRICING METHOD: List

UNIT COST: €0.50

MARKUP:

END DATE:

SPECIAL PRICE: €1.00

REGULAR UNIT PRICE: €1.00

CUSTOMER UNIT PRICE: €1.00

SUBSCRIPTION TERM:

OPTIONAL:

PACKAGE PRODUCT CODE:

START DATE:

SUBTOTAL: €1.00

Активация Windows

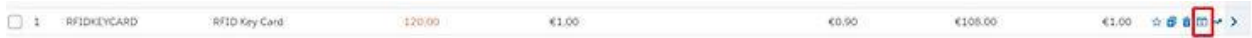
QUOTE TOTAL: €1.00

4. Checking Discount Details:

Next to the “Cheesy Extravaganza” pizza on the quote, a “\$” sign appears to indicate a Discount Schedule is applied.



[Salesforce CPQ by Smriti Sharan](#)



Clicking on it allows the sales rep (and any other user with the necessary permissions) to view the discount tiers, providing transparency on how the final price was calculated.

What is the structure of Discount Schedule?

The typical structure of a discount schedule involves defining specific tiers. Each tier includes tier name, lower bound, upper bound and Discount (percentage/amount) applicable on that tier.

There are two ways of the Discount Schedule Working: –

- Slab
- Range

1. Slab Discount Schedule:

Domino's wants to offer a Slab discount during a local festival. They set the following tiers:



[Salesforce CPQ by Smriti Sharan](#)

- 1-5 pizzas: 0% discount
- 6-10 pizzas: 5% discount
- 11+ pizzas: 10% discount

Using the Slab method, Salesforce CPQ calculates the discount as follows:

The first 5 pizzas get a 0% discount.

The next 5 pizzas (pizzas 6 to 10) receive a 5% discount.

The final 2 pizzas (pizzas 11 and 12) receive a 10% discount.

Each “slab” or tier of pizzas is discounted according to its respective bracket, not the total order quantity.

2. Range Discount Schedule:

Domino’s decides to offer a Range discount during a major sports event.

They set the following tiers:



[Salesforce CPQ by Smriti Sharan](#)

- 1-7 pizzas: 0% discount
- 8-14 pizzas: 7% discount
- 15+ pizzas: 15% discount

Using the Range method, Salesforce CPQ calculates the discount as follows:

- The total 10 pizzas fall within the 8-14 range.
- Therefore, all 10 pizzas receive a 7% discount.

The discount is applied to the total order based on the range in which the total quantity falls.

What are the Steps to Set Up a Discount Schedule?

1. Open Salesforce CPQ application from the app launcher.
2. Type 'Discount Schedule' in the search box of the app launcher.
3. Open the Discount Schedule tab and click on 'new'.



[Salesforce CPQ by Smriti Sharan](#)

4. Set the schedule name,
5. Set the type as Range or Slab according to the requirement.
6. Select the Discount unit from Percent, Amount and price options as per the requirement.
7. For the aggregation scope select Quote or Group as per the requirement.
8. Click on 'Save'.

What are the steps to Create Discount Tiers and Associate the Discount Schedule with Product?

1. Open the Discount Schedule in which we want to create Discount Tiers.
2. Click on 'Edit Tier'.
3. Set the name of first-tier name.
4. Set the lower bound limit of the first tier.
5. Set the upper bound limit of the first tier.
6. Set the discount (percentage/amount).



[Salesforce CPQ by Smriti Sharan](#)

7. Click on + button to add a new tier and repeat the steps from 3 to 6.
8. Repeat step 7 as per the requirement of the tiers needed.
9. Click on Save.
10. Open the Product tab from navigation bar and select the product where we want to associate the Discount Schedule.
11. Click on the 'Edit' button.
12. For Discount Schedule Field enter the created discount schedule name.
13. Click on 'Save' button



[Salesforce CPQ by Smriti Sharan](#)

Quote

Template



[Salesforce CPQ by Smriti Sharan](#)

What are Salesforce CPQ quote templates?

Salesforce CPQ quote templates are pre-defined layouts that determine how quotes are presented to customers. They provide a consistent appearance and help standardize the quoting process.

How do I create a quote template in Salesforce CPQ?

To create a quote template in Salesforce CPQ, navigate to the Quote Templates setup, define template details, configure template content, add dynamic quote terms, preview, and test, and finally activate the template.

Explain quote template structure?

Quote Template Structure includes:



[Salesforce CPQ by Smriti Sharan](#)

1. Quote Template Information

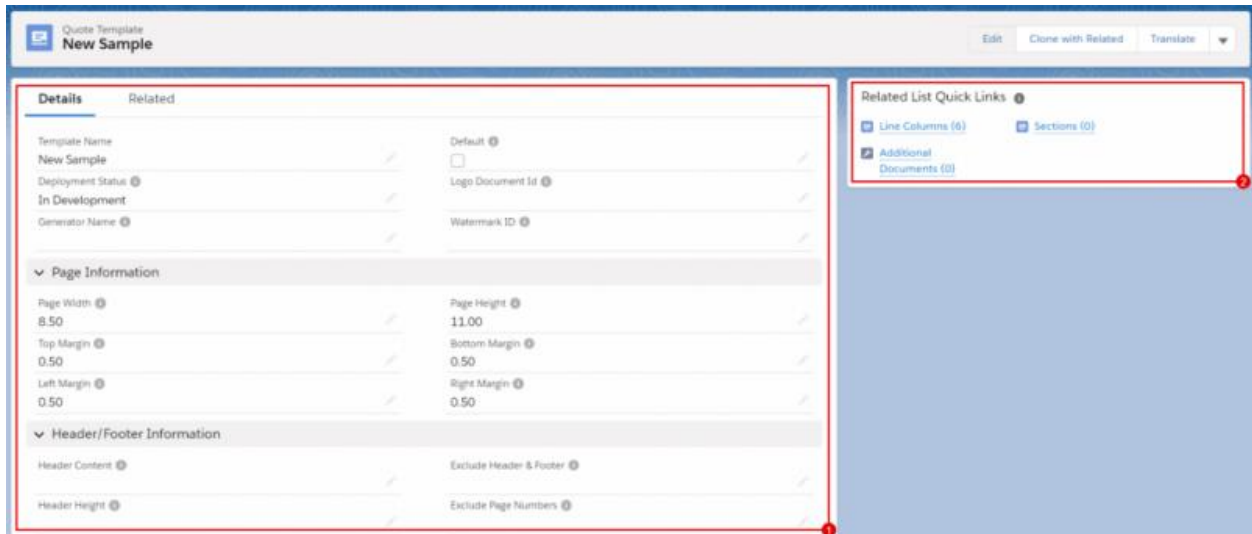
- Page Information
- Header/Footer Information
- Corporate Information
- Style Information
- Group Style Information
- Section Titles
- Print Options

1. Related Lists

- Line Columns
- Template Sections
- Additional Documents



[Salesforce CPQ by Smriti Sharan](#)



What are the steps for creating quote templates?

There are five main steps to follow when creating a quote template:

1. Creating the Quote Template itself
2. Storing the necessary logo in Documents
3. Creating the Template Content
4. Creating the Template Sections



[Salesforce CPQ by Smriti Sharan](#)



[Salesforce CPQ by Smriti Sharan](#)

Salesforce CPQ Contract Management

What is CLM (Contract Lifecycle Management)?



[Salesforce CPQ by Smriti Sharan](#)



Scenario – “Frosty Bliss,” is a popular [frozen yogurt company](#), operates. It is renowned for its mouth-watering flavors like chocolate, strawberry surprise, and magnificent mango, along with a dazzling array of toppings.

Behind the scenes, there’s a web of contracts and agreements running.

These range from [deals with local farmers](#) for fresh strawberries and mangoes, arrangements with chocolate suppliers for the richest cocoa, [leases for the shop spaces](#) in prime locations, and the [health insurance policies](#) for the employees.

CLM ensures all agreements with the fruit farmers are honored during harvest season, that the chocolate supplies are fair trade, and that every



[Salesforce CPQ by Smriti Sharan](#)

employee is safeguarded by their insurance. CLM key in handling contracts.

Explain the steps in the Contract Management Lifecycle?



Using the scenario of “Frosty Bliss,” a popular frozen yogurt company, I’ll explain how the contract lifecycle management (CLM) steps. I will also incorporate Conga (a comprehensive CLM solution) and DocuSign (a



[Salesforce CPQ by Smriti Sharan](#)

digital signature application) so as to give a full understanding of how these tools are utilized in this process.

1. Contract Request: Frosty Bliss plans to introduce 3 new exotic flavors in the next quarter. The legal team uses [Conga to initiate a contract request](#), inputting specifics for 10 new ingredient suppliers, and [detailing required quantities, delivery, and quality specifics](#).
2. Collaboration: Utilizing Conga, the legal team collaborates with the R&D department, which has determined a need for 5 additional unique toppings for the new flavors. They use [Conga's centralized platform to create drafts](#), employing previous [templates](#) but [modifying](#) for the new ingredients and toppings, involving feedback from 7 different departments.
3. Approval: The contract drafts are sent to 12 key internal stakeholders via Conga. Each stakeholder uses the platform's [review feature](#), ensuring alignment with Frosty Bliss's financial allowances and stringent quality benchmarks.



[Salesforce CPQ by Smriti Sharan](#)

4. Negotiation: Negotiations commence with potential suppliers, facilitated by Conga's communication tracking features.

5. Execution (Signing): After finalizing negotiations, contracts are sent for [signing through DocuSign to expedite the process](#). DocuSign's seamless interface allows for immediate and legally binding agreements, cutting down the contract signing duration [from a 2-week to a 3-day turnaround](#).

6. Ongoing Management: Frosty Bliss, utilizing Conga, oversees the contracts' lifecycle, tracking supplier adherence to the agreed-upon standards and timelines. The system [alerts the legal team if suppliers deviate](#) by more than 10% from the contract terms, prompting reviews for potential amendments, renewals, or terminations.

7. Analytical Insights: Conga's advanced analytics provide Frosty Bliss with [real-time insights from the contract data](#), revealing that 85% of



[Salesforce CPQ by Smriti Sharan](#)

suppliers consistently meet contract terms and identifying a 30% uptick in seasonal demand requiring proactive adjustments.

2. How does the integration of Conga and Adobe benefit the end-to-end contract management process within Salesforce?



Integrating Conga and Adobe with Salesforce streamlines the contract management process by facilitating features like

- contract redlining
- electronic signing



[Salesforce CPQ by Smriti Sharan](#)

- automatic versioning of documents

The screenshot shows the Salesforce CPQ interface. At the top, there is a navigation bar with 'Salesforce CPQ' and various menu items like 'Home', 'Accounts', 'Opportunities', 'Quotes', 'Products', 'Reports', 'Dashboards', 'Favorite Product', 'Quote Templates', and 'Quote Terms'. Below the navigation bar, there is a header for the current document: 'QUOTE - Q-03975' and 'View Redlines'. The main content area is split into two parts. On the left, there is a 'View Redlines' sidebar with a 'MODIFIED' section and an 'ORIGINAL' section. The 'MODIFIED' section shows 'Delivery Method' set to 'Download' and 'Payment Terms' set to 'Net 90'. The 'ORIGINAL' section shows the same settings. On the right, there is a document preview. The document is a quotation from 'Conga Contracts' to 'BlueSky Corp'. It includes contact information for Conga Contracts and BlueSky Corp, a table with columns for 'SALESPERSON', 'EXT', 'EMAIL', 'DELIVERY METHOD', and 'PAYMENT TERMS', and a closing section with a signature and a thank you message.

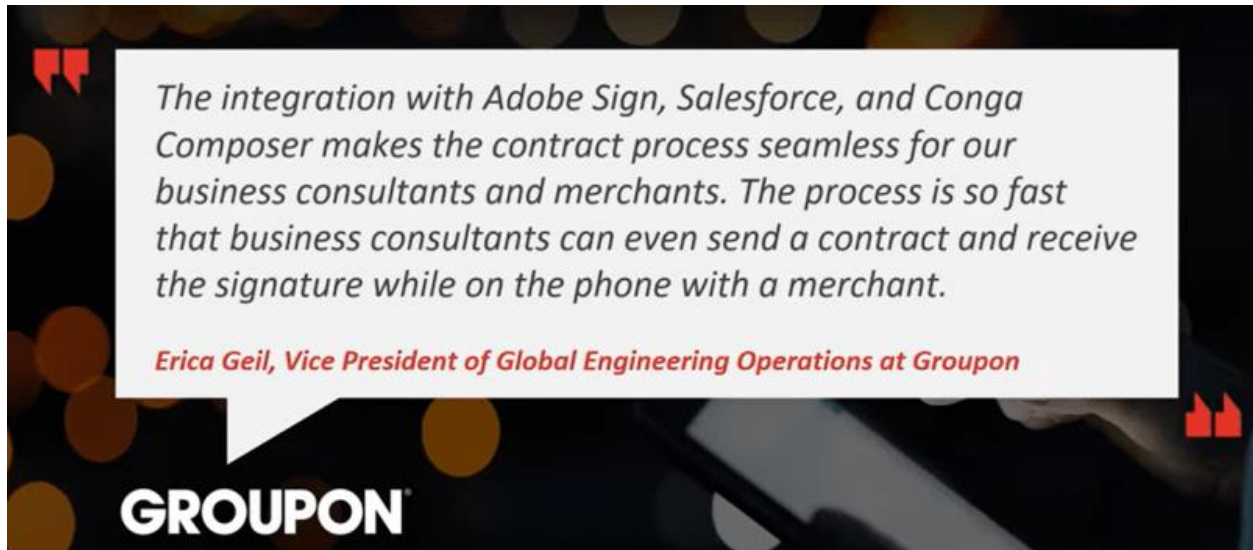
SALESPERSON	EXT	EMAIL	DELIVERY METHOD	PAYMENT TERMS
Sally Sales	X	steelbrick_demo@salesforce.com	ShipDownload	Net 15Net 30

This ensures a seamless transition from lead to cash, enhancing accuracy, and speeding up the contract negotiation and signing process.

3. Could you discuss the specific benefits companies experience when integrating Adobe's electronic signature solutions?



[Salesforce CPQ by Smriti Sharan](#)



Integrating Adobe's e-signature solutions drastically [reduces the signature cycle time](#), from days to mere minutes in some cases. It's not just about speed, though. The benefits also include improved accuracy, enhanced compliance, increased adoption by sales teams, and more professional experience for both employees and clients.



[Salesforce CPQ by Smriti Sharan](#)

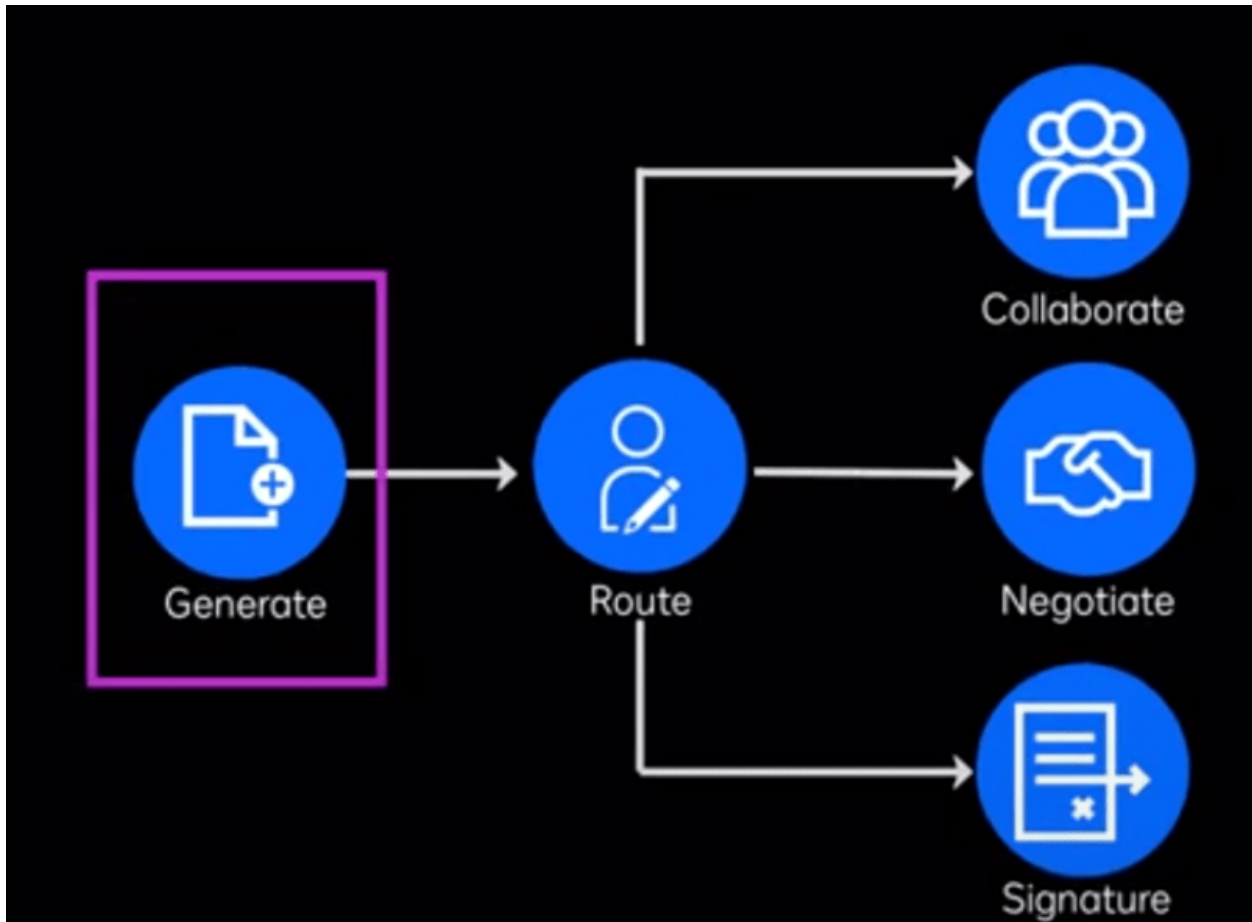
A screenshot of an Adobe Sign document interface. The top left shows the Adobe Sign logo. Below it, there's a header with 'Alternative actions -' and a document ID 'Q-03975-20170713-1031'. The main content area is labeled 'Page 1 of 2'. The form contains the following fields: 'Signature: _____', 'Effective Date: ___/___/___', 'Name (Print): _____', and 'Title: _____'. Below these fields, there is a message: 'Please sign and email to Sally Sales at steelbrick_demo@salesforce.com'. A 'THANK YOU FOR YOUR BUSINESS!' message is centered below the form. At the bottom left, there is a yellow 'Next' button. At the bottom right, there is a yellow box with the text 'Signature: Click here to sign'.

e-Signature helps in the following ways:

- Set it without programmers.
- Simplify the workflow for reps.
- Stay in salesforce.
- Send and sign on any device at any time.
- Reduce errors and improve compliance.
- We can customize it.



[Salesforce CPQ by Smriti Sharan](#)



4. What is the disadvantage of not integrating with third-party tools such as Adobe Sign or DocuSign for e-signature?

1. Manual Processes and Delays: Without integration, documents and contracts created or managed in Salesforce would need to be manually sent for signature, downloaded, or transferred to Adobe products.



[Salesforce CPQ by Smriti Sharan](#)

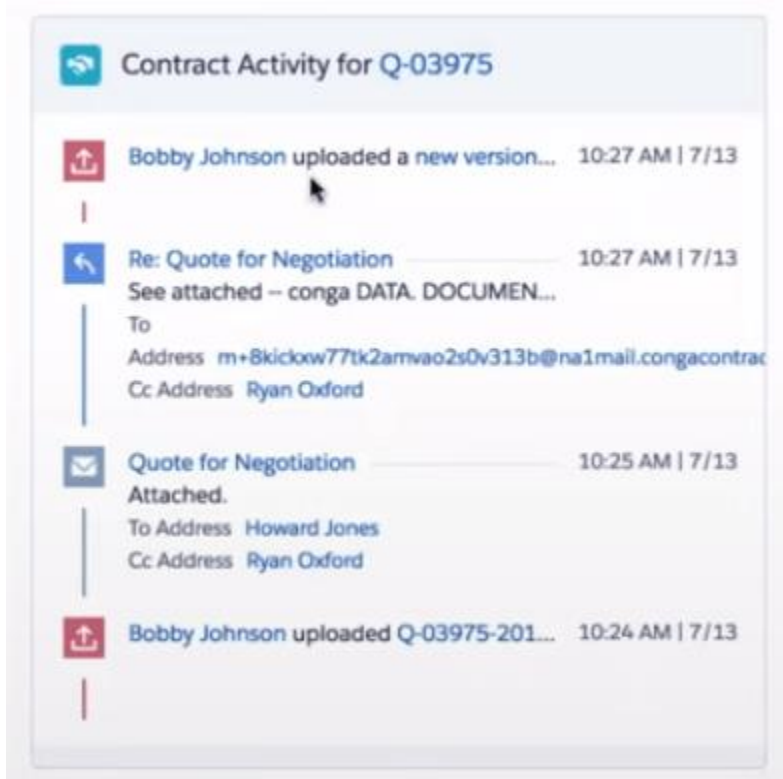
2. increased Error Potential: Manual data entry or transfer between Salesforce and Adobe products increases the risk of human errors, such as inputting incorrect information or sending the wrong document version for signature.

3. Inefficiency and Poor User Experience: Users would have to switch between systems to complete tasks, leading to inefficiency and a disjointed user experience.

How do Conga and Salesforce CPQ handle dynamic document generation and management during the negotiation phase?



[Salesforce CPQ by Smriti Sharan](#)



Salesforce CPQ is adept at automating the quote generation process with accurate pricing, while Conga offers [advanced document generation](#) and management features. During negotiations, [changes can be tracked, and new versions created](#) and stored within Salesforce, ensuring all parties have access to the latest information. Also, the system supports various file formats, contributing to its versatility in real-world scenarios.



[Salesforce CPQ by Smriti Sharan](#)

How does automation in Salesforce help in managing renewals and avoiding contract lapses?

Automation within Salesforce significantly improves renewal management by proactively [tracking contract end dates and triggering alerts for upcoming renewals](#). This ensures timely engagements with customers to initiate renewal discussions.

How do Conga's AI-based capabilities enhance contract management within Salesforce?

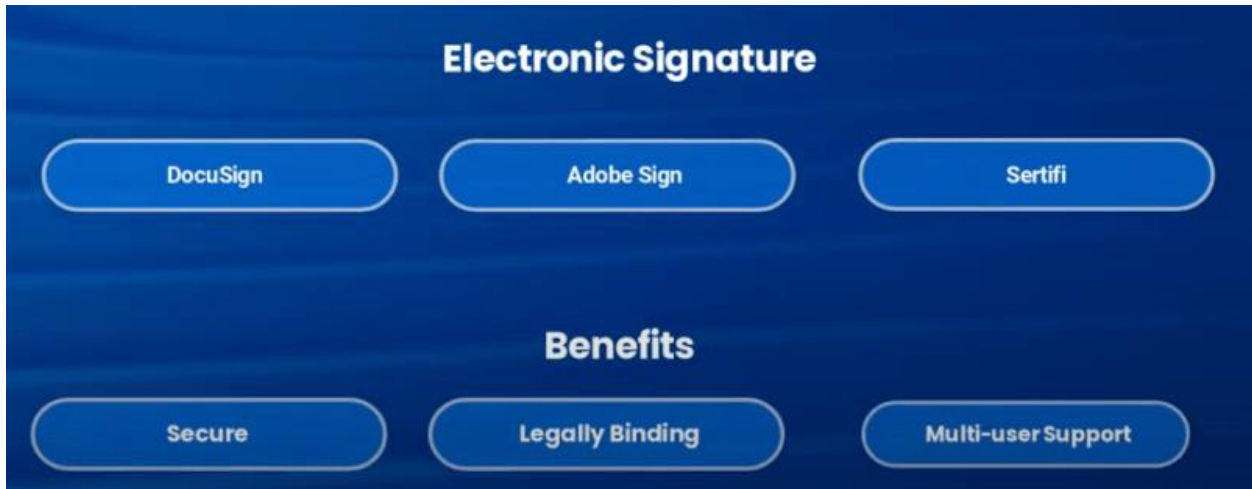
By using machine learning algorithms, Conga can identify:

- patterns and anomalies in contracts
- suggest optimal terms.
- flag potential risks
- deviations from standard clauses

Which is better for Salesforce to integrate – DocuSign or Adobe Sign as electronic signature platforms?



[Salesforce CPQ by Smriti Sharan](#)



DocuSign: Known for its ease of use, DocuSign offers features like [reusable templates](#), customizable [branding](#), [bulk sending](#), and [advanced fields](#). The user interface is intuitive, which is favorable for businesses looking for straightforward solutions.



[Salesforce CPQ by Smriti Sharan](#)

DocuSign eSignature for Salesforce

Send agreements for signature, capture information, track, and store without leaving Salesforce

DocuSign eSignature for Salesforce CPQ

Generate quotes and proposals with Salesforce data and quickly send them for signature

DocuSign CLM for Salesforce

Generate agreements, centralize negotiations, create sophisticated workflows, and store and search

DocuSign Gen for Salesforce

Automatically generate polished, configurable agreements with just a few clicks from Salesforce

DocuSign Negotiate for Salesforce

Accelerate the creation, negotiation, and approval of customized agreements from Salesforce

Adobe Sign: This platform integrates well with various Adobe products and offers strong workflow capabilities.

Best practice is to create contract from order or from opportunity and why?



[Salesforce CPQ by Smriti Sharan](#)

Could you elaborate on how automation can be used in the sales process?

Auto-generating Orders: Once a Quote is accepted, an automation can trigger the creation of an Order, reducing manual data entry.

Discount Approval Processes: Automation can be set up to initiate approval workflows for discounts that exceed certain thresholds, ensuring proper oversight.

Contract Creation: Contracts can be generated automatically when an Order is approved, saving time, and ensuring consistency.

Syncing Quote Data: Automation can keep Quote and Opportunity data in sync, providing a real-time view of the sales pipeline.

In the sales process, what triggers the creation of an Order from a Quote, and what information does an Order capture?



[Salesforce CPQ by Smriti Sharan](#)

An **Order** is typically created from a **Quote** when the customer accepts the Quote, signaling their commitment to purchase. An Order captures essential information, including product details, quantities, pricing, shipping information, and terms of sale.

Can you explain the role of Contracts in Salesforce in more detail?

What types of information are typically included in a Contract record?

Contracts formalize agreements between the company and the customer.

A Contract record in Salesforce typically includes information such as:

- Terms and conditions
- Service-level agreements (SLAs)
- Billing terms
- Start and end dates
- Renewal terms



[Salesforce CPQ by Smriti Sharan](#)

- Pricing details
- Obligations and commitments

Contracts serve as legally binding documents that outline the terms of the relationship between the company and the customer.

What are the typical steps involved in automating the creation of an Order from a Quote in Salesforce?

- Defining the triggering event, such as Quote acceptance.
- Configuring Flow to initiate Order creation.
- Mapping relevant data from the Quote to the new Order record.
- Implementing error handling and validation rules to ensure data accuracy.

This automation reduces manual effort and ensures a smooth transition from Quote to Order.



[Salesforce CPQ by Smriti Sharan](#)

What are some key considerations for companies when deciding whether to implement CPQ (Configure, Price, Quote) in addition to Sales Cloud?

- The complexity of their product offerings and pricing structures.
- The need for streamlined quoting processes and accurate pricing calculations.
- The desire for a more efficient and error-free quoting experience for sales teams.

How does the contracting process work in Salesforce CPQ, particularly from opportunities or orders?

In Salesforce CPQ, the contracting process [can be initiated directly from an opportunity or order by using the contractor checkbox](#). Once this box is checked, it triggers the automation within Salesforce CPQ.



[Salesforce CPQ by Smriti Sharan](#)

Can you explain the role of the ‘renewal model’ in Salesforce CPQ and how it affects contracts?

The ‘renewal model’ field in Salesforce CPQ is key to defining how the [system tracks the products we quote and sell on our accounts](#). We can set the renewal model in the subscriptions and renewals package settings.

Salesforce CPQ applies that value by default to the renewal model field on the accounts. However, we can override this setting at the account level if needed. This model determines how subscription products and assets are handled during the renewal process.

During the contracting process, what are ‘subscribed assets,’ and how are they different from other assets or subscriptions?

‘Subscribed assets’ refer to a specific record generated during the contracting process in Salesforce CPQ. The ‘subscribed asset’ is a [junction object that links the asset and the subscription](#), providing a detailed record of hardware products.



[Salesforce CPQ by Smriti Sharan](#)

How does Salesforce CPQ handle the conversion of quoted products into assets or subscriptions?

For subscription products, CPQ automatically creates subscription records, while for perpetual products, it creates asset records based on the asset conversion criteria set for each product. For instance, if a product is marked “one per unit” for asset conversion, CPQ will create an **individual asset record for each unit sold**. This automation streamlines the sales process, ensuring accurate record-keeping for different types of sales.

Can you elaborate on the process of creating contracts from opportunities in Salesforce CPQ?

To create a contract from an opportunity in Salesforce CPQ, certain conditions and settings must be met.

- the opportunity must have **at least one subscription product**.
- there **must be a primary quote** linked to the opportunity,



[Salesforce CPQ by Smriti Sharan](#)

- contract-based renewal method must be set on the opportunity's account.
- the '[contracted](#)' checkbox on the opportunity is selected.

Salesforce CPQ kicks off the automation process, creating subscription records for subscription products and asset records for perpetual products based on the quote lines.

Can you discuss the role of “Asset Conversion” during the contracting process?

“Asset Conversion” is a field determining whether a product should be converted to an asset or not. This is particularly important for products that we don't need to track as assets, like [complementary products or swags included in a bundle](#). Users can set this field to “[None](#)” for such items.

However, for main products, they can choose “One Per Unit” or “One Per Code Line,” which instructs Salesforce CPQ to create either individual



[Salesforce CPQ by Smriti Sharan](#)

asset records for each unit sold or a single asset record for all units included in a code line, respectively.

Could you elaborate on the differences between “Contract-Based” and “Asset-Based” renewal models?

The “Contract-Based” renewal model is used when you need to keep detailed records of [subscription products](#), particularly their start and end dates. This model triggers the creation of a contract record and corresponding subscription or asset records.

Asset-Based” renewal model is more appropriate when you don’t sell subscription products or when you use only “percentage of total” products to represent them. In this model, only [asset records are created, which are then associated with your account.](#)

Can you elaborate on the difference between ‘One Per Unit’ and ‘One Per Quote Line’ in the Asset Conversion setting?



[Salesforce CPQ by Smriti Sharan](#)

In the Asset Conversion setting, 'One Per Unit' and 'One Per Quote Line' are options that dictate how products are converted to assets in Salesforce CPQ.

- 'One Per Unit' creates an individual [asset record for each unit of a product sold](#), beneficial for tracking items with unique identifiers like serial numbers.
- 'One Per Quote Line' consolidates [all units of a product sold into a single asset record](#), useful for general tracking without the need for individual item details.

How does Salesforce CPQ handle zero-quantity subscription products during the contracting process?

Salesforce CPQ has a unique approach to handling subscription products with zero quantity. During the contracting process, if an opportunity includes a subscription quote line with a quantity of zero, CPQ will still create a [subscription record with zero quantity on the contract](#). This method



[Salesforce CPQ by Smriti Sharan](#)

ensures accurate representation and tracking of all quoted products, even if their quoted quantity is zero.

How does the 'Asset Conversion' field work during the contracting process, and can you provide examples of when to use different settings?

'None': Use this when you do not want the product to convert into an asset. This is common for items that don't require after-sale tracking or service, such as promotional items or swag.

'One Per Unit': This setting creates one asset record for each unit of the product sold. This is particularly useful for items that need individual tracking after the sale, often used for [products with serial numbers](#) or that require individual service/support.

'One Per Quote Line': With this, regardless of the quantity sold, only one asset record is created for each quote line. This is efficient for [bulk products or when individual tracking per unit is unnecessary](#).



[Salesforce CPQ by Smriti Sharan](#)

Can the contracting process be initiated from orders in Salesforce CPQ?

Create a contract from the order by selecting the [order's Contracted field](#). This process is synchronous by default. However, if you have many order products to contract, you can use Salesforce CPQ's asynchronous contracting process instead.

What is the billing process in Salesforce CPQ?

[Salesforce Billing is an add-on package](#) that inherits key records and information from Salesforce CPQ. After a sales rep finalizes a quote and orders it within Salesforce CPQ, Salesforce Billing picks up the order record for invoicing, payment, and revenue recognition.

How do I renew a contract in Salesforce CPQ?

To renew contracts from your account, [go to your account's Contracts related list, select all the contracts to renew, and then click Renew](#)



[Salesforce CPQ by Smriti Sharan](#)

Contracts. Salesforce CPQ asks you to pick a master contract. The renewal quote inherits its end date from the master contract's end date.

What happens to products on a quote once a deal is contracted in Salesforce CPQ?

Once a deal is finalized / “contracted”, the details of the products on the quote are locked in. This means they are officially recorded and can't be changed casually, as they are part of a formal agreement between you and your customer. These details are often used to create other records that help track what the customer has bought, like “assets” or “subscriptions.”

Can the renewal model on an account differ from the package's renewal model?

Yes, it can. Imagine the “renewal model” as a set of rules for how a customer can [continue their service after their initial period ends](#) (like extending a rental).

- These rules can be standard – coming from the “package,”



[Salesforce CPQ by Smriti Sharan](#)

- customized for a specific customer – the “account”

What records are created when an opportunity or order is contracted in Salesforce CPQ?

When “opportunity” or “order” is finalized, Salesforce CPQ creates a “contract” record, which is like a formal written agreement. It also creates “asset” or “subscription” records, if necessary, to keep track of what the customer bought and how long or how they can use it.

How does Salesforce CPQ manage amendments and renewals?

Amendments and renewals are like [updates to the original agreement](#). If a customer wants to change something after the deal is made (amendment) or continue their service after their subscription period ends (renewal),

What information can be tracked for subscriptions in Salesforce CPQ?



[Salesforce CPQ by Smriti Sharan](#)

– For subscriptions, Salesforce CPQ can keep tabs on several important details, like

- what product the subscription is for
- how long it lasts.
- the costs
- the renewal date (when the customer needs to decide if they want to continue),
- any specific terms or rules about the subscription.

What happens to original quote lines after contracting an opportunity in Salesforce CPQ?

The original quote lines (the initial product and pricing details offered to the customer) are preserved, meaning they're kept as they were for reference.

However, they're also used as the basis for the official agreement or

“contract.” [They're like a snapshot of what was offered and agreed upon initially.](#)



[Salesforce CPQ by Smriti Sharan](#)

What does it mean if the asset conversion field for a product is set to “none”?

If the “asset conversion” is set to “none,” it means that when the product is sold, it won’t be turned into an “asset” record in the system. It’s like saying, “We don’t need to keep long-term track of this product as a permanent item for the customer.”

What happens when an opportunity is marked as “Closed Won” in Salesforce CPQ?

It means the deal is successfully agreed upon with the customer.

Salesforce CPQ then gets to work finalizing the details, creating a “contract” and any “asset” or “subscription” records if needed, based on what the customer agreed to buy.

How does the “Contactor” checkbox influence the automation process in Salesforce CPQ?



[Salesforce CPQ by Smriti Sharan](#)

– The “Contractor” checkbox is a bit like telling the system, ” this deal is ready to be made official.” When checked, it triggers Salesforce CPQ to start certain automated processes, like creating the formal “contract” and other related records. It’s a way of moving the deal to the next stage in the system.

How does Salesforce track contract expiration dates for renewal purposes?

Salesforce allows users to track contract expiration dates by storing this information in the Contract object. Each contract record has a field named “Contract Term” (number of months) and “Start Date.” The system calculates the “End Date” by adding the contract term (duration) to the start date.

What’s the standard process for contract amendments in Salesforce?

The standard process for contract amendments in Salesforce typically involves creating an Amendment Case or using the [“Amend” button](#). This



[Salesforce CPQ by Smriti Sharan](#)

action creates a new quote record related to the original contract, allowing users to make necessary changes like adding or removing products, changing pricing, or altering terms. Once the amendment quote is finalized and approved, it can be synced with the original contract, effectively updating it.

How can you create a renewal opportunity in Salesforce?

Renewal opportunities can be created manually by users or automatically through process automation. For automatic creation, you would typically set criteria that identify when a contract is due for renewal (e.g., a certain number of days before the contract's end date), and then define an action to create a new opportunity with specific details (like stage, amount, close date, etc.) related to the renewal.

Can you amend multiple contracts simultaneously in Salesforce?

Standard Salesforce functionality [doesn't allow for the bulk amendment](#) of multiple contracts simultaneously due to the complexity and unique details



[Salesforce CPQ by Smriti Sharan](#)

involved in contract amendments. However, individual contracts can be amended.

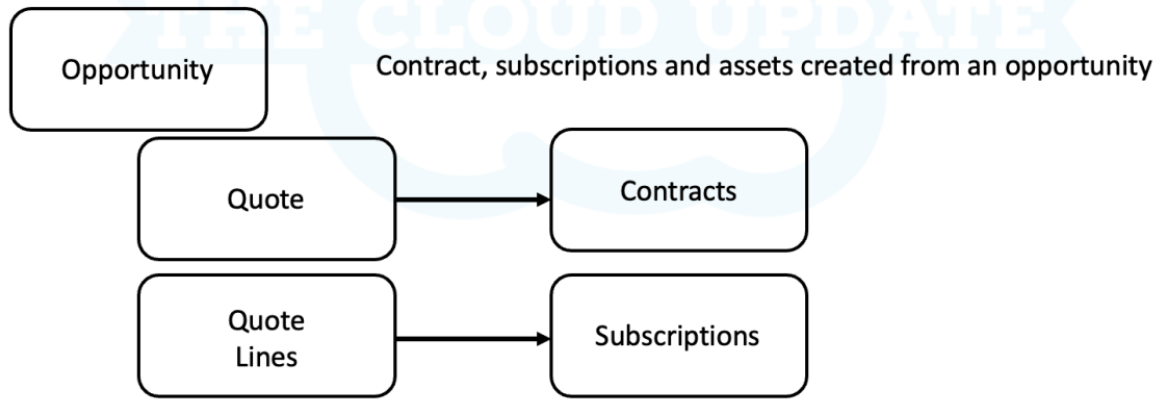
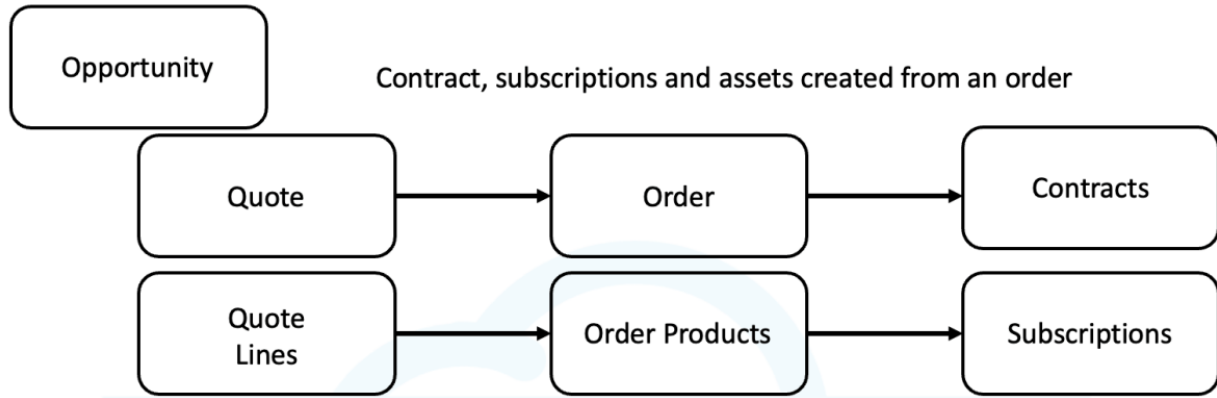
How do you handle contract renewals with multiple amendments in Salesforce?

For contracts with multiple amendments, each change is typically tracked with a separate amendment record related to the original contract. Upon renewal, you will consolidate all changes from the original contract and its amendments into the new renewal contract. Salesforce CPQ can streamline this process, ensuring that all amendments are reflected accurately in the renewal quote and subsequent contract.

Contract Created from Order vs Contract Created from Opportunity?



[Salesforce CPQ by Smriti Sharan](#)



Salesforce Contracts



[Salesforce CPQ by Smriti Sharan](#)

Contract **00000100**

Account Name	Status	Contract Start Date	Contract End Date	Contract Term (months)
Meta	Draft	12/7/2022	12/6/2025	36

Progress bar: [Green bar with checkmarks]

Related Details

Subscriptions (1) New

Subscription #	Product	Start Date	End Date
SUB-0000000	Warranty - Yearly	12/7/2022	12/6/2025

[View All](#)

Now that your contract has been generated, you'll be able to manage the contract lifecycle from the record.

Once generated your contract will have some values from the quote it was generated from.

Explain Contract Start Date and Contract End Date?

- *Contract Start date* will be the start date that was selected on your quote. If your quote did not have a start date, no contract is generated.



[Salesforce CPQ by Smriti Sharan](#)

- *Contract End Date* will be calculated from your contract start date + subscription term.

How do renewals and amendments work in Salesforce CPQ?

In Salesforce CPQ, renewals and amendments are ways to update customer contracts. A renewal is when a customer decides to continue their contract after it ends, kind of like renewing a library book so you can keep it longer.

An amendment is when you change something in the contract while it's still going on, like if a customer wants more services or needs to cancel something they don't use anymore.


How to create a renewal opportunity and quote?

- Create a renewal opportunity by clicking on 'renewal forecast' checkbox on contract.
- Create a quote for renewal opportunity by clicking the 'renewal quoted' checkbox on contract.



[Salesforce CPQ by Smriti Sharan](#)

▼ Renewal Information

Renewal Forecast 

Renewal Quoted 

What happens when 'Contracted' checkbox is checked on opportunity?

When 'contracted' checkbox is checked then






- A new contract gets created.

Opportunity **New Biz** Stage changed successfully.



▼ Sales Information

Expected Revenue	Amount
USD 234,936.23	USD 234,936.23

▼ Deal Desk

ARR 	ACV (1st Year) 
USD 228,760.00	USD 228,760.00
MRR 	TCV 
USD 19,063.33	USD 234,936.23
NRR 	
USD 6,176.23	

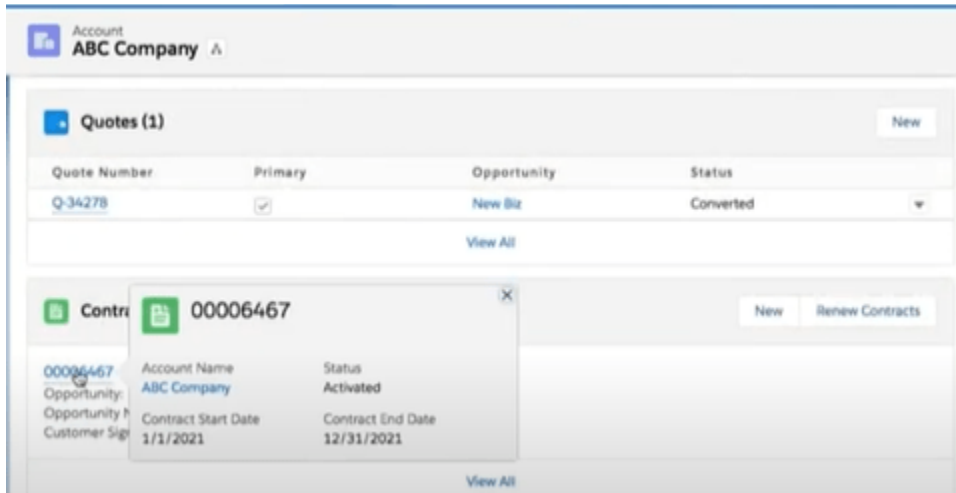
▼ Salesforce CPQ Information

Create Contracted Prices 	Contracted 
<input type="checkbox"/>	<input checked="" type="checkbox"/>



[Salesforce CPQ by Smriti Sharan](#)

- The created Contract is attached to the related list of Account.



- Subscriptions and Assets are attached to the related list of Account



[Salesforce CPQ by Smriti Sharan](#)

Account **ABC Company** ▲

[View All](#)

Orders (0) New

Subscriptions (4) New

Subscription #	Product	Quantity	Net Price
SUB-0038348	Time and Expense SaaS Suite	1.00	USD 0.00
SUB-0038349	T&E Suite Admin License	1.00	USD 960.00
SUB-0038350	T&E Suite Time and Expense U...	350.00	USD 648.00
SUB-0038351	Expense Reimbursement Block	1.00	USD 1,000.00

[View All](#)

Assets (2) New

Asset Name	Product Code	Quantity	Price
Additional Expense Report F...	S-TE-ADDR	1.00	USD 1.23
T&E Admin Training	S-TE-ADTR	1.00	USD 550.00

What are some important fields on contract?

- Account Name
- Status
- Contract Start Date
- Contract End Date
- Contract Term



[Salesforce CPQ by Smriti Sharan](#)

Contract 00006467				
Account Name ABC Company	Status Activated	Contract Start Date 1/1/2021	Contract End Date 12/31/2021	Contract Term (months) 12

Why are only ‘Subscriptions’ are shown but Assets not shown as related list of contracts?

Assets are only one time purchase so they are not added to contracts.

Contract **00006467**

Activated

Related Details

Subscriptions (4) New

Subscription #	Product	Quantity	Start Date
SUB-0038348	Time and Expense SaaS Suite	1.00	1/1/2021
SUB-0038349	T&E Suite Admin License	1.00	1/1/2021
SUB-0038350	T&E Suite Time and Expense U...	350.00	1/1/2021
SUB-0038351	Expense Reimbursement Block	1.00	1/1/2021

[View All](#)

Co-Terminated Quotes (0) New



[Salesforce CPQ by Smriti Sharan](#)

Can you discuss the standard process for creating and managing a renewal in Salesforce CPQ?

So, imagine a customer's contract is about to end, and we want them to continue using our service. In CPQ, we create a "renewal opportunity" which is like a reminder that we need to get them to sign up again. Then, we make a "renewal quote" with all the pricing info. We can change things if we need to, like the price or what's included. Once everything looks good, and the customer agrees, we update the contract to show they're staying with us.

What are some best practices or advanced strategies for managing early renewals in CPQ?

Early renewals are when customers decide to renew their contract before it ends. This can be good for them because they might get better deals. For managing these in CPQ, it's smart to use certain fields that control pricing, like "renewal pricing method" which decides how we price renewals. We



[Salesforce CPQ by Smriti Sharan](#)

should also check settings that control when the renewal info gets sent out so that it's not too early or too late. And always double-check that we're not charging customers extra by accident!

Can you handle bulk amendments or bulk renewals in Salesforce CPQ, and if so, how?

Yes, we can! Handling lots of changes or renewals at once is called "bulk" processing. In Salesforce CPQ, we use special tools that let us update many records at the same time, instead of one by one. This can be done through data loaders or by using CPQ's own bulk features. It's like going through a fast checkout at the supermarket when you have lots of items. But we have to be extra careful to make sure that all the updates are correct and that nothing gets mixed up.

How do you deal with proration on amended subscriptions in Salesforce CPQ?



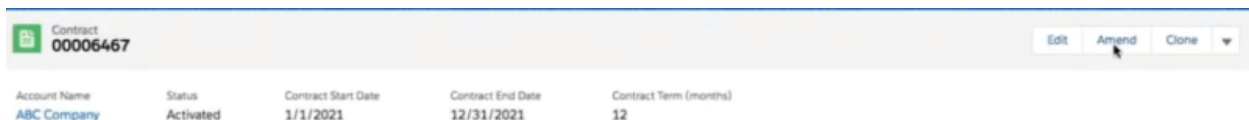
[Salesforce CPQ by Smriti Sharan](#)

“Proration” is when you adjust the price based on the time. Like, if a customer adds a service halfway through their contract, they shouldn’t have to pay the full price for that month. In Salesforce CPQ, we set up proration rules that automatically figure out these adjustments. This makes sure customers are charged fairly, and everything adds up correctly at the end of the day.

Can you walk us through the steps you take for adjusting a contract when a customer is being double-charged due to an early renewal?

If a customer’s being charged twice because they renewed early,

1. First, I go into their account and find the current contract.
2. Then, I click ‘Amend’ to start making changes. I need to give them back the money for the time they were double charged, so I set the start date for the changes to when the new period begins.





[Salesforce CPQ by Smriti Sharan](#)

1. Once you click on amend button you see list of subscriptions which can be ammended

00006467 Amend Contract Amend Cancel
 Displaying 4 of 4 subscriptions

SUBSCRIPTION #	PRODUCT	QUANTITY
SUB-0038348	Time and Expense SaaS Suite	1.0
SUB-0038349	T&E Suite Admin License	1.0
SUB-0038350	T&E Suite Time and Expense User License	350.0
SUB-0038351	Expense Reimbursement Block	1.0

1. Once we click on amend button an amendment opportunity is created in the backed
2. You are taken to Quote line Editor screen where you can make changes.

Q-34497 | Edit Quote
Total: USD 0.00

Add Products Save Quick Save Cancel Calculate Delete Lines Add Group

Quote Information

Start Date: 1/1/2021 End Date: 12/31/2021 Subscription Term:

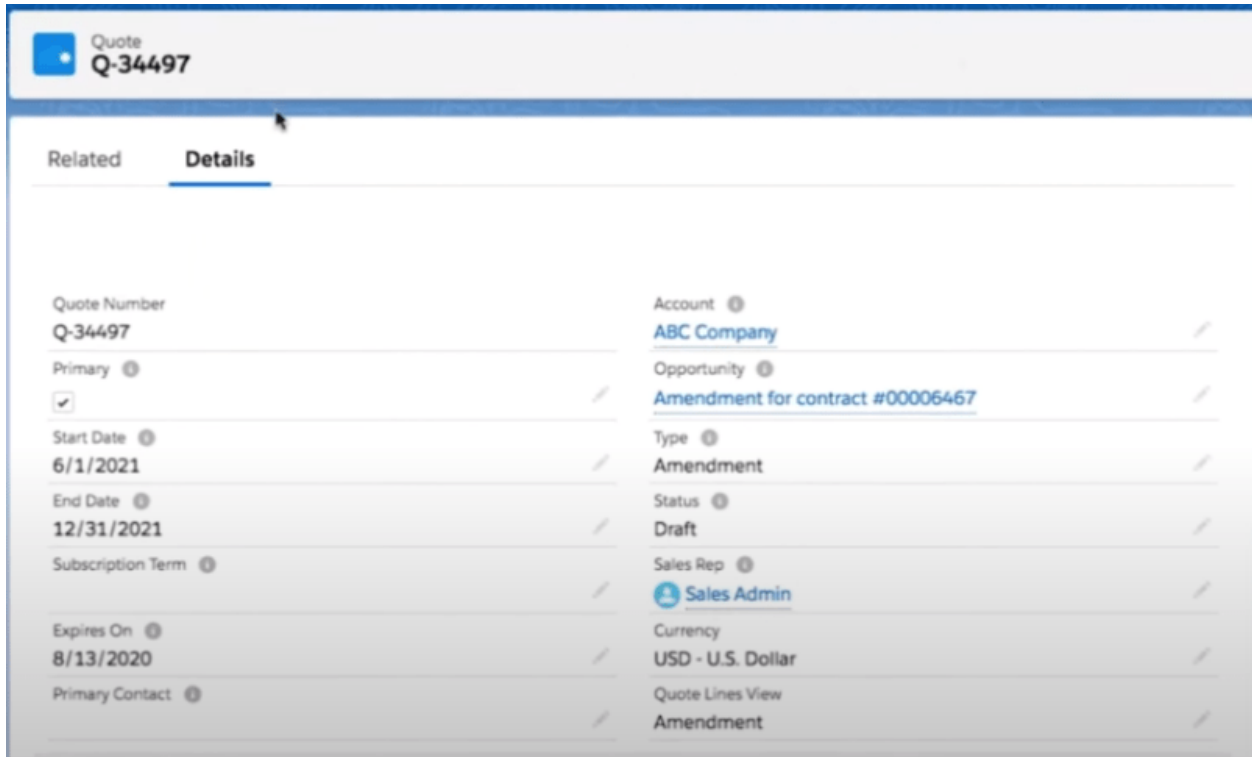
Additional Disc. (%): Target Customer Amount: Quote Lines View: Amendment

#	PRODUCT NAM...	PRIOR QUANTITY	QUANTITY	LIST UNIT PRICE	ORIGINAL DISCOUNT...	ADDITIONAL DISC.	NET UNIT PRICE	NET TOTAL
1	Time and Expense SaaS Suite	1	1	USD 0.00	0.00%		USD 0.00	USD 0.00
2	T&E Suite Admin License	1	1	USD 80.00	0.00%		USD 960.00	USD 0.00
3	T&E Suite Time and Expense User License	350	350	USD 60.00	0.00%		USD 720.00	USD 0.00
4	Expense Reimbursement Block	1	1	USD 0.00	0.00%		USD 0.00	USD 0.00
5	Additional Expense Report Fee	1	1	USD 1.23	0.00%		USD 1.23	USD 0.00
6	T&E Admin Training	1	1	USD 550.00	0.00%		USD 550.00	USD 0.00



[Salesforce CPQ by Smriti Sharan](#)

3. The quote will be created and attached to Amendment Opportunity.



1. You will not see all the products but only the products with changes made.



[Salesforce CPQ by Smriti Sharan](#)

A screenshot of a Salesforce CPQ Opportunity record. The record is titled 'Amendment for contract #00006467'. The main content area is divided into several sections: 'Sales Information' showing 'Expected Revenue' of USD 618.50 and 'Amount' of USD 6,185.00; 'Deal Desk' with metrics for ARR (USD 960.00), MRR (USD 0.00), NRR (USD 5,625.00), ACV (1st Year) (USD 960.00), and TCV (USD 6,185.00); and 'Salesforce CPQ Information' with checkboxes for 'Create Contracted Prices' and 'Contracted'. On the right-hand side, there are two related lists: 'Quotes (1)' showing a quote with number 'Q-34497', net amount 'USD 6,185.00', and status 'Draft'; and 'Products (2)' showing 'T&E Suite Admin License' with quantity 1.00 and sales price USD 560.00, and 'Professional Services Hours (Blended Rate)' with quantity 25.00 and sales price USD 225.00. At the top right of the record, there are buttons for '+ Follow', 'New Quote', 'Edit', and 'Clone'.

Note – Amendment opportunity doesn't create additional contract but keep all the changes to existing contract.

1. When you go to the contract, the new amendment quote will be added as related list as 'Co – Terminated Quotes'



[Salesforce CPQ by Smriti Sharan](#)

Contract 00006467

Related Details

Subscriptions (5) New

Subscription #	Product	Quantity	Start Date
SUB-0038348	Time and Expense SaaS Suite	1.00	1/1/2021
SUB-0038349	T&E Suite Admin License	1.00	1/1/2021
SUB-0038350	T&E Suite Time and Expense U...	350.00	1/1/2021
SUB-0038351	Expense Reimbursement Block	1.00	1/1/2021
SUB-0038352	T&E Suite Admin License	1.00	6/1/2021

[View All](#)

Co-Terminated Quotes (1) New

Quote Number	Net Amount	Status	Created By
Q-34497	USD 6,185.00	Converted	Sales Admin

[View All](#)

How do you ensure accuracy when crediting back a customer during an amendment due to an early renewal?

I make sure that the start date I use for the credit matches the start of their new contract. Also, I only set to zero the products that are being renewed. If they're not renewing everything, you must pay close attention and only select the right products.



[Salesforce CPQ by Smriti Sharan](#)

Could you explain the importance of the 'Amend' function in Salesforce CPQ during the contract renewal process?

The 'Amend' function is super important when we're dealing with contracts. It lets us make changes to a contract that's already going on, like when we need to give a customer their money back because they renewed it early.

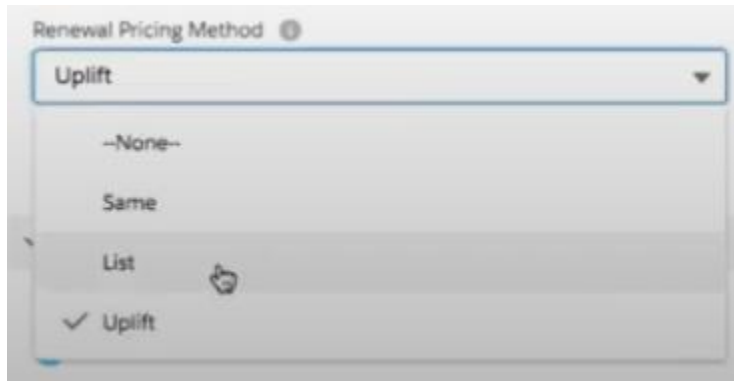
In your experience, what are the common challenges you face during the early renewal process and how do you handle them?

- One big challenge is making sure you don't mix things up when some products are being renewed and others aren't.
- Another challenge is making sure the credit is right.

What are renewal pricing methods?



[Salesforce CPQ by Smriti Sharan](#)

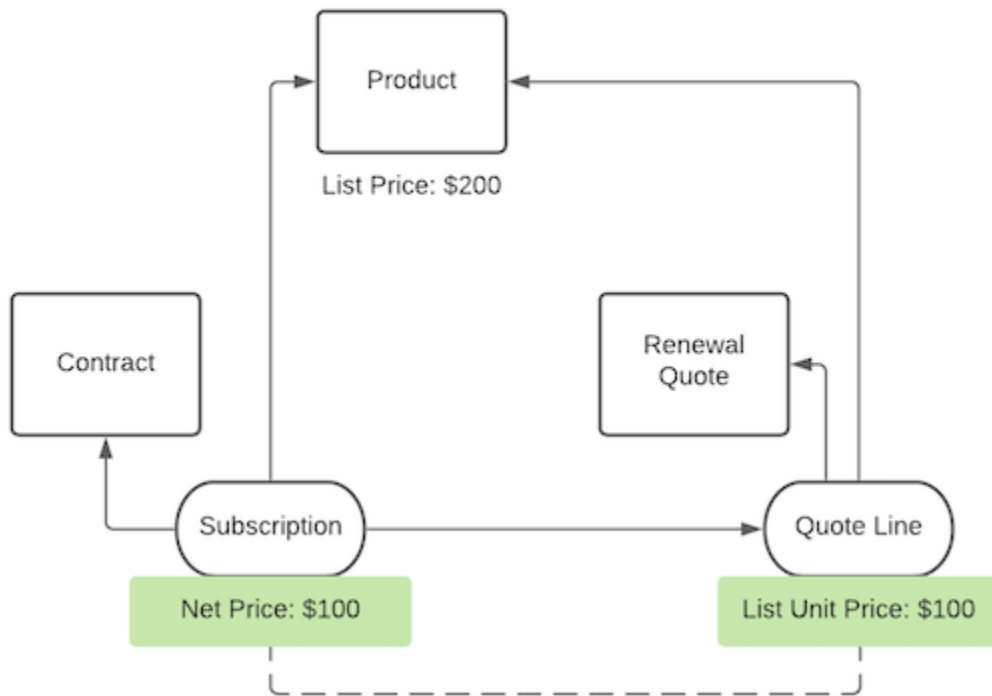


Same – This will take the same price, including discounts, from the original Opportunity and apply them to the Renewal Opportunity /Quote at the time of creating the Renewal. Meaning if you had a list price of 100 and then gave a 10% discount and it was sold for \$90, then on Renewal Quote, you will see the same values flow through. Note that this discount is only on the customer price and does not take into account Partner or Distributor discounting.



[Salesforce CPQ by Smriti Sharan](#)

Renewal Pricing
Method: Same

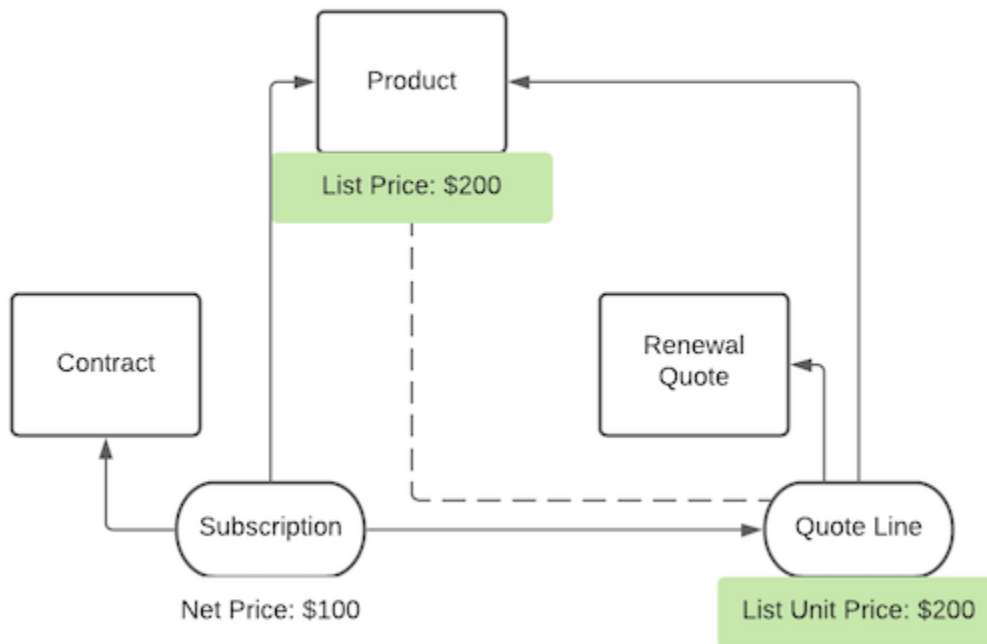


List – It takes the List price of the Product based on the Pricebook in used in the Original Opportunity unless you have noted a Renewal Pricebook ID on the Contract, in which case it will take that List Price. There will be no flow-through of Discounts if this option is selected.



[Salesforce CPQ by Smriti Sharan](#)

Renewal Pricing
Method: List

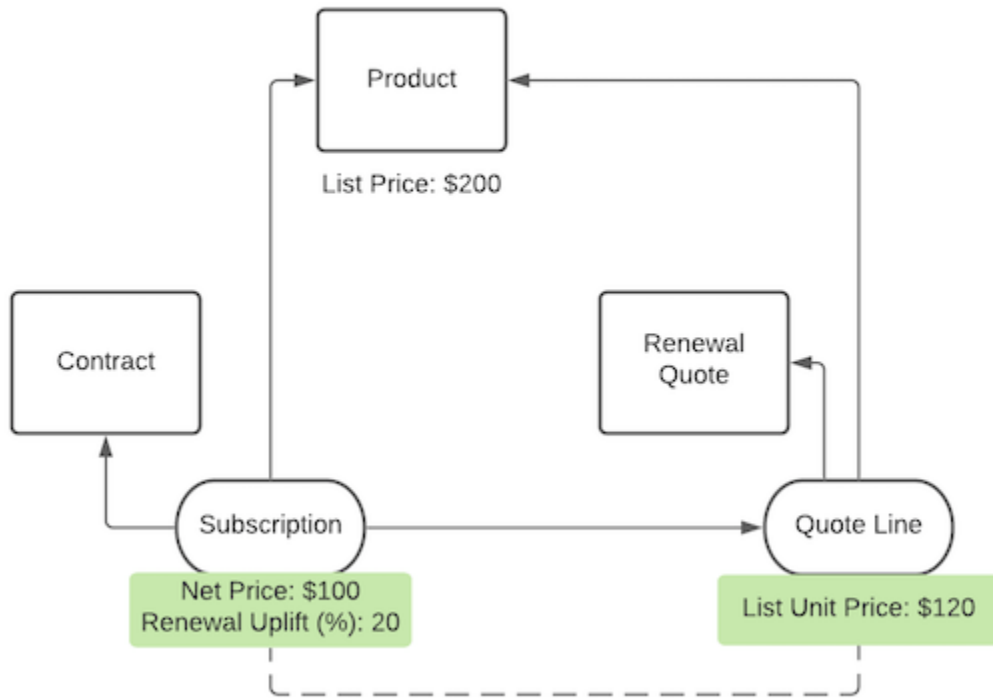


Uplift – This is the only option that requires additional setup if you choose to use it. Uplift is for when you want to take your Customer price and add some percentage of increase on the Renewal Quote. In addition to marking the field on the Account, you will need to set the Renewal Uplift Rate at either the Subscription or Contract level. If both are set, the Subscription takes precedence over the Contract.



[Salesforce CPQ by Smriti Sharan](#)

Renewal Pricing
Method: Uplift



What's the difference between 'Co-Terminated Quotes' and regular quotes in Salesforce CPQ?

'Co-Terminated Quotes' are special because they're used when we make changes to a contract that's already going on. Regular quotes are just the usual offers we make to customers.



[Salesforce CPQ by Smriti Sharan](#)

In Salesforce CPQ, how do you handle a scenario where a customer wants to downgrade or remove services during renewal?

If a customer wants less or different stuff at renewal, I use the 'Amend' function to remove or change what they don't want anymore.

What steps do you take in Salesforce CPQ when a customer wants to renew their contract earlier than the original end date?

When customers want to renew early

- I first create a new quote for the renewal.
- I then adjust the start date to match when the customer wants to renew and make sure any old charges stop when the new contract starts. This way, the customer doesn't get charged twice for the same thing.

Can you explain the process of transferring assets during an amendment in Salesforce CPQ?



[Salesforce CPQ by Smriti Sharan](#)

Transferring assets is like moving a toy from one box to another. In Salesforce CPQ, if a customer wants to move a service (the toy) from one contract (the box) to another, I use the 'Amend' feature to move it, making sure it's removed from the old contract and added to the new one.

What is the difference between contract-based renewals and asset-based renewals?

Contract-based renewals involve the entire contract, while asset-based renewals deal with specific assets or subscriptions within a contract.

What is contracted pricing in Salesforce CPQ?

Contracted pricing refers to special prices associated with a particular customer, which are considered during the quoting process.

What is the functionality of the "Ignore Parent Contracted Prices" checkbox?



[Salesforce CPQ by Smriti Sharan](#)

It allows users to bypass the default behavior of cascading contracted prices down to related accounts.

What is the role of the “Renew Contracts” button in Salesforce CPQ?

This feature allows users to select multiple contracts for an account and merge them into one master contract for renewal.

Can you change the pricing of existing products during the amendment process?

No, you can't change the pricing of existing products; you can only modify quantities or add new products.

Why is there a separate opportunity and quote created for amendments?

This helps in maintaining a detailed history and traceability between the original and amended contracts.



[Salesforce CPQ by Smriti Sharan](#)

What happens if you set the “Generate Contracted Price” action at the quote level?

It will apply the contracted price only for that quote.

How does Salesforce CPQ support early contract termination?

Users can set the quantity to zero for all SKUs from a specific date, effectively terminating the contract early.

Can you perform partial contract renewals in Salesforce CPQ, and if so, how?

Yes, this is possible through asset-based renewals, which allow for renewing specific assets or subscriptions within a contract, rather than the entire contract.

What is the role of the “Master Contract” in the renewal process?

A Master Contract can be created from multiple contracts for an account, consolidating them into a single contract for ease of renewal.



[Salesforce CPQ by Smriti Sharan](#)

How does contracted pricing affect discount schedules on products?

The discount schedule set on the contracted price will override any existing discount schedules on the product, making it the prevailing pricing for the duration of the contract.

What records does Salesforce CPQ create when you initiate an amendment?

Salesforce CPQ creates an amendment opportunity with a close date equal to your contract's start date and an amendment quote with start and end dates matching your contract's terms. The opportunity and quote will include specific naming conventions and details reflecting their nature as amendments.

How does Salesforce CPQ calculate product quantity and price on your amendment opportunity?

Product quantity and price on your amendment opportunity are calculated based on the difference between the original quote and the amendment



[Salesforce CPQ by Smriti Sharan](#)

quote. The amendment opportunity also updates its 'Amended Contract' and 'Primary Quote' fields with links to the contract you amended and your original quote.

If several contracts under an account are due for renewal, how does Salesforce CPQ consolidate them?

When multiple contracts are due for renewal, Salesforce CPQ allows you to select all the contracts you want to renew from the account's Contracts related list. After clicking "Renew Contracts," you're prompted to pick a master contract. The system then creates a single renewal opportunity and quote, consolidating all the subscriptions and covered assets from the selected contracts.

What happens if you just select 'Renewal Quoted' and not 'Renewal Forecast'?

You can also skip clicking **Renewal Forecast** and click **Renewal Quoted** to create your renewal opportunity and renewal quote at the same



[Salesforce CPQ by Smriti Sharan](#)

time. However, it's recommended to use Renewal Forecast so you can track pipeline opportunities for record-keeping purposes before finalizing the renewal quote.

There is a quote with a subscription product(s) for 12 months, and you want to increase the quantity and increase the term to 18 months for all products. How will you approach it?

There are two ways to handle this:

1. If you do not need to adjust pricing, you can **adjust the Start Date and End Date of the Quote** to be what you need it to be, as well as the Quantity.
2. If you do need to adjust pricing, you will need to zero out the existing line, which will give you credit for the cost, and then add the product back, adjusting the Quantity and Price. Then you will need to update the End Date to the net new End Date.



[Salesforce CPQ by Smriti Sharan](#)

The caveat to both options above has to do with the Contract's End Date vs. the End Date of the Subscription.

How to renew multiple contracts from accounts?

You have several contracts that have occurred on an account that have different subscription products with different end dates. You want to merge them together and renew them all into a single opportunity and quote.

To do the scenario you would use the Renew Contracts or Renew Service Contracts

Points to Note:

1. If you have two contracts and one has expired, you need to select the one with the later date.
2. The master contract determines the start and end date of the new renewal quote.
3. If you have two contracts and neither has expired the recommendation by [Salesforce](#) is to make the one with the earlier



[Salesforce CPQ by Smriti Sharan](#)

close date to ensure there are no gaps between the dates of the products on those contracts.

4. Whichever Contract is the master will set the length of the new Renewal Quote
5. This is designed for accounts that are doing Contract-based Renewals as opposed to Asset-based Renewals.

Steps to achieve this scenario:

1. Go to the account and select the Contracts you want to Renew

Accounts > Kevin Inc.
Service Contracts

2 items selected

New Renew Service Contracts

	Contract Name T	Contract Number	Start Date	End Date	Status	Created Date
1	<input type="checkbox"/> Service Contract	00000009	4/30/2020	4/29/2022	Active	5/3/2020, 9:26 AM
2	<input checked="" type="checkbox"/> Service Contract	00000010	5/1/2021	4/30/2022	Inactive	5/3/2020, 9:26 AM
3	<input type="checkbox"/> Service Contract	00000011	4/24/2020	4/23/2021	Active	5/3/2020, 9:26 AM
4	<input type="checkbox"/> Service Contract	00000004	2/16/2020	2/17/2020	Expired	2/18/2020, 9:35 AM
5	<input type="checkbox"/> Service Contract	00000005	2/3/2020	2/16/2020	Expired	2/18/2020, 9:50 AM
6	<input checked="" type="checkbox"/> Service Contract	00000007	1/1/2020	12/31/2022	Active	3/5/2020, 8:35 AM
7	<input type="checkbox"/> Service Contract	00000008	4/21/2020	4/20/2021	Active	4/21/2020, 2:01 PM

2. Select the contract that is ending earlier

Renew Service Contracts

Select Master Contract (optional) Renew Cancel

Master	Contract Number	Start Date	End Date
<input checked="" type="radio"/>	00000010	5/1/2021	4/30/2022
<input type="radio"/>	00000007	1/1/2020	12/31/2022

Renew Cancel

3. Click Renew



[Salesforce CPQ by Smriti Sharan](#)

4. You will then be taken back to the Account page
5. Go to the Opportunity-related list to find the new Renewal Opportunity refresh to see the Renewal Opportunity link
6. From the Renewal Opportunity go to the Quote and click Edit Lines to see the newly created Quote

The screenshot shows the 'Edit Quote' interface in Salesforce CPQ. At the top, there are buttons for 'Add Products', 'Add Group', 'Delete Lines', 'Quick Save', 'Calculate', 'Cancel', and 'Save'. Below this is the 'Quote Information' section, which includes a 'Start Date' field set to 5/1/2022 (highlighted with a green box), an 'End Date' field set to 4/30/2023, and a 'Subscription Term' field. Below the information is a table with the following data:

#	PRODUCT CODE	PRODUCT NAME	QUANTITY	LIST UNIT PRICE	ADDITIONAL DISC.	NET UNIT PRICE	NET TOTAL
1	OFFSITEVIDEOSTORAGE1GB	Offsite Video Storage 1GB	1.00	\$1.00		\$12.00	\$12.00
2	CLOUDSTORAGE	Cloud Storage 1GB	1.00	\$20.00		\$6.67	\$6.67
SUBTOTAL:							\$18.67

At the bottom right of the table, it says 'QUOTE TOTAL: \$18.67'.

7. Here you can see that Contract had ended on 4/30/2022 and the new quote starts 5/1/2022 – one day after the end of the contract, so no time is lost

How to cancel a contract?

To properly cancel a contract, you need to do an amendment action in which you zero out all of the Quote Lines on the Contract.



[Salesforce CPQ by Smriti Sharan](#)

Q-00103 | Edit Quote
Total: -\$39.91

Add Products Add Group Delete Lines Quick Save Calculate Cancel Save

Quote Information

Start Date: 4/8/2022 End Date: 8/31/2022 Subscription Term: []

Additional Disc. (%): []

Q-#	PRODUCT CODE	PRODUCT NAME	SUBSCRIPTION T...	QUANTITY	LIST UNIT PRICE	ADDL...	NET UNIT PRICE	NET TOTAL	
1	CREATIVITYSUITE	Creativity Suite License	Renewable	0.00	\$100.00		\$39.91	-\$39.91	[] []

Subtotal: -\$39.91

Once all the lines are churned (zeroed out), you can contract the Quote through the Opportunity or the Order, depending on if you need an Order for this process.

Your Contract will end up with new negative lines for all items churned; however, there will be no indicator on the Contract that it has been terminated.



[Salesforce CPQ by Smriti Sharan](#)

Subscriptions (2) New			
Subscription #	Product	Quantity	List Price
SUB-0000051	Creativity Suite License	1.00	\$100.00
SUB-0000052	Creativity Suite License	-1.00	\$39.91
View All			

Negative Subscription Line

The Subscription Lines will have a Terminated Date of the date you cancelled everything.

Related	Details
Subscription # SUB-0000052	Account ⓘ Grand Hotels & Resorts Ltd
Quantity ⓘ -1.00	Contract ⓘ 00000116
Product ⓘ Creativity Suite License	Contract # ⓘ 00000116
Product Name ⓘ Creativity Suite License	Start Date ⓘ 4/8/2022
Quote Line ⓘ QL-0000519	End Date ⓘ 8/31/2022
	Subscription Start Date ⓘ 4/8/2022
	Subscription End Date ⓘ 8/31/2022
	Terminated Date ⓘ 4/7/2022

Terminated Date on the Subscription



[Salesforce CPQ by Smriti Sharan](#)

If you want your contracts to reflect this cancellation/churn, you can add a value to the Status picklist and then have automation based on the Terminated Dates to set it.

What are the steps to create renewal opportunities?

1. Close-Won your opportunity
2. Ensure that there is a CPQ quote related to the Opportunity and that it is the Primary Quote for the Opportunity
3. Click the Contracted checkbox to kickoff the renewal process
 1. NOTE: A Contract will only be generated if you have a Primary Quote and there are Subscription based Products on that Quote

A screenshot of the Salesforce CPQ 'Opportunity Detail' page. The page shows various fields for the opportunity, including 'Opportunity Owner', 'Private', 'Opportunity Name', 'Account Name', 'Type', and 'Lead Source'. Below these fields is a section titled 'Salesforce CPQ Information' which contains several checkboxes: 'Primary Quote' (checked), 'Contracted' (checked), 'Ordered' (unchecked), and 'Create Contracted Prices' (unchecked). The 'Contracted' checkbox is highlighted with a red box. At the top of the page, there are navigation links for 'Quotes (1)', 'Products (Standard Price Book) (3)', 'Quote Documents (0)', 'Orders (0)', 'Contracts (1)', and 'Renewal Contracts (0)'. The 'Contracts (1)' link is also highlighted with a red box. Action buttons for 'Edit', 'Delete', 'Clone', 'Include Document', and 'Create' are visible at the top right of the detail section.



[Salesforce CPQ by Smriti Sharan](#)

- On the Contract, you will note that Subscriptions were automatically generated based on the Products and the setup for the Products

Subscriptions							New Subscription
Action	Subscription #	Product	Quantity	List Price	Start Date	End Date	
Edit Del	SUB-0000009	Copy Paper Letter	1.00	\$85.00	12/10/2018	12/9/2019	

- Click the Activate button to “turn on” your Contract

Contract 00000104

Click to add topics: ⓘ

Subscriptions (1) | Co-Terminated Quotes (0) | Renewal Opportunities (0) | Amendment Opportunities (0) | Contracted Prices (0) | Orders (0) | Contract History (1) | Open Activities (0) | Activity History (0)

Contract Detail

[Edit](#) [Delete](#) [Clone](#) [Activate](#) [Amend](#)

Contract Owner	[Change]	Status	Draft
Contract Number	00000104	Contract Start Date	12/10/2018
Account Name	Klein Manor	Contract End Date	12/9/2019
Customer Signed By		Contract Term (months)	12
Customer Signed Title		Owner Expiration Notice	
Customer Signed Date		Company Signed By	
Price Book		Company Signed Date	

▼ Salesforce CPQ General Information

Opportunity	Sample Renewal Process Cpq	Opportunity Pricebook Id	01s5A000005EUjh
Quote	Q-00031	Combine Subscription Quantities	<input type="checkbox"/>
Order		Amendment & Renewal Behavior	Latest End Date

▼ Salesforce CPQ Renewal Information

Renewal Forecast	<input type="checkbox"/>	Renewal Pricebook Id	
Renewal Opportunity Stage		Renewal Opportunity Record Type Id	
Renewal Opportunity		Renewal Term	12
Renewal Owner		Renewal Uplift (%)	
Default Renewal Contact Roles	<input checked="" type="checkbox"/>	MDQ Renewal Behavior	
Default Renewal Partners	<input checked="" type="checkbox"/>	Preserve Bundle Structure	<input checked="" type="checkbox"/>
Renewal Quoted	<input type="checkbox"/>	Master Contract	<input type="checkbox"/>



[Salesforce CPQ by Smriti Sharan](#)

6. Click the Renewal Forecast checkbox – this will create the Renewal Opportunity automatically

A screenshot of the 'Salesforce CPQ Renewal Information' settings. The 'Renewal Forecast' checkbox is checked and highlighted with a red box. Other settings include 'Renewal Opportunity Stage', 'Renewal Opportunity', 'Renewal Owner', 'Default Renewal Contact Roles' (checked), 'Default Renewal Partners' (checked), and 'Renewal Quoted' (unchecked).

Renewal Forecast	<input checked="" type="checkbox"/>
Renewal Opportunity Stage	
Renewal Opportunity	
Renewal Owner	
Default Renewal Contact Roles	<input checked="" type="checkbox"/>
Default Renewal Partners	<input checked="" type="checkbox"/>
Renewal Quoted	<input type="checkbox"/>

1. By default, the name of the Renewal Opportunity will be just that “Renewal Opportunity

A screenshot of the 'Salesforce CPQ Renewal Information' settings. The 'Renewal Forecast' checkbox is checked. The 'Renewal Opportunity' field is highlighted with a red box and contains the text 'Renewal Opportunity'. Other settings include 'Renewal Opportunity Stage', 'Renewal Owner', 'Default Renewal Contact Roles' (checked), 'Default Renewal Partners' (checked), and 'Renewal Quoted' (unchecked).

Renewal Forecast	<input checked="" type="checkbox"/>
Renewal Opportunity Stage	
Renewal Opportunity	Renewal Opportunity
Renewal Owner	
Default Renewal Contact Roles	<input checked="" type="checkbox"/>
Default Renewal Partners	<input checked="" type="checkbox"/>
Renewal Quoted	<input type="checkbox"/>

7. You now have a Renewal Opportunity that can be amended and later can be Quoted.



[Salesforce CPQ by Smriti Sharan](#)

When we are creating an Amendment/Renewal Opportunity from an existing Opportunity, we want a custom field value to be populated from the respective contract custom field?

Creating Twin Fields – Create custom field in contract object and opportunity with same API name and field type

How can you renew the subscription with an existing client?

If you already have a renewal opportunity, when you get closer to the end of the subscription term, you might want to sell them the renewal subscriptions to client again. To do this, you need a renewal quote.

It can be generated by marking the Renewal Quoted checkbox as true.

This renewal quote can be sent out, it can be updated and changed as needed and then received back in so that you can subsequently create a brand-new contract.

1. First, click the **Accounts** tab from the navigation bar.



[Salesforce CPQ by Smriti Sharan](#)

2. Go to the necessary company and click the **Related** tab.
3. Access the Contract Number link in the Contracts related list and then click **Edit**.
4. Mark the **Renewal Quoted** checkbox as true, which will tell CPQ to generate a quote on the renewal opportunity.

Note that the renewal quote has the Start Date one day after the Close Date of the renewal opportunity, while the End Date is based on the Renewal Term of the contract.

How to amend a contract in salesforce CPQ?

Let's see how Salesforce CPQ can update an existing Contract to include higher quantities of subscription products, prorated for the remaining duration of the Contract and automatically create an amendment Opportunity and Quote:



[Salesforce CPQ by Smriti Sharan](#)

Create an amendment Quote and an Opportunity to increase the quantity of a subscription product from 1 to 10, starting 3 months into the contract.

Navigate to that **Account** and go to the **Contract Number link** in the Contracts related list. Then, click **Amend**, and again **Amend**.

How Option Constraining Group work in Salesforce CPQ works?

This feature comes in handy when you want to make mandatory or exclude options based on combinations of other options.

Let's look at an example where you want to prevent the sales reps from selling Product A unless Product B have been included in the quote:

- A. Click the **Products** tab on the navigation bar.
- C. Go to the **Related** tab.
- D. In the **Options** related list, select **View All** and note the number of the Option Name for the necessary options:



[Salesforce CPQ by Smriti Sharan](#)

- Product A.
- Product B.

E. Click **New** in the **Option Constraints** related list and fill in the following fields:

- For the **Constraint Name** field, enter: Product A requires Product B
- For the **Constrained Option** field, enter: < PO-##### of Product A >
- For the **Constraining Option** field, enter: < PO-##### of Product B >
- For the **Option Constraint Group** field, enter: You could enter here any text string, but it's best practice to make it descriptive and easy to understand and remember

F. Click **Save**.

How can you renew the subscription with an existing client?



[Salesforce CPQ by Smriti Sharan](#)

If you already have a renewal opportunity, when you get closer to the end of the subscription term, you might want to sell them the renewal subscriptions to client again. To do this, you need a renewal quote.

It can be generated by marking the Renewal Quoted checkbox as true.

This renewal quote can be sent out, it can be updated and changed as needed and then received back in so that you can subsequently create a brand-new contract.

1. First, click the **Accounts** tab from the navigation bar.
2. Go to the necessary company and click the **Related** tab.
3. Access the Contract Number link in the Contracts related list and then click **Edit**.
4. Mark the **Renewal Quoted** checkbox as true, which will tell CPQ to generate a quote on the renewal opportunity.



[Salesforce CPQ by Smriti Sharan](#)

Note that the renewal quote has the Start Date one day after the Close Date of the renewal opportunity, while the End Date is based on the Renewal Term of the contract.

How to amend a contract in salesforce CPQ?

Let's see how Salesforce CPQ can update an existing Contract to include higher quantities of subscription products, prorated for the remaining duration of the Contract and automatically create an amendment Opportunity and Quote:

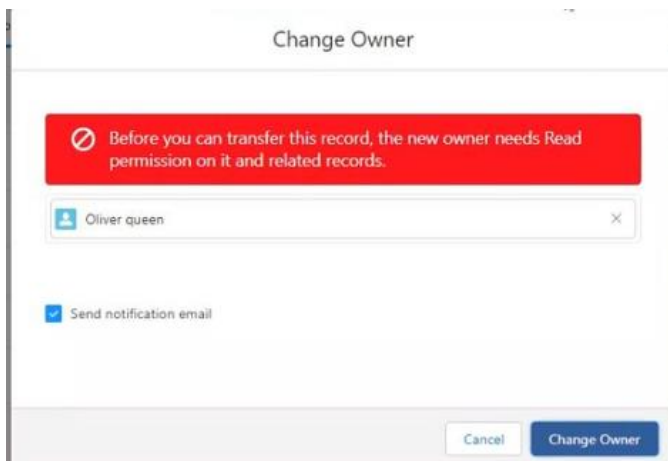
Create an amendment Quote and an Opportunity to increase the quantity of a subscription product from 1 to 10, starting 3 months into the contract.

Navigate to that **Account** and go to the **Contract Number** link in the Contracts related list. Then, click **Amend**, and again **Amend**.



[Salesforce CPQ by Smriti Sharan](#)

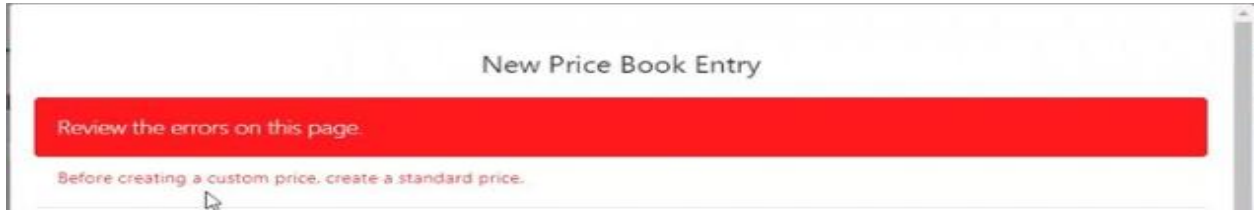
General Errors



Error occurs when attempting to transfer a record due to lack of read permission for the new owner.



[Salesforce CPQ by Smriti Sharan](#)



An error occurs when trying to add a product to custom price book without creating a standard price first.