



TRAILHEAD

Context Rules

Exercise Guide

Version 1.0

TABLE OF CONTENTS

Preface	5
Overview	5
What You Will Learn	5
Prerequisites	6
Product Overview	6
Industries CPQ	6
Rules	6
Context Rules Framework	6
What are Rule Sets?	7
What are Context Rules?	7
What are Context Dimensions?	9
What are Context Scopes?	9
What are Context Mappings?	9
Exercise 6-1: Creating a Qualification Context Rule	10
What is a Vlocity Picklist?	11
Task 1: Create a Picklist for Account Status	12
What are context scopes?	15
Task 2: Review context scopes	18
What is a context dimension?	20
Task 3: Create a context dimension	21
What is a context mapping?	23
Task 4: Create a context mapping	23
Task 5: Create a context rule	25
Task 6: Create rule conditions	27
Task 7: Create a rule set	29
Review	33
Exercise 6-2: Deploying a Qualification Context Rule	34
Task 1: Add a rule set to a promotion	35
Task 2: Test the rule set in the Cart	37
Task 3: Change the account status to New	39

Task 4: Retest the rule set in the Cart	40
Troubleshooting Guide	41
Review	42
Exercise 6-3: Creating Context Rules for Price Lists	43
How are context rules used for pricing?	44
Task 1: Create a context dimension that will allow users to type in values for rule conditions	45
Task 2: Create a context mapping to account record type	47
Task 3: Create a context rule for consumer's account	49
Task 4: Create a context rule for non-consumer accounts	52
Task 5: Create a rule set for consumer accounts	55
Task 6: Create a rule set for business accounts	58
Task 7: Assign the new rules sets to price lists	61
Task 8: Test the new rules in the Cart	64
Troubleshooting Guide	65
Review	66
Exercise 6-4: Creating Context Rules for Price List Entries	67
How are context rules used for price list entries?	68
What is an interface?	68
What is the TightestMatchInterface?	69
What are condition weights?	69
Task 1: Review the interface implementation	70
Task 2: Assign the new accounts rule set to a price list entry	71
Task 3: Create a rule to apply a child price list	77
Task 4: Change the condition weight to make the new account pricing the tightest match	90
Troubleshooting Guide	92
Review	94
Exercise 6-5: Using Context Rules for Regional Product Launches Challenge	95
Task 1: Determine where the regional data is stored	96
Task 2: Determine which context scope you will use	96
Task 3: Create a "Region" context dimensions and context mapping to the regional data	96

Task 4: Create a “High Value Regions” context rule using the Region context dimension	96
Task 5: Create a “High Value Consumer Regions” rule set	96
Task 6: Assign the High Value Consumer Regions rule set to the 5G Data Plan product	96
Task 7: Test the new rule in the Cart	97
Task 8: Tune the rule set for best performance	97

Preface

These training exercises are based on the Fall' 20 release of Salesforce Industries Communications, Media, and Energy & Utilities Clouds. For additional information about the topics covered, see the documentation available in Vlocity Success Community at <https://success.vlocity.com>.

Overview

This Context Rules training module provides a general overview of context rules. Following the conceptual overview, it provides exercises in which you create context rules.

This is a hands-on course with practical lab exercises. The lab exercises are designed to be used with the provided training playground.

What You Will Learn

When you complete this training, you will be able to:

- Explain context mappings, context scopes and context dimensions.
- Create a context dimension.
- Create a qualification rule.
- Specify rule conditions.
- Create a context mapping using a context scope.
- Add a rule to a rule set.
- Apply a qualification rule to a promotion.
- Test a qualification rule.
- Use the TightestMatchInterface.
- Create context rules for price lists and price list entries.
- Control which price lists display in the Cart.
- Create context rules for child price lists.
- Use context dimension weights to determine which price is applied to a product.

Prerequisites

The prerequisites for this training include a solid understanding of basic Salesforce concepts and functionality. You should also have a working knowledge of your industry business objectives.

Product Overview

Industries CPQ

Industries CPQ has a powerful rules engine that ensures that quotes and orders are valid before they're submitted, minimizing order delays, customer confusion and dissatisfaction. Industries CPQ picks up where your CRM ends, providing real-time product, pricing, and business rules using natural language and declarative administration. Using Industries CPQ, you can:

- Design basic and advanced availability, eligibility, compatibility, and pricing rules
- Standardize new product and service introductions
- Ensure accurate capture of new, move, add, change and disconnect orders

Industries CPQ supports both B2B and B2C lead-to-cash processes. Our open architecture allows you to use Industries CPQ components, services, and user interface selectively as individual components and to integrate process flows with external systems, all using a unified user interface.

Rules

Rules allow you to change standard product and pricing behaviors in the Cart in accordance with your business objectives. There are two rules frameworks that work in tandem: Advanced Rules and Context Rules.

Context Rules Framework

The Context Rules Framework is used to determine which products and promotions appear on the Qualified and Disqualified tabs in the Cart, whether a penalty applies to an account and what the penalty will be. There are two primary types of Context Rules:

- **Qualification** rules determine which products and promotions appear on the Qualified and Disqualified tabs in the Cart.
- **Penalty** rules determine whether a penalty applies and what the penalty will be. Penalties can be applied in relation to conditions set on promotions, contracts or accounts. For example, early termination of an agreement might involve penalty fees for a consumer. When creating a penalty rule for an account or contract, you create it as an object-level rule.

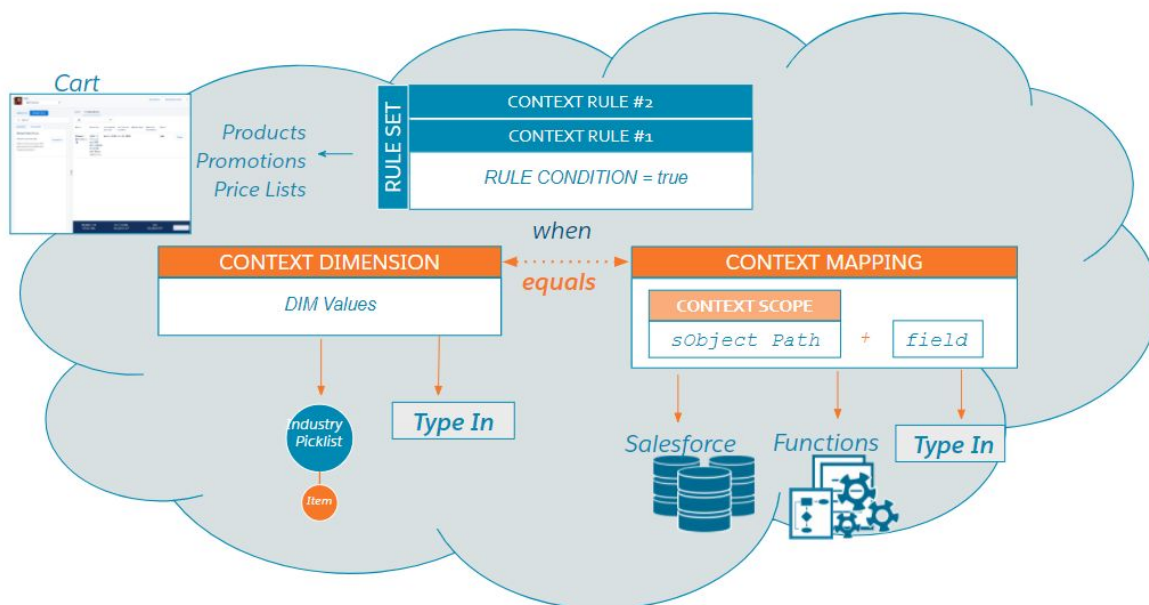
The Context Rules Framework consists of rule sets, context rules, context dimensions, context scopes and context mappings.

What are Rule Sets?

Rule sets contain one or more context rules and are applied to products, promotions or price lists.

What are Context Rules?

Context rules are analogous to entity filters in the Advanced Rules Framework. They contain the information needed to determine when a rule set should fire in the Cart. Context rules contain rule conditions, which are made up of context dimensions, context scopes and context mappings.



What are Context Dimensions?

A context dimension is a variable that describes the possible values that can be used in a rule condition. It can be reused across multiple rule conditions. Context dimensions are compared by the rules engine against data that comes from an sObject, function, or static value defined in the context mapping.

What are Context Scopes?

Context scopes describe the relational path from a root sObject, such as an Order, to related sObjects, and they are used by context mappings to pinpoint the fields on the sObjects or computed data that will be matched against context dimension variables for evaluation by the Context Rule Service engine.

What are Context Mappings?

Context mappings provide the “magic link” that allows the rules engine to compare context dimension variables to data stored in sObjects, calculated via a function, or typed in during design-time within a specified context scope. Context mappings are cached in the org cache.



NOTE:

If you were to write a context rule condition as an expression, it would be:

Rule Condition = `true` WHEN `{{Context Dimension}} == {{Context Mapping[Context Scope.Source Expression]}}`

Exercise 6-1: Creating a Qualification Context Rule

Scenario

Ada, Infiwave's Pricing Manager, created a promotion to introduce the iPhone X, but wants to limit the promotion to new customers only. Eliza, Infiwave's Developer, knows the best way to do this is to use a context rule to determine whether a customer is a new customer before offering the promotion to them. This change should help Infiwave to meet the goal of increasing the company's new customer base.

Goals

- Identify the different types of context rules
- Understand context dimensions, context scopes and context mappings
- Create a qualification rule
- Specify rule conditions
- Add a rule to a rule set

Tasks

1. Create a picklist for Account Status
2. Review context scopes
3. Create a context dimension
4. Create a context mapping
5. Create a context rule
6. Create rule conditions
7. Create a rule set

Time: 15 mins




ALERT:

If you've just received your training playground, add your email address to the system administrator profile to ensure you receive all system notifications. In the upper-right, click on the **Avatar** and select **Settings**. Enter your email address in the **Email** field on the Personal Information page and click **Save**.

What is a Vlocity Picklist?

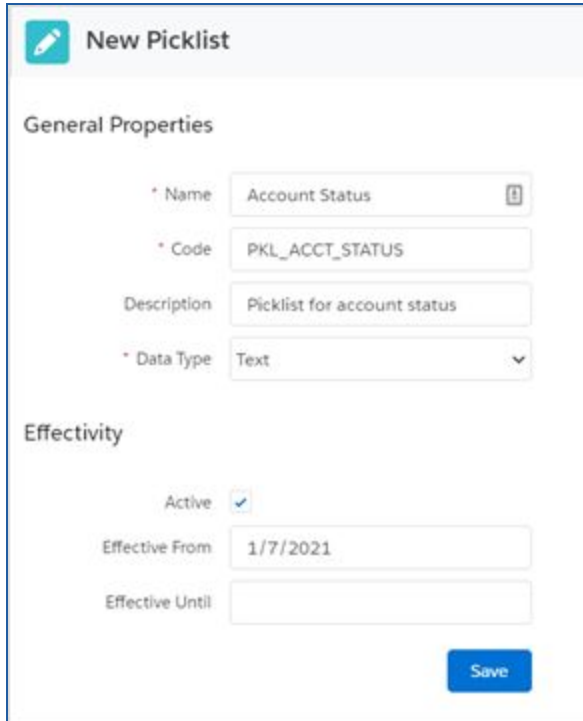
Vlocity picklists perform a similar function to Salesforce picklists, but they are custom objects called Vlocity picklists, with the API name: `vlocity_cmt__Picklist__c`. Vlocity picklists can be created with data types of text, number, Boolean (True/False), date or date time. Vlocity picklists are designed to be globally reusable across EPC, including within a context dimension. This picklist will be used to provide values for the context dimension that you will create later in this module.

Task 1: Create a Picklist for Account Status

1. Using the Lightning App Launcher , click **Configure Price Quote (CPQ)**.
2. In the Lightning navigation bar, click **Vlocity Product Console**.
3. Under **Foundation**, next to **Picklist**, create a new picklist by clicking the plus icon (+).
4. In the **New Picklist** dialog, enter the following information.

Field	Value	Notes
Name	Account Status	
Code	PKL_ACCT_STATUS	Using a PKL_ prefix to easily identify picklists is recommended. This value is not required to be unique.
Description	Picklist for account status	
Data Type	Text	
Active	✓	Setting the Active flag is recommended, although it is not currently enforced.
Effective From	[today's date]	Setting the Effective From (and Effective Until, if desired) fields is recommended, although it is not currently enforced.

5. Click **Save**.



New Picklist

General Properties

* Name: Account Status

* Code: PKL_ACCT_STATUS

Description: Picklist for account status

* Data Type: Text

Effectivity

Active: ☒

Effective From: 1/7/2021

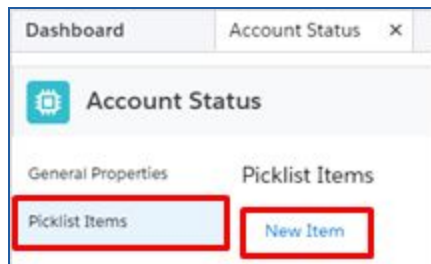
Effective Until:

Save

The next step is to create the picklist items. These picklist items will appear in the context dimension, which then will be used by the context rules service engine to match against specified sObject fields. Consequently, the items in the picklist that you specify here must match exactly the picklist items for the corresponding sObject field. In this exercise, the picklist items you create will be matched against the picklist items for the Status field of the Account sObject.

It is good practice to review the field in the Salesforce interface to confirm the spelling and capitalization of the picklist items. In this exercise, you will create a picklist item for “New.” To save time and typing, you won’t create picklist items for the rest of the Status field values (as you would in a real deployment).

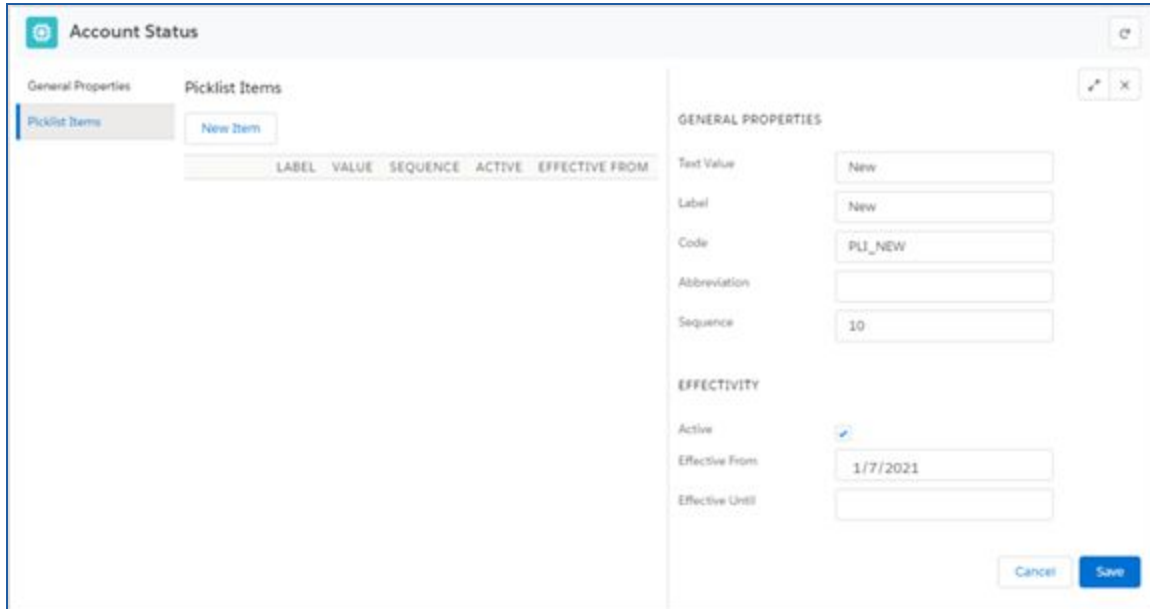
6. Click the **Picklist Items** facet.
7. Click **New Item**.



8. Enter the following information.

Field	Value	Notes
Text Value	New	This value will be evaluated programmatically by the Context Rules engine. The data type of this field is set by the picklist data type; because we specified a text data type on the picklist, we have a text entry field on the picklist item.
Label	New	If this picklist is ever re-used for an attribute, the label will be visible to the user at run-time in the Cart.
Code	PLI_NEW	This value is not required to be unique.
Sequence	10	Using sequences of 10, 20, 30 to simplify later insertions is recommended.
Active	✓	Setting the Active flag is recommended, although it is not currently enforced.
Effective From	[today's date]	Setting the Effective From (and Effective Until, if desired) fields is recommended, although it is not currently enforced.

9. Click **Save**.



The screenshot shows the 'Account Status' configuration page in Salesforce. On the left, there are tabs for 'General Properties' and 'Picklist Items'. The 'Picklist Items' tab is active, showing a table with columns: LABEL, VALUE, SEQUENCE, ACTIVE, and EFFECTIVE FROM. A 'New Item' button is located above the table. On the right, the 'GENERAL PROPERTIES' section contains input fields for Text Value (New), Label (New), Code (PLI_NEW), Abbreviation, and Sequence (10). Below this, the 'EFFECTIVITY' section has a checkbox for Active (checked), and date pickers for Effective From (1/7/2021) and Effective Until. At the bottom right are 'Cancel' and 'Save' buttons.

10. In the **Product Console** tab navigation bar, click **X** to close any open tabs, which should return you to the Dashboard.

Now that the picklist has been created, you can use this picklist in a context dimension. Before you can create the context dimension and subsequently the context mapping, you must have context scopes set up.

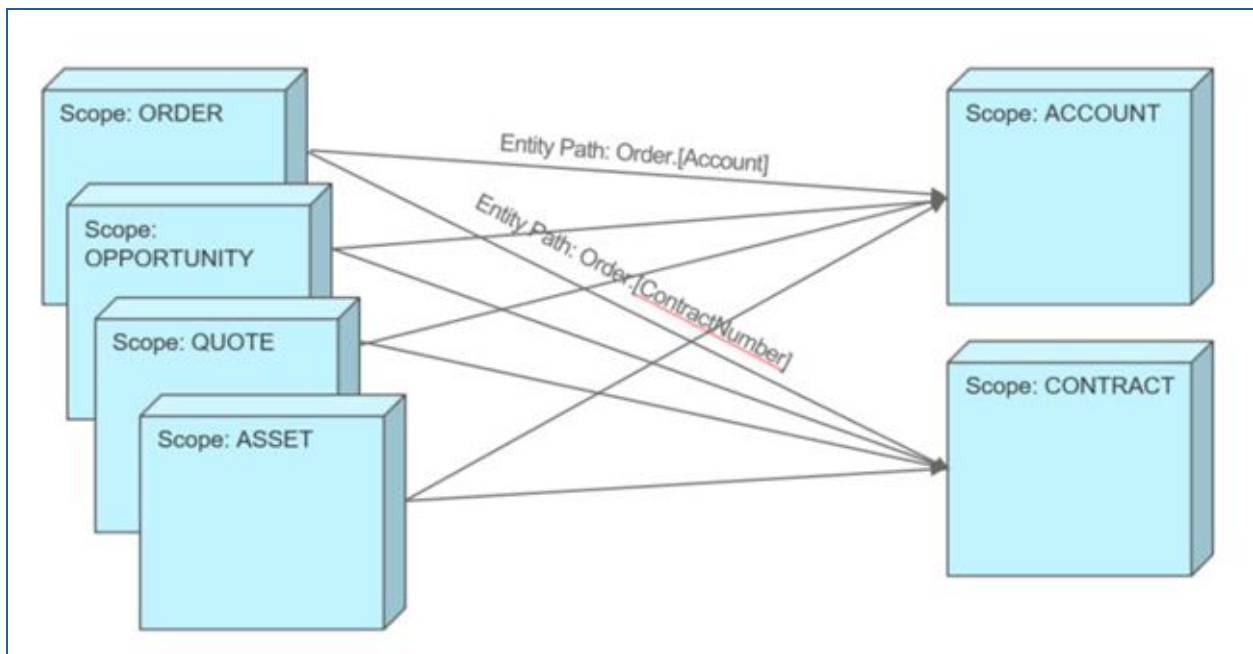
What are context scopes?

Context scopes describe the relational path from a root sObject, such as an Order, to related sObjects. They are used by context mappings to pinpoint the fields on the sObjects that will be matched against context dimension variables in the org cache for evaluation by the Context Rule Service engine.

Currently, Salesforce Industries supports the following root context scopes.

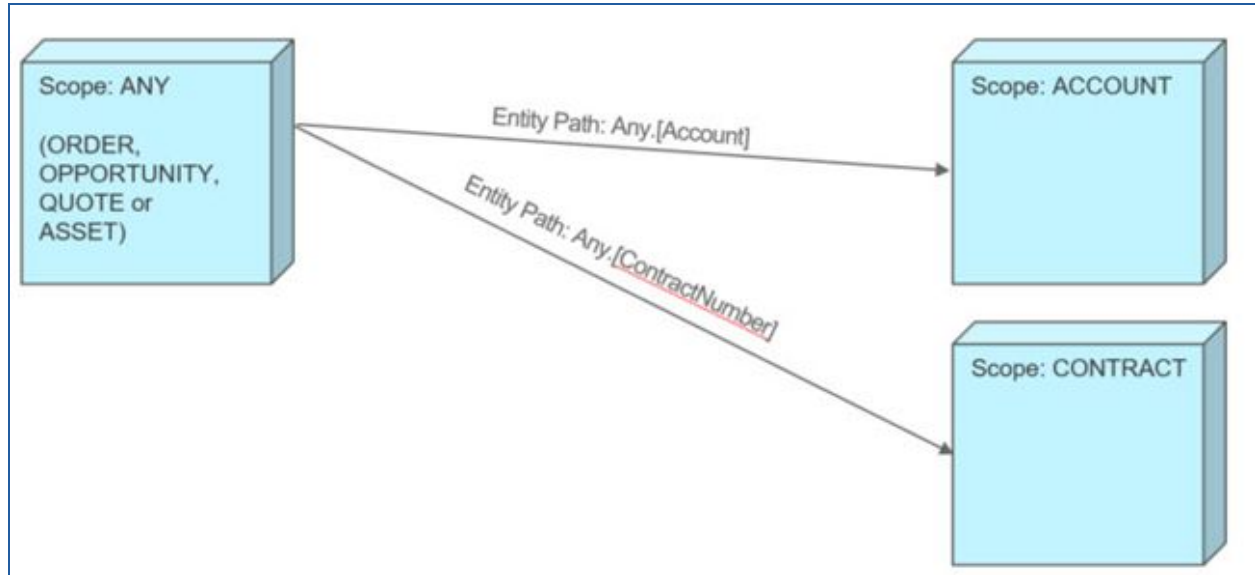
- Order
- Opportunity
- Quote
- Asset
- Any

Each of these scopes can then reference Account and Contract scopes.



The **Any** context scope allows you to create Account or Contract scopes that are agnostic with regard to the root context scope. For example, instead of creating separate Order.Account and Quote.Account scopes, you can create a single **Any.Account** scope that will find the associated account for all root entities including orders and quotes.

While the Any context scope allows more flexibility and efficiency when defining scopes, in certain cases when you know that the value can come from only one sObject, you may want to explicitly specify that object as the root scope entity. By doing so, you limit the number of fields that are retrieved and stored in the org cache. In other words, when the context scope is Order.Account, the engine retrieves fields from the account sObject only. If the context scope is Any.Account, the engine retrieves fields from all root entities: Order, Opportunity, Quote, and Asset. For example, the Originating Channel field only exists on the Order object. If you specify Any as the root entity, the system will still look into all root entities.





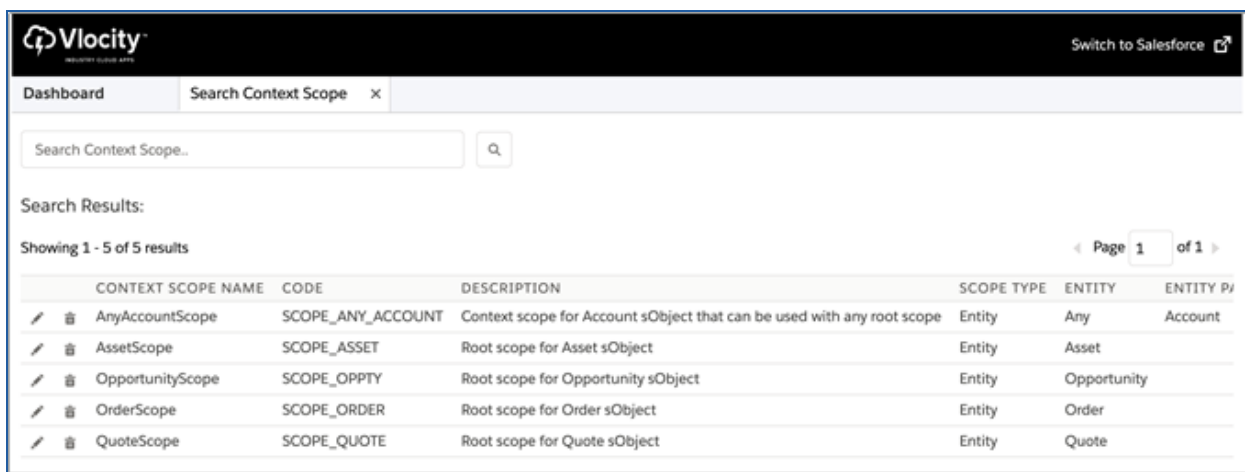
NOTE:

We have pre-built all the currently supported context scopes for you in your training playground. In future releases, scopes will be installed as part of the managed package.

In the next task, you will review two context scopes, including the one that you will use in creating your context rule, the Any.Account context scope.


Task 2: Review context scopes

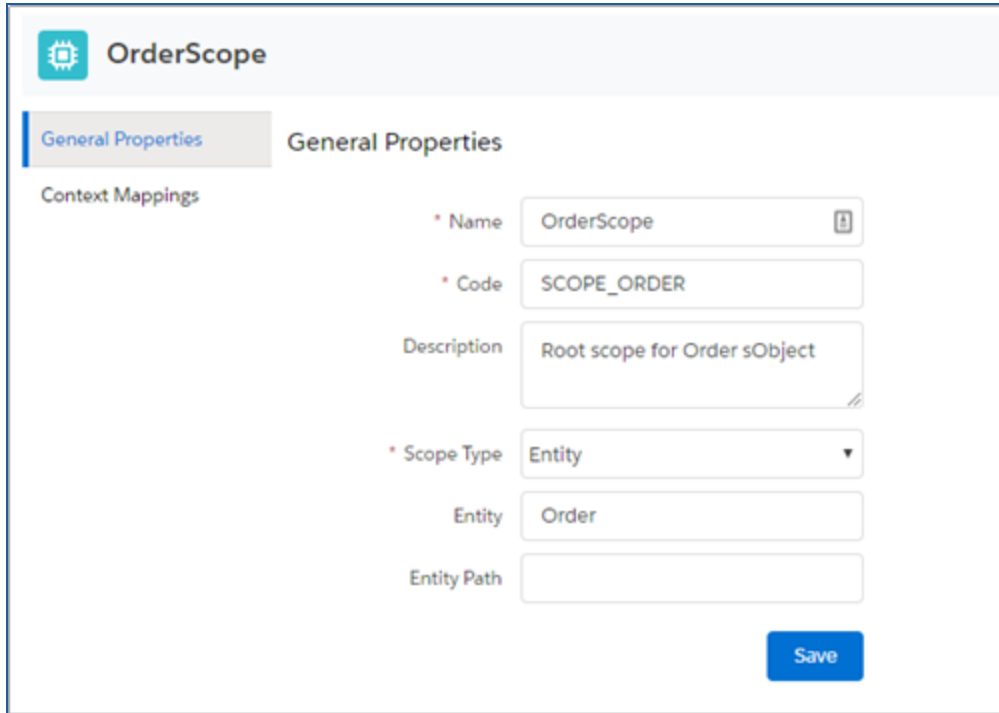
1. In the Dashboard under **Rules**, click the **search icon**  next to **Context Scope**.
2. In the **Search Context Scope...** dialog, press **Enter** or click the search icon. 
3. Review the context scopes in your training playground.



The screenshot shows the Vlocity training playground interface. At the top, there's a header with the Vlocity logo and a 'Switch to Salesforce' button. Below the header, there's a tab labeled 'Search Context Scope'. A search bar with the placeholder text 'Search Context Scope..' and a search icon is present. Below the search bar, it says 'Search Results:' and 'Showing 1 - 5 of 5 results'. A table displays the search results with columns: CONTEXT SCOPE NAME, CODE, DESCRIPTION, SCOPE TYPE, ENTITY, and ENTITY PATH. The table lists five context scopes: AnyAccountScope, AssetScope, OpportunityScope, OrderScope, and QuoteScope. The OrderScope is highlighted.

CONTEXT SCOPE NAME	CODE	DESCRIPTION	SCOPE TYPE	ENTITY	ENTITY PATH
AnyAccountScope	SCOPE_ANY_ACCOUNT	Context scope for Account sObject that can be used with any root scope	Entity	Any	Account
AssetScope	SCOPE_ASSET	Root scope for Asset sObject	Entity	Asset	
OpportunityScope	SCOPE_OPPTY	Root scope for Opportunity sObject	Entity	Opportunity	
OrderScope	SCOPE_ORDER	Root scope for Order sObject	Entity	Order	
QuoteScope	SCOPE_QUOTE	Root scope for Quote sObject	Entity	Quote	

4. Find the **OrderScope** root context scope, and click the **edit icon**. 
5. Notice that the **Entity** is **Order**, and because it is a root scope, there is no Entity Path specified.



OrderScope

General Properties

Context Mappings

* Name OrderScope

* Code SCOPE_ORDER


Description Root scope for Order sObject

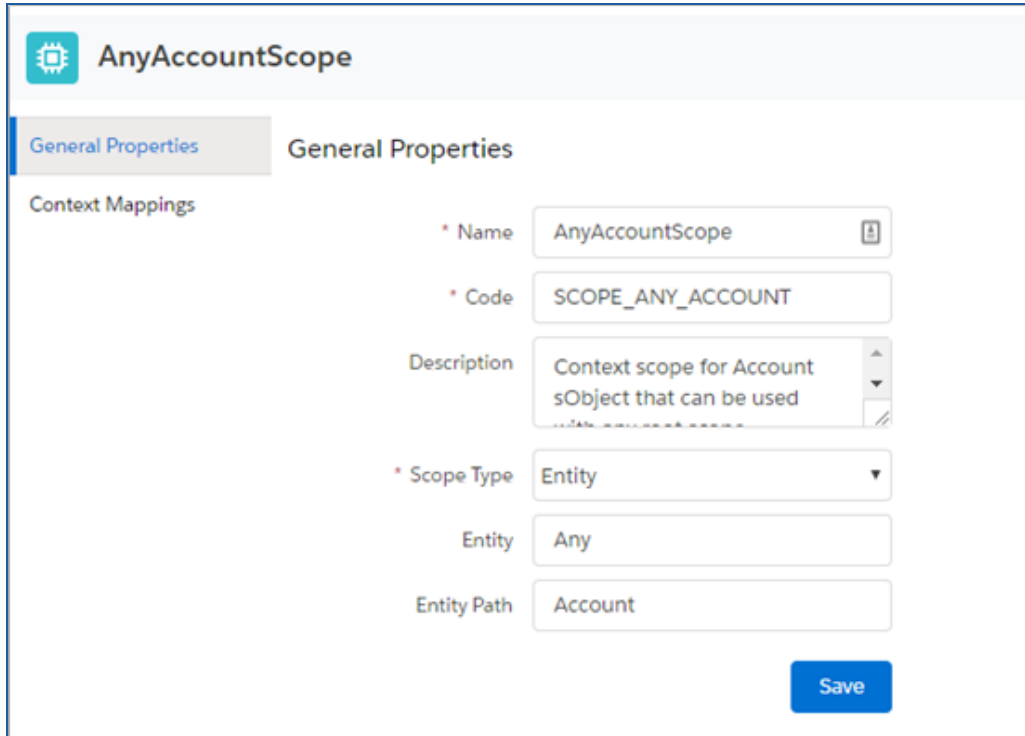
* Scope Type Entity

Entity Order

Entity Path

Save

6. In the **Product Console** tab navigation bar, click **X** to close the **OrderScope** tab.
7. Find the **AnyAccountScope** context scope and click the edit icon. 
8. Notice that the **Entity** is **Any**, and the **Entity Path** is **Account**. This will allow this account context scope to be used for orders, quotes, opportunities and assets.



9. In the **Product Console** tab navigation bar, click **X** to close the **AnyAccountScope** tab and the **Search Context Scope** tab.

Now that you have checked the context scopes are defined; the next step is to create the context dimension variable.

What is a context dimension?

A context dimension is a variable that stores possible values to use in a context rule condition. You can reuse context dimensions across multiple rule conditions.

Context dimensions are compared by the rules engine against data that comes from an sObject, a function, or a static value defined in the context mapping.

In this task, you will create a context dimension to store picklist item values for account status, which will then be matched against the account sObject Status field through a context mapping.

Task 3: Create a context dimension

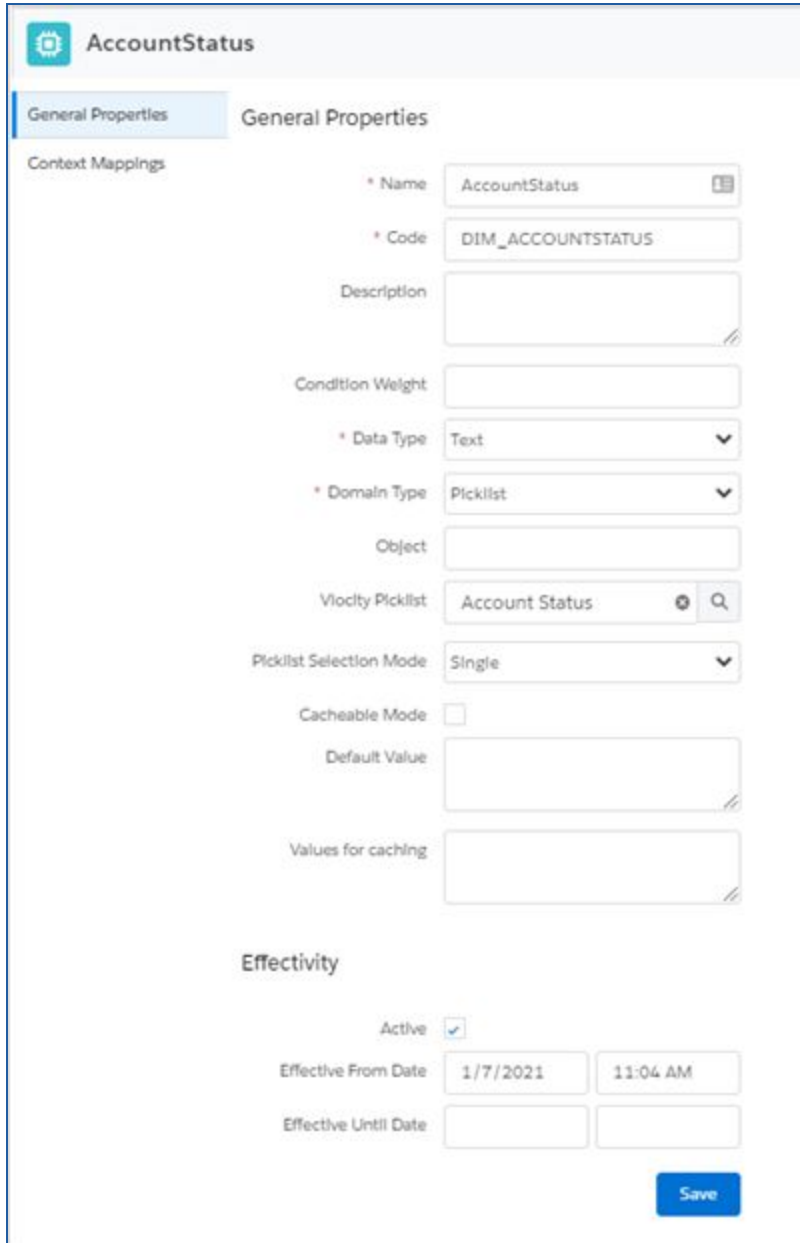
1. On the **Dashboard**, under Rules, click the plus sign (+) next to **Context Dimension** to create a new Context Dimension.
2. In the **New Context Dimension** dialog, enter the following information.

Field	Value	Notes
Name	AccountStatus	Context dimension names cannot contain any spaces due to requirements from the rule engine.
Code	DIM_ACCOUNTSTATUS	This is used when creating custom expressions and should be unique with no spaces.
Data Type	Text	This defines the type of information that will be contained in the variable, such as Text, Number, Date, DateTime or Boolean.
Domain Type	Picklist	Domain type describes how the rule condition values appear when creating a rule condition. You are using a picklist, but your domain type may also be either an object lookup or type in.
Vlocity picklist	Account Status	This is the picklist we created in the first task of this exercise.
Picklist Selection Mode	Single	The context rule service supports only single picklist selections.
Active	✓	Setting the Active flag is recommended, although it is not currently enforced.

Effective From **[today's date]**

Setting the Effective From (and Effective Until, if desired) fields is recommended, although it is not currently enforced.

3. Click **Save**.



The screenshot shows the 'AccountStatus' context rule configuration page. The 'General Properties' tab is selected. The 'Context Mappings' section contains the following fields:

- * Name: AccountStatus
- * Code: DIM_ACCOUNTSTATUS
- Description: (empty text area)
- Condition Weight: (empty text field)
- * Data Type: Text
- * Domain Type: Picklist
- Object: (empty text field)
- Velocity Picklist: Account Status
- Picklist Selection Mode: Single
- Cacheable Mode: ☐
- Default Value: (empty text area)
- Values for caching: (empty text area)

The 'Effectivity' section contains the following fields:

- Active: ☒
- Effective From Date: 1/7/2021 11:04 AM
- Effective Until Date: (empty date and time fields)

A 'Save' button is located at the bottom right of the form.



NOTE:

The Cacheable Mode, Default Value and Values for Caching properties are used only with the Digital Commerce API Caching feature. They are not used with the CPQPartition platform cache.

What is a context mapping?

A context mapping is the magic link that allows the rules engine to compare context dimensions to data stored in sObjects, calculated using a function, or typed in during design-time within a specified context scope. Context scopes and context dimensions must be defined before configuring context mapping.

Context mappings are stored in the org cache to optimize performance.

Task 4: Create a context mapping

1. On the left sidebar, click the **Context Mappings** facet.
2. Click **New Context Mapping**.
3. In the **Context Mapping** dialog, enter the following information.

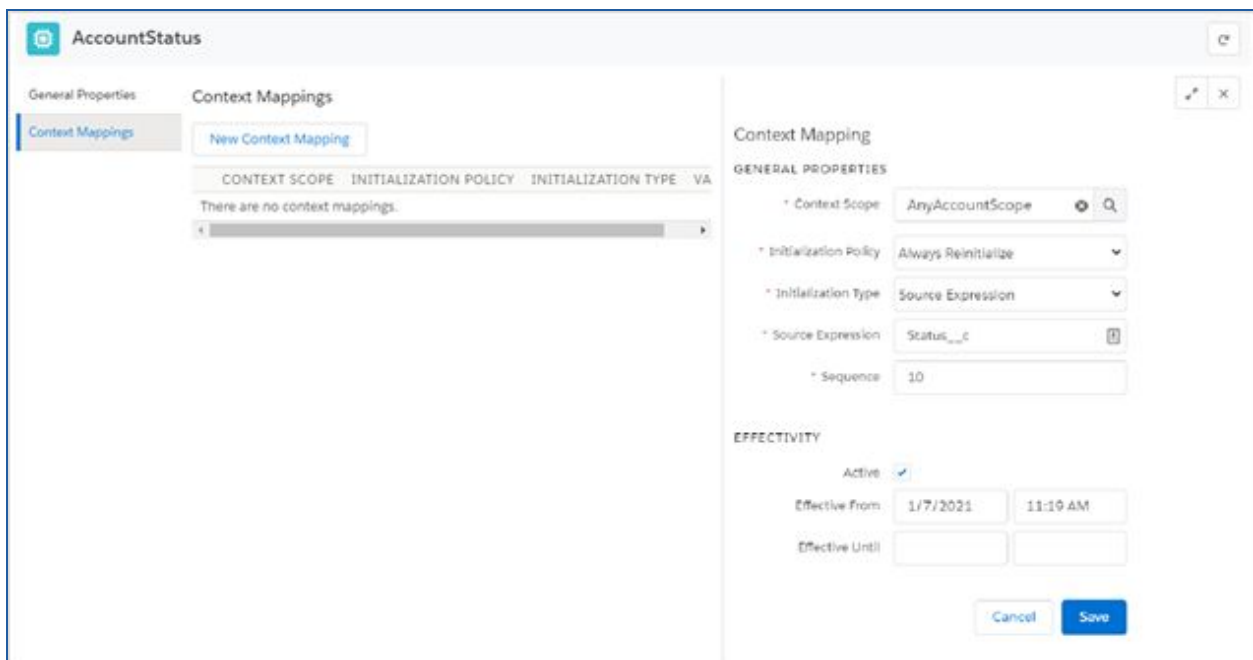
Field	Value	Notes
Context Scope	AnyAccountScope	
Initialization Policy	Always Reinitialize	For entity scopes (like this one), the rules engine will always reinitialize by default, essentially ignoring this selection.
Initialization Type	Source Expression	This specifies where to retrieve the data to use to compare to the rule condition. Here you are using the source expression for the account status, as outlined below.
Source Expression	Status__c	This field will appear once Source Expression is selected for Initialization

Type. This field should specify the API name of the sObject field that will be compared to the value stored in the context dimension.

Note that the value contains two underscores after “Status.”

Sequence	10
Active	✓
Effective From	[today's date]

- Click **Save**.
- In the **Product Console** tab navigation bar, click **X** to dismiss the **AccountStatus** tab.



The screenshot shows the Salesforce 'AccountStatus' context mapping configuration page. The left sidebar has tabs for 'General Properties' and 'Context Mappings', with 'Context Mappings' selected. Below the tabs is a 'New Context Mapping' button and a table with columns 'CONTEXT SCOPE', 'INITIALIZATION POLICY', 'INITIALIZATION TYPE', and 'VA'. The table is currently empty, showing 'There are no context mappings.' The right pane is titled 'Context Mapping' and contains two sections: 'GENERAL PROPERTIES' and 'EFFECTIVITY'. In 'GENERAL PROPERTIES', the 'Context Scope' is set to 'AnyAccountScope', 'Initialization Policy' is 'Always Reinitialize', 'Initialization Type' is 'Source Expression', 'Source Expression' is 'Status__c', and 'Sequence' is '10'. In the 'EFFECTIVITY' section, the 'Active' checkbox is checked, 'Effective From' is '1/7/2021 11:19 AM', and 'Effective Until' is empty. At the bottom right are 'Cancel' and 'Save' buttons.

Now that you’ve got the foundation in place, you can create the context rule.

Task 5: Create a context rule

The context rule contains header information and one or more conditions. For a qualification rule, evaluating the value against the rule condition determines whether the rule passes or fails.



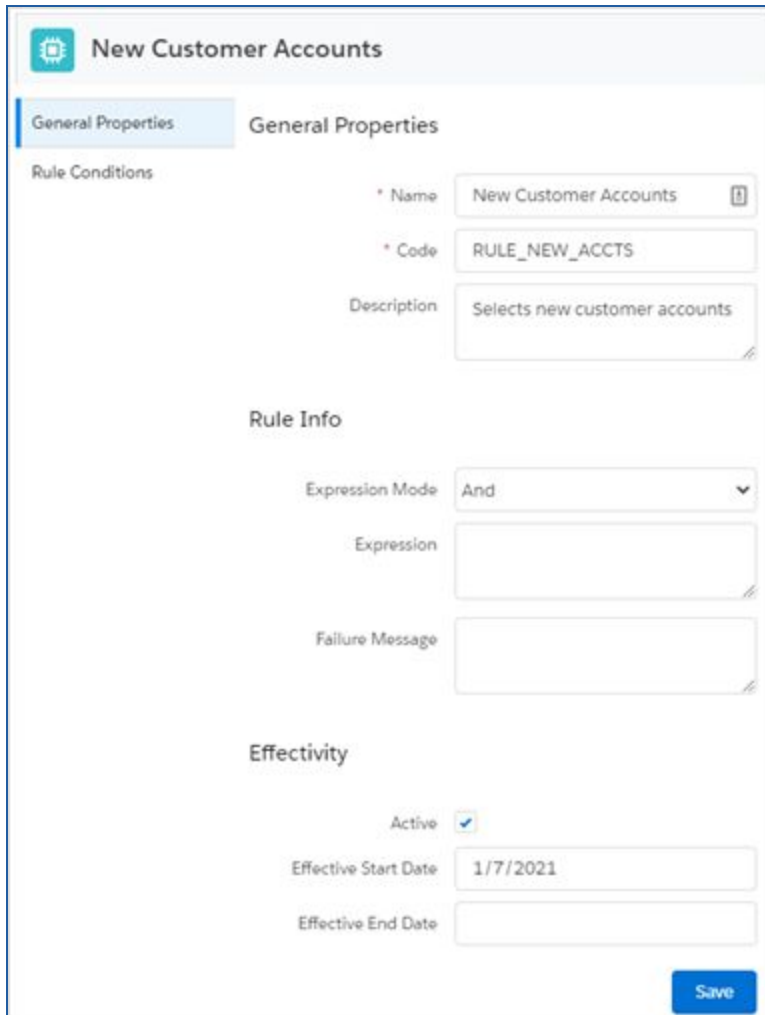
NOTE:

Context rules and rule conditions are cached in the org cache of the CPQPartition. After they are invoked the first time, you then need to clear the cache in order for changes to be reflected at run time.

1. In the **Dashboard** under **Rules**, click the plus icon (+) next to **Rule** to create a new rule.
2. In the **New Rule** dialog, enter the following information.

Field	Value	Notes
Name	New Customer Accounts	
Code	RULE_NEW_ACCTS	The Rule Code may be used within custom expressions and therefore should be unique and should not contain any spaces.
Description	Selects new customer accounts	Describes the purpose of the context rule.
Expression Mode	And	When you create multiple rule conditions, it is important to specify whether they are evaluated as AND, OR or a CUSTOM combination.
Active	✓	
Effective From	[today's date]	

3. Click **Save**.



NOTE:

Although there is a field to enter a failure message for the context rule, this is currently not in use. Failure messages should instead be set on the rule set, which you will cover later in this exercise.

Task 6: Create rule conditions

1. In the left sidebar, click the **Rule Conditions** facet.
2. Using the **Add Condition** dropdown menu, select **Simple**.
3. In the **Rule Condition** window on the right, enter the following information.

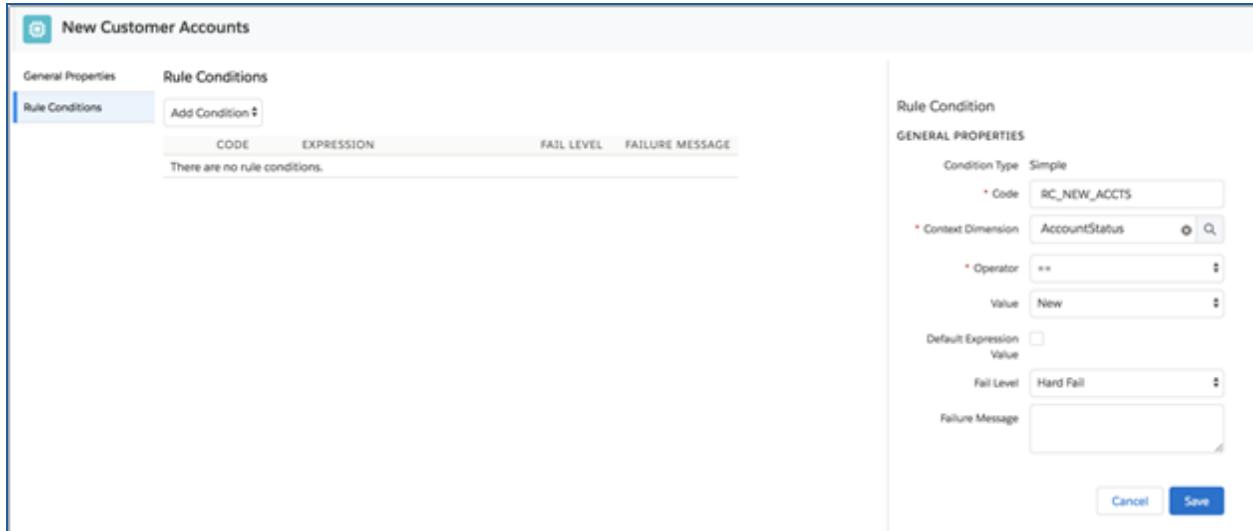
Field	Value	Notes
Code	RC_NEW_ACCTS	Rule Condition Code may not contain any spaces due to requirements from the rules engine.
Context Dimension	AccountStatus	Use the lookup to select the context dimension you created earlier in this exercise.
Operator	==	This is the operator used to compare the value to the context mapping.
Value	New	These values are coming from the AccountStatus Vlocity picklist items that you created earlier in this exercise.
Fail Level	Hard Fail	There is currently no functional difference in the Cart between hard fail or soft fail, other than a change in the color of the failure message (soft fail failure messages display in yellow, hard fail failure messages display in red). Both hard fail and soft fail are placed in the Disqualified list. However, if querying the shared catalog using the Cart-based APIs, the rules framework returns the PassIdList, which are objects that have passed all rule levels, and it also returns the HardFailList, SoftFailList and SoftPassList. These include the ids that failed, the rule sets they failed on and the failure messages.

4. Click **Save**.



NOTE:

The Default Expression Value is used only with the Digital Commerce API Caching feature. It is not used with the CPQPartition platform cache.



New Customer Accounts

General Properties | **Rule Conditions**

Rule Conditions

Add Condition

CODE	EXPRESSION	FAIL LEVEL	FAILURE MESSAGE
There are no rule conditions.			

Rule Condition

GENERAL PROPERTIES

Condition Type: Simple

* Code: RC_NEW_ACCTS

* Context Dimension: AccountStatus

* Operator: ==

Value: New

Default Expression: ☐

Fail Level: Hard Fail

Failure Message:

Cancel Save

5. In the **Product Console** tab navigation bar, click **X** to dismiss the **New Customer Accounts** tab.

Task 7: Create a rule set




NOTE:

Rule sets are cached in the org cache of the CPQPartition. After they are invoked the first time, you then need to clear the cache in order for changes to be reflected at run time.


1. In the **Dashboard** under Rules, click the plus sign (+) next to **Rule Set** to create a new Rule Set.
2. In the **New Rule Set** dialog, enter the following information.

Field	Value	Notes
Name	New Accounts Rule Set	
Description	Used to qualify new accounts	
Rule Type	Qualification	This controls how the rule will be used in the Cart.
Rule Purpose	Eligibility	This field currently is not functional but is planned for future use.
Expression Mode	And	Used to determine how multiple rules are evaluated. (Note: if you choose "Custom" here, you will also need to define the expression to be used in the Expression field immediately below.)
Failure Message	Valid only for new accounts	This message will display in the Disqualified subtab of the Cart when applied to a product or promotion. It is not used when applied to a price list or price list entry.
Active	✓	
Effective From	[today's date]	

3. Click **Save**.

 **New Rule Set**

General Properties

* Name 

Description

* Rule Type

* Rule Purpose



Rule Set Info

Expression Mode

Expression

Failure Message

Header Rule

Header Rule  

Effectivity

Active ☒

Effective Start Date

Effective End Date

Save



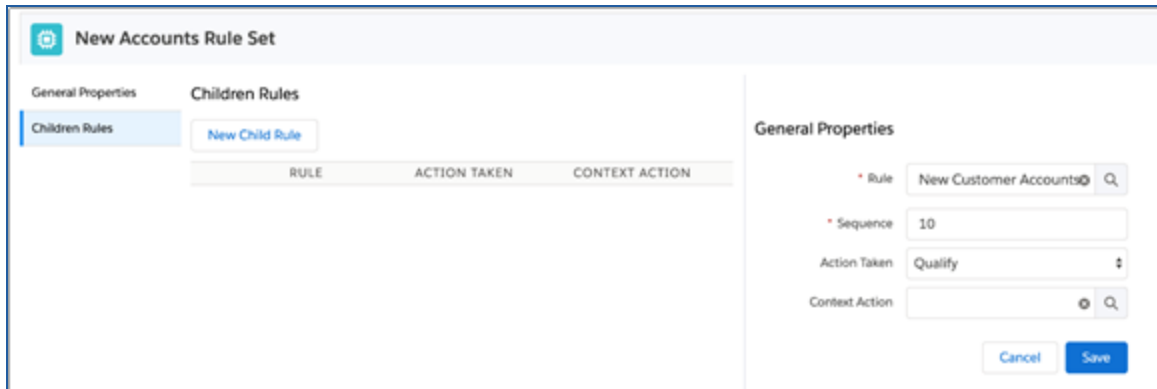
NOTE:

You are about to go and add your first rule to the rule set. When you have multiple child rules in a rule set, you can specify which one is executed first by selecting it from the Header Rule lookup. If the header rule does not pass, processing stops on the entire rule set. This allows you to make your rule set evaluation more efficient.

4. In the left sidebar, click the **Children Rules** facet.
5. Click **New Child Rule**.
6. In the **General Properties** dialog on the right, enter the following information

Field	Value	Notes
Rule	New Customer Accounts	You can select one or multiple rules as child rules of a rule set. If you select multiple child rules, the expression mode for the rule set and other factors determine how those rules are applied.
Sequence	10	
Action Taken	Qualify	This specifies the action to take when the rule evaluates to true. For a qualification context rule, the possible actions are to qualify or to not qualify. These correspond to the Qualified and Disqualified tabs in the Cart. The Policy Action option listed applies to penalty rules, and requires a corresponding Context Action to be specified. Therefore, you can ignore the Policy Action and Context Action for now.

7. Click **Save**.



New Accounts Rule Set

General Properties Children Rules

Children Rules [New Child Rule](#)

RULE	ACTION TAKEN	CONTEXT ACTION
------	--------------	----------------

General Properties

* Rule 🔍

* Sequence

Action Taken ⌵

Context Action ⚙️ 🔍

[Cancel](#) [Save](#)

8. In the **Product Console** tab navigation bar, click **X** to dismiss the **New Accounts Rule Set** tab.

Review

Confirm your understanding by answering these questions.

1. What is a context dimension? How is it used in a context rule?
2. Which is a context scope and how is it used?
3. What is a context mapping? What types of mappings (initialization types) can be created?
4. Which context rule components are cached?

Exercise 6-2: Deploying a Qualification Context Rule

Scenario

Ada has already created a promotion, the iPhone X Intro Promo, which offers a special deal for new customers. Eliza now wants to apply the new customer qualification context rule to the promotion so she can see it in action. Then, she's ready to test it in the Cart.

Goals




- Apply a rule to a promotion
- Test a qualification rule

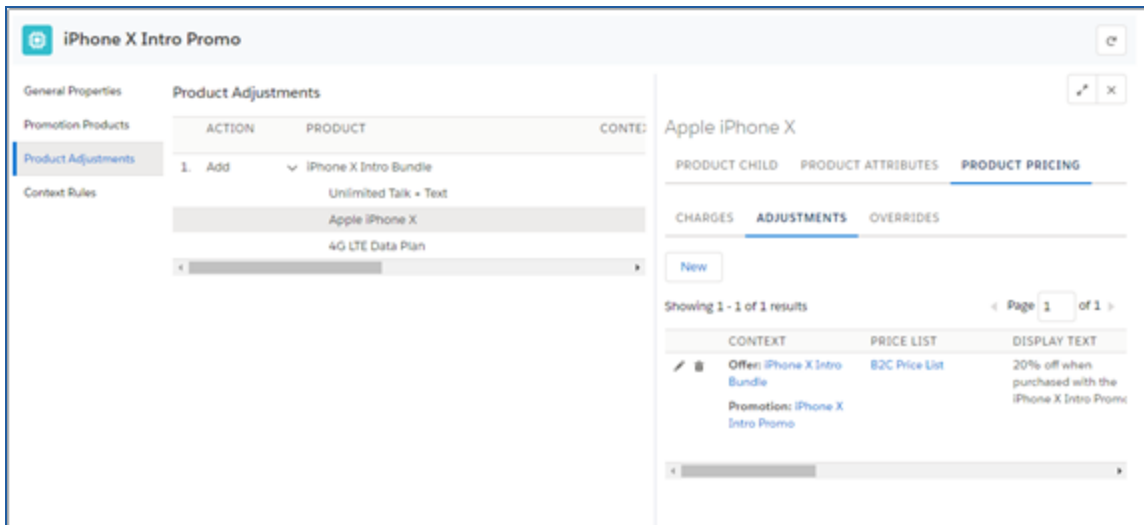
Tasks

1. Add a rule set to a promotion
2. Test the rule set in the Cart
3. Change the account status to New
4. Retest the rule set in the Cart

Time: 15 mins

Task 1: Add a rule set to a promotion

1. In the **Dashboard** under **Product Management**, click the **search icon**  next to **Promotion**.
2. In the **Search Promotion...** dialog, enter **iPhone**, and then hit **Enter** or click the **search icon** .
3. Click the **edit icon**  next to the **iPhone X Intro Promo**.
4. Review the promotion. Notice that it offers a 20% pricing adjustment on the iPhone X plus a \$20 pricing override on a data plan product.



The screenshot displays the 'iPhone X Intro Promo' configuration interface. On the left, the 'Context Rules' facet is selected in the sidebar. The main area is divided into 'General Properties' and 'Product Adjustments'. The 'Product Adjustments' table lists the following items:

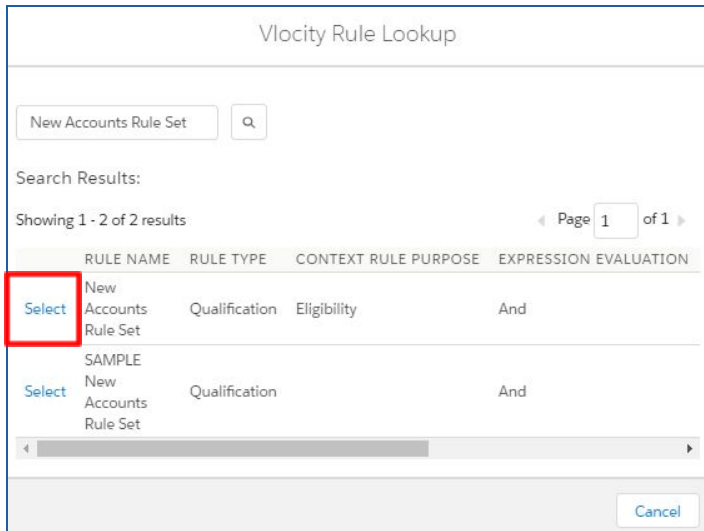
ACTION	PRODUCT	CONTEXT
1. Add	iPhone X Intro Bundle	
	Unlimited Talk + Text	
	Apple iPhone X	
	4G LTE Data Plan	

The right pane shows the 'Apple iPhone X' product details. Under the 'PRODUCT PRICING' tab, the 'ADJUSTMENTS' sub-tab is active, displaying a table with one result:

CONTEXT	PRICE LIST	DISPLAY TEXT
Offer: iPhone X Intro Bundle Promotion: iPhone X Intro Promo	B2C Price List	20% off when purchased with the iPhone X Intro Promo

5. In the left sidebar, click the **Context Rules** facet, then click **Add Rule Set**.

6. In the **Add Rule Set** dialog box, search for and select **New Accounts Rule Set**.



Velocity Rule Lookup

New Accounts Rule Set

Search Results:

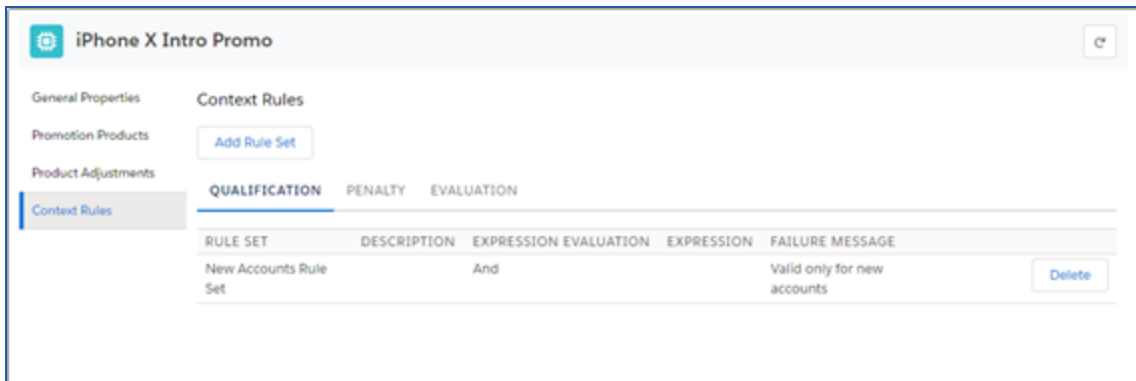
Showing 1 - 2 of 2 results

Page 1 of 1

	RULE NAME	RULE TYPE	CONTEXT RULE PURPOSE	EXPRESSION EVALUATION
Select	New Accounts Rule Set	Qualification	Eligibility	And
Select	SAMPLE New Accounts Rule Set	Qualification		And

Cancel

7. Click **Save**.
8. The rule set will appear in the Context Rules **QUALIFICATION** list for the promotion.
9. Click **PENALTY** and **EVALUATION**. This is where each type of rule respectively would display, if they have been assigned.



iPhone X Intro Promo

General Properties

Promotion Products

Product Adjustments

Context Rules

Add Rule Set

QUALIFICATION PENALTY EVALUATION

RULE SET	DESCRIPTION	EXPRESSION EVALUATION	EXPRESSION	FAILURE MESSAGE
New Accounts Rule Set		And		Valid only for new accounts

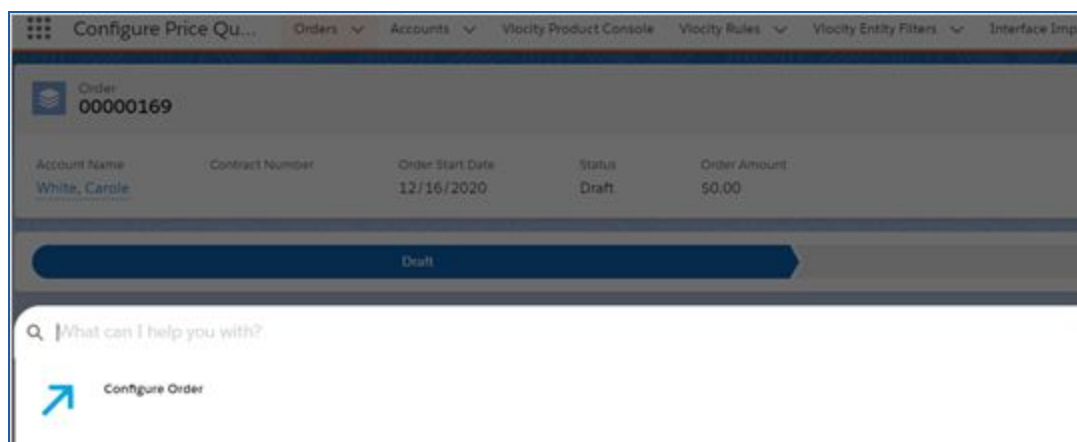
Delete

Task 2: Test the rule set in the Cart

1. In the Lightning navigation bar, click the **Orders** tab.
2. Click **New**.
3. In the **New Order** dialog, enter the following information.

Field	Value
Order Name	Context Rule Test Order - Promo
Account Name	White, Carole
Order Start Date	[today's date]

4. Click **Save**
5. From the Power Launcher, click **Configure Order**. This invokes the Cart.

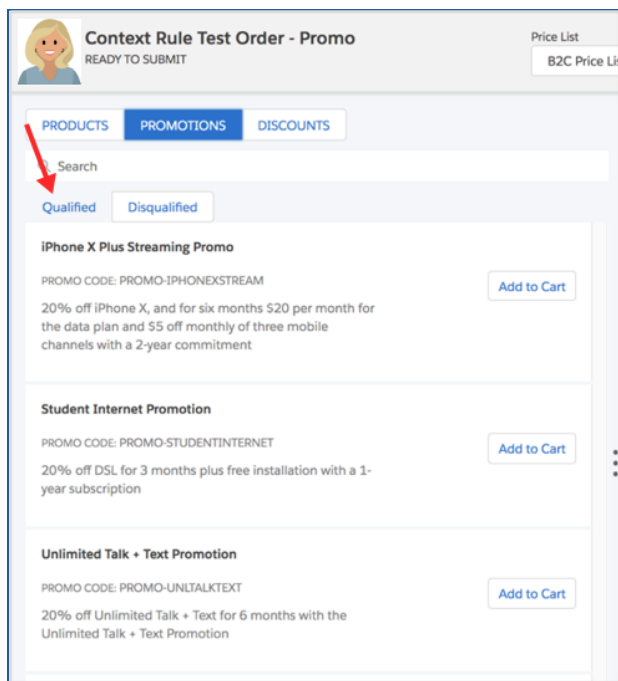


NOTE:

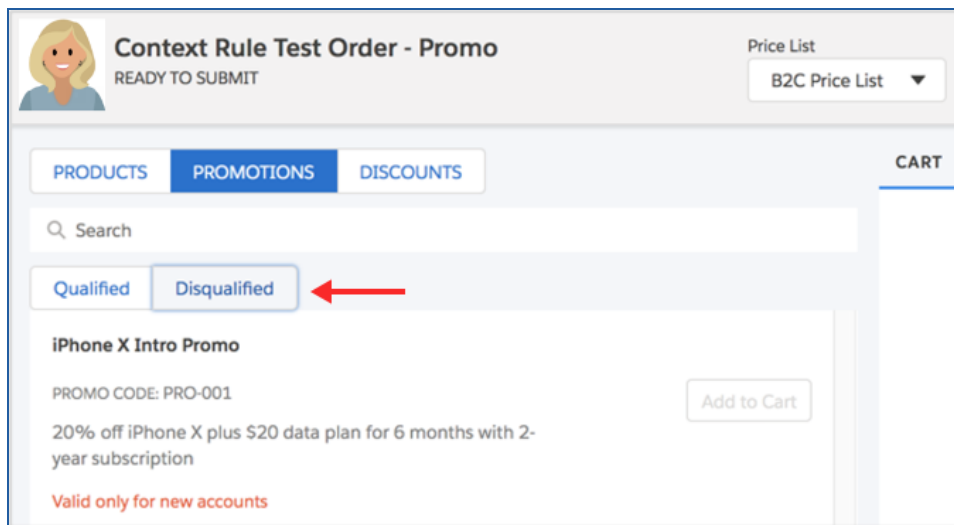
The Power Launcher is a Lightning Web Component (LWC) that gets and displays a list of OmniStudio Actions. Configure Order is set up for your convenience in the training playground.

6. Using the price list dropdown menu in the cart header, select **B2C Price List**. The product list will dynamically refresh.
7. In the product list pane, click the **PROMOTIONS** tab

8. Notice that the **iPhone X Intro Promo** does not appear on the **Qualified** subtab for Carole White.



9. Click the **Disqualified** subtab.

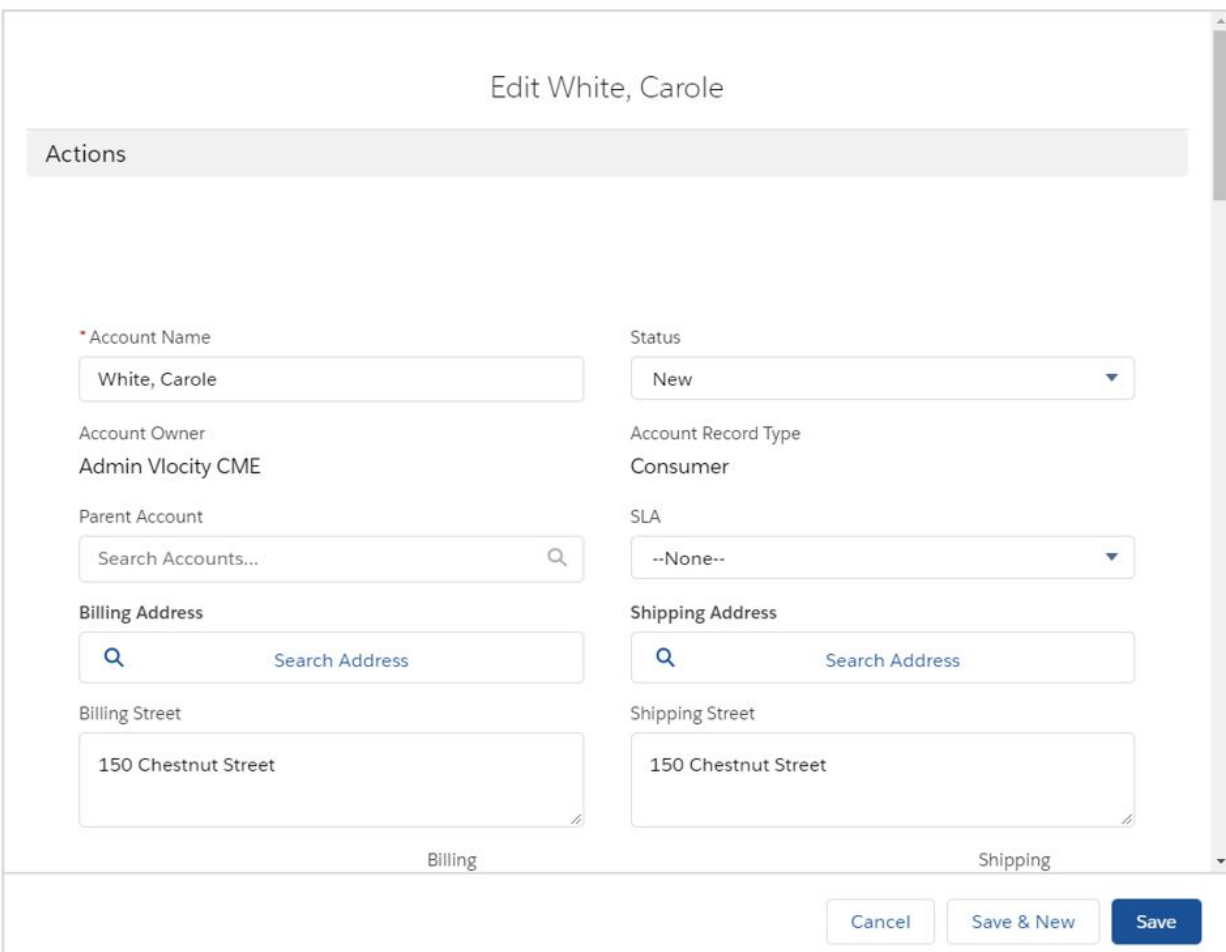


10. Notice that the promotion appears, and the red failure message from the rule set states that this promotion is only valid for new accounts.

Next, you'll change Carole White's account status to see if she qualifies for the promotion.

Task 3: Change the account status to New

1. In the cart header, click **View Record** to navigate back to the order.
2. In the order header, click **White, Carole** under **Account Name**.
3. Click **Edit**.
4. Change **Status** to **New**.
5. Click **Save**.



Edit White, Carole

Actions

* Account Name
White, Carole

Status
New

Account Owner
Admin Vlocity CME

Account Record Type
Consumer

Parent Account
Search Accounts...

SLA
--None--

Billing Address
Search Address

Shipping Address
Search Address

Billing Street
150 Chestnut Street

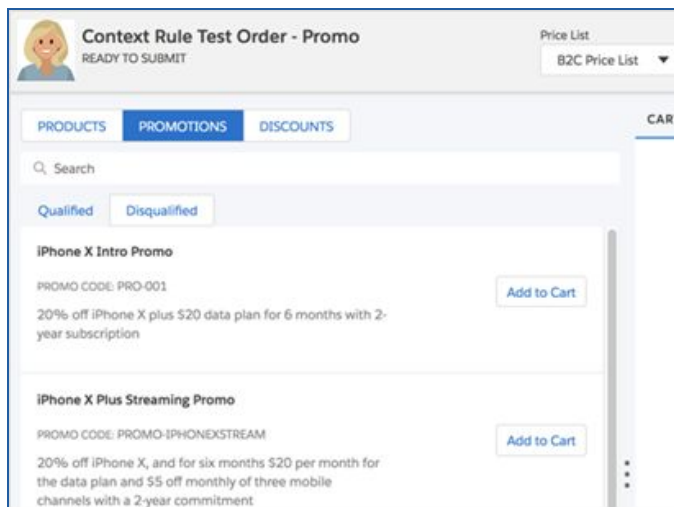
Shipping Street
150 Chestnut Street

Billing Shipping

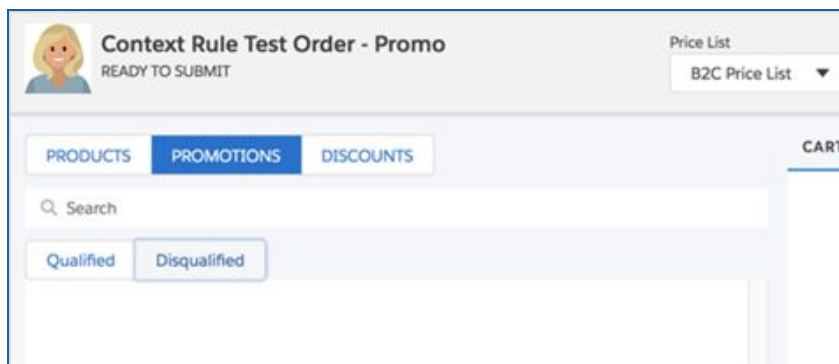
Cancel Save & New Save

Task 4: Retest the rule set in the Cart

1. In the Lightning navigation bar, click **Orders**.
2. Click the order number link of **Context Rule Test Order - Promo**.
3. From the **Power Launcher**, click **Configure Order** to return to the cart.
4. In the product list pane, click **PROMOTIONS**.
5. Notice that the promotion now appears on the **Qualified** subtab for Carole White, and the **Add to Cart** button is enabled.



6. Click the **Disqualified** subtab.
7. Notice that the promotion no longer appears on the Disqualified subtab for Carole White.



Troubleshooting Guide

Before you proceed, ensure that the iPhone X Intro promotion displays on the PROMOTIONS Qualified subtab. If it does not appear there and instead appears on the PROMOTIONS Disqualified subtab, your context rule is not working, and it will affect the subsequent exercises. You can troubleshoot by checking the following settings or by comparing your rule with the SAMPLE rule in your training playground.



NOTE:

Context rules, rule conditions and rule sets are cached in the org cache of the CPQPartition platform cache. When troubleshooting, you will need to clear the cache in order for changes to be reflected at run time. To clear the cache, go to the **Vlocity CMT Administration** tab. You may need to select this from the **More** dropdown. Under the Admin Console heading, click **Maintenance Jobs**, and click the **Start** button beside CLEAR MANAGED PACKAGE CACHE.

- **Account - Status?** The **White, Carole** account status field should be set to **New**.

Account Name	Status
White, Carole	New

- **Context Dimension - Name?** The **AccountStatus** context dimension **Name** should have no spaces, as shown below.

* Name	AccountStatus
--------	---------------

- **Context Mapping - Source Expression?** The context mapping for the **AccountStatus** context dimension should have a **Source Expression** with two underscores, as shown below.

* Source Expression	Status__c
---------------------	-----------

- **Context Rule - Code?** The Code should have no spaces, as shown below.

* Code	RULE_NEW_ACCTS
--------	----------------

Review

Confirm your understanding by answering these questions.

1. How can you see what context rules are applied to any given product or promotion?
How are they organized?
2. Where does the failure message come from that displays on the Disqualified promotions list?
3. Why did we not have to refresh the platform cache in order to see the change in Carole White's account status?

Exercise 6-3: Creating Context Rules for Price Lists

Scenario

When it comes down to it, the rubber meets the road in pricing. In the same way that context rules can control which products and promotions display in the Cart, Eliza can use context rules to control which price lists display. She can create a rule to make the B2C price list available only to consumer accounts, and another rule to make the B2B price list available only to business accounts.

Goals

- Create context rules for price lists
- Control which price lists display in the Cart
- Create context dimensions with type in values

Tasks

1. Create a context dimension that will allow users to type in values for rule conditions
2. Create a context mapping to account record type
3. Create a context rule for consumers account
4. Create a context rule for non-consumer accounts
5. Create a rule set for consumer accounts
6. Create a rule set for business accounts
7. Assign the new rules sets to price lists
8. Test the new rules in the Cart

Time: 30 mins

How are context rules used for pricing?

In the same way that context rules can be used to determine eligibility for products and promotions, they can also be used to determine eligibility for price lists and price list entries. Context rules can be assigned to price lists (parent and child) as well as price list entries.

- When a context rule is used with a parent price list, it controls what price lists the user will see in the Cart price list dropdown menu.
- When a context rule is used with a child price list or a price list entry, it is used to determine what price to apply to products in the Cart based upon rule criteria such as customer segment, account status or geography.

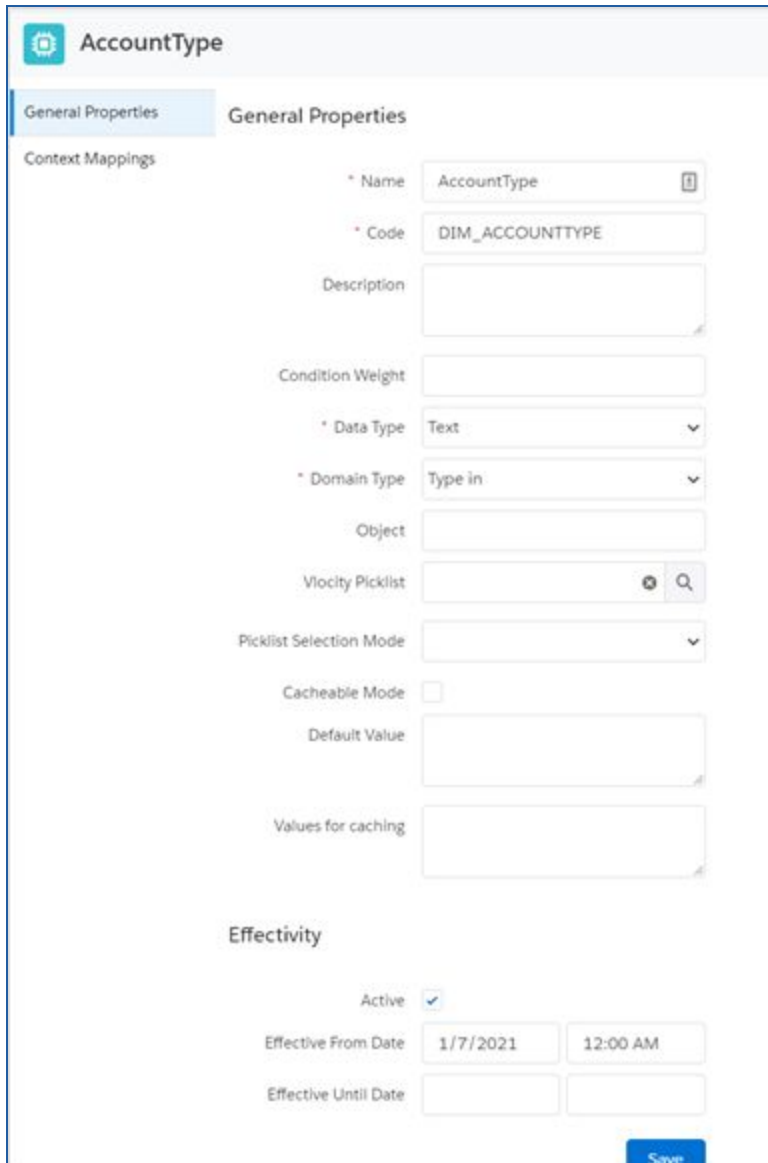
In this exercise, you'll create context rules to control which price lists display in the Cart. You will create a context rule for consumer accounts to qualify them for the B2C Price List, and then another context rule for all other account types to qualify them for the B2B Price List.

Task 1: Create a context dimension that will allow users to type in values for rule conditions

1. In the Lightning tab navigation bar, click **Vlocity Product Console**.
2. Under **Rules**, click the plus sign (+) next to **Context Dimension**.
3. In the **New Context Dimension** dialog, enter the following information.

Field	Value	Notes
Name	AccountType	Context dimension names cannot contain any spaces due to requirements from the rules engine.
Code	DIM_ACCOUNTTYPE	
Data Type	Text	
Domain Type	Type in	This setting allows users to type in values when creating the rule condition. In the previous exercise, you used a picklist, which ensures data consistency. However, sometimes you may want to allow users to enter freeform values, and this setting provides that functionality.
Active	✓	Setting the Active flag is recommended, although it is not currently enforced.
Effective From Date	[today's date]	Setting the Effective From (and Effective Until, if desired) fields is recommended, although they are not currently enforced.

4. Click **Save**.



The screenshot shows the Salesforce configuration page for a context rule named "AccountType". The page has a header with the Salesforce logo and the name "AccountType". Below the header, there are two tabs: "General Properties" (selected) and "Context Mappings". The "General Properties" tab contains the following fields:

- Name:** AccountType
- Code:** DIM_ACCOUNTTYPE
- Description:** (empty text box)
- Condition Weight:** (empty text box)
- Data Type:** Text
- Domain Type:** Type in
- Object:** (empty text box)
- Velocity Picklist:** (empty text box with a search icon)
- Picklist Selection Mode:** (empty dropdown menu)
- Cacheable Mode:** (checkbox, unchecked)
- Default Value:** (empty text box)
- Values for caching:** (empty text box)

Below the "General Properties" section, there is an "Effectivity" section with the following fields:

- Active:** (checkbox, checked)
- Effective From Date:** 1/7/2021
- Effective From Time:** 12:00 AM
- Effective Until Date:** (empty text box)
- Effective Until Time:** (empty text box)

At the bottom right of the page, there is a blue "Save" button.

Task 2: Create a context mapping to account record type



1. On the left sidebar, click the **Context Mappings** facet.
2. Click **New Context Mapping**.
3. In the **Context Mapping** dialog, enter the following information.


Field	Value	Notes
Context Scope	AnyAccountScope	
Initialization Policy	Always Reinitialize	For entity scopes (like this one), the rules engine will always reinitialize by default, essentially ignoring this selection.
Initialization Type	Source Expression	This tells Industries CPQ where to retrieve the data to use to compare to the rule condition. Here you are using the source expression defined in the Source Expression field below.
Source Expression	<code>RecordType.Name</code>	This field will display once <i>Source Expression</i> is selected for Initialization Type. <code>RecordType.Name</code> is the API name of the <code>sObject</code> field that will be compared to the value stored in the context dimension. Note in this example, we are tracing the data hierarchy to map this to the name of the account record type. When this is used in a rule run on an order, the hierarchy will be <code>Order.Account.RecordType.Name</code> .
Sequence	10	
Active	✓	
Effective From	[today's date]	


4. Click **Save**.


Context Mapping

GENERAL PROPERTIES

* Context Scope  

* Initialization Policy 

* Initialization Type 

* Source Expression 

* Sequence

EFFECTIVITY

Active ☒

Effective From

Effective Until

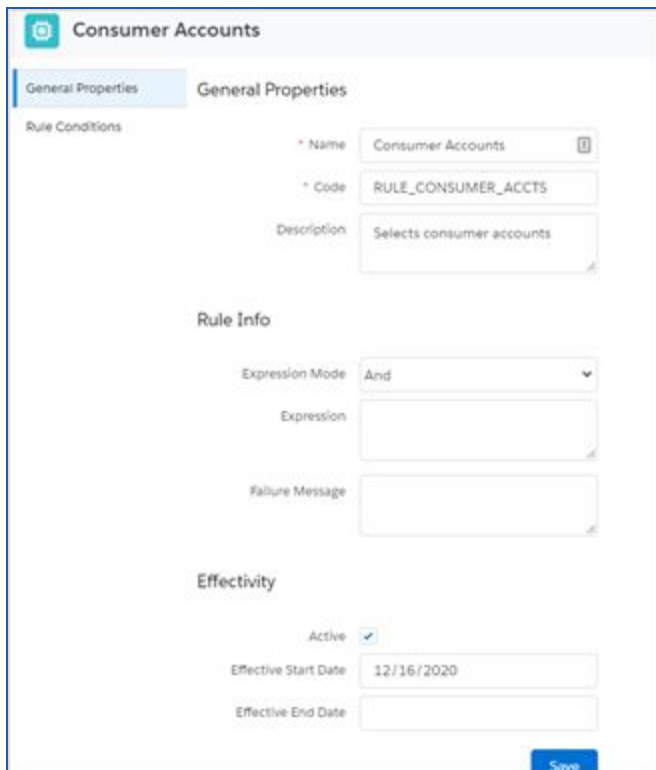
5. In the **Product Console** tab navigation bar, click **X** to dismiss the **AccountType** tab.

Task 3: Create a context rule for consumer's account

1. In the **Dashboard** under **Rules**, click the plus icon (+) next to **Rule**.
2. In the **New Rule** dialog, enter the following information.

Field	Value	Notes
Name	Consumer Accounts	
Code	RULE_CONSUMER_ACCTS	Rule Code must be unique and cannot contain any spaces.
Description	Selects consumer accounts	
Expression Mode	And	
Active	✓	
Effective From	[today's date]	

- Click **Save**.



- In the left sidebar, click the **Rule Conditions** facet.
- Using the **Add Condition** dropdown menu, select **Simple**.
- In the **Rule Condition** window on the right, enter the following information.

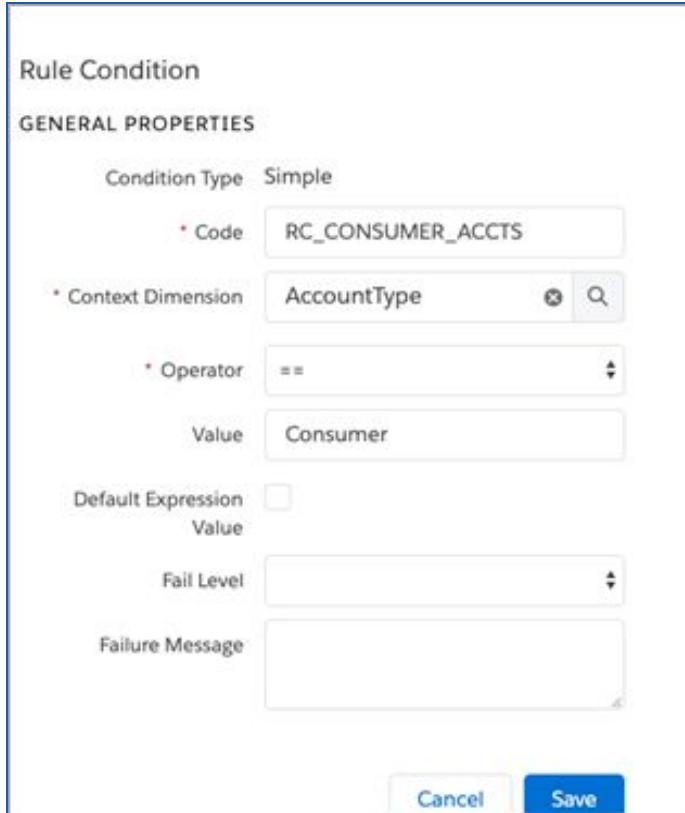
Field	Value	Notes
Code	RC_CONSUMER_ACCTS	Rule Condition Code may not contain any spaces due to requirements from the rules engine.
Context Dimension	AccountType	This is the context dimension you created earlier in this exercise.
Operator	==	
Value	Consumer	Notice that this is a freeform text entry field due to the Domain Type that was set in the context dimension.

7. Click **Save**.



NOTE:

The Default Expression Value is used only with the Digital Commerce API Caching feature. It is not used with the CPQPartition platform cache.



The screenshot shows the 'Rule Condition' dialog box with the following fields and values:

- Condition Type:** Simple
- * Code:** RC_CONSUMER_ACCTS
- * Context Dimension:** AccountType
- * Operator:** ==
- Value:** Consumer
- Default Expression Value:** ☐
- Fail Level:** (empty dropdown)
- Failure Message:** (empty text area)
- Buttons:** Cancel, Save

8. In the **Product Console** tab navigation bar, click **X** to dismiss the **Consumer Accounts** tab.

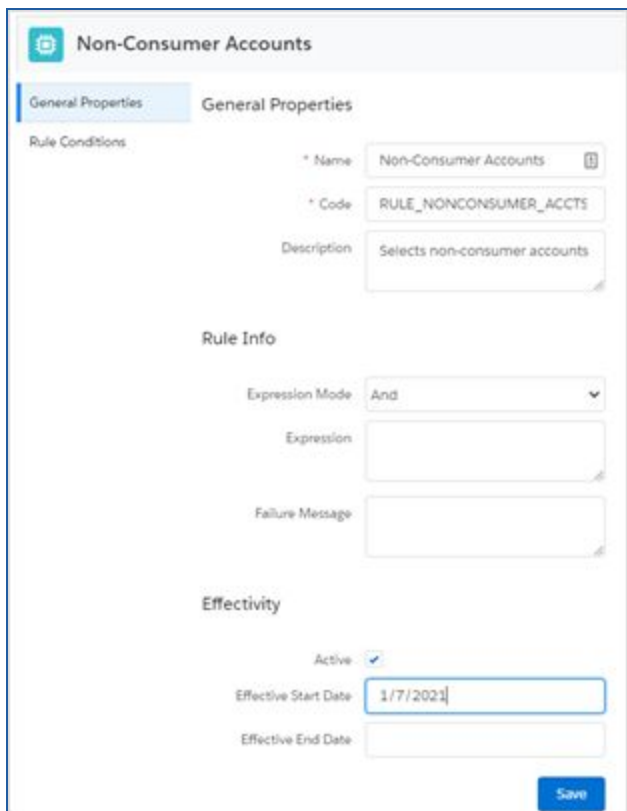
Now that you have a rule for consumer accounts, which you will use with the B2C Price List, you need to create another rule for non-consumer accounts, which you will use with the B2B Price List.

Task 4: Create a context rule for non-consumer accounts

1. In the **Dashboard** under **Rules**, click the plus icon (+) next to **Rule** to create a new rule.
2. In the **New Rule** dialog, enter the following information.

Field	Value	Notes
Name	Non-Consumer Accounts	
Code	RULE_NONCONSUMER_ACCTS	Rule Code must be unique and cannot contain any spaces.
Description	Selects non-consumer accounts	
Expression Mode	And	
Active	✓	
Effective From	[today's date]	

- Click **Save**.



Non-Consumer Accounts

General Properties

Rule Conditions

* Name: Non-Consumer Accounts ⓘ

* Code: RULE_NONCONSUMER_ACCTS

Description: Selects non-consumer accounts

Rule Info

Expression Mode: And ▼

Expression:

Failure Message:

Effectivity

Active: ☒

Effective Start Date: 1/7/2021

Effective End Date:

Save

- In the left sidebar, click the **Rule Conditions** facet.
- Using the **Add Condition** dropdown menu, select **Simple**.
- In the **Rule Condition** window on the right, enter the following information.

Field	Value	Notes
Code	RC_NONCONSUMER_ACCTS	Rule Condition Code must be unique and may not contain any spaces due to requirements from the rules engine.
Context Dimension	AccountType	This is the context dimension we created earlier in this exercise.

Operator	!=	This will select all items with values that are not equal to “Consumer.”
Value	Consumer	Notice that this is a freeform text entry field due to the Domain Type that was set in the context dimension.

7. Click **Save**.



NOTE:

As discussed previously the Default Expression Value is used only with the Digital Commerce API Caching feature. It is not used with the CPQPartition platform cache. You should also not set the fail level or failure message at the individual rule level. Instead, this is done on the rule set.

Rule Condition

GENERAL PROPERTIES

Condition Type

Simple

* Code

RC_NONCONSUMER_ACCTS

* Context Dimension

AccountType

* Operator

!=

Value

Consumer

Default Expression Value

☐

Fail Level

Failure Message

Cancel

Save

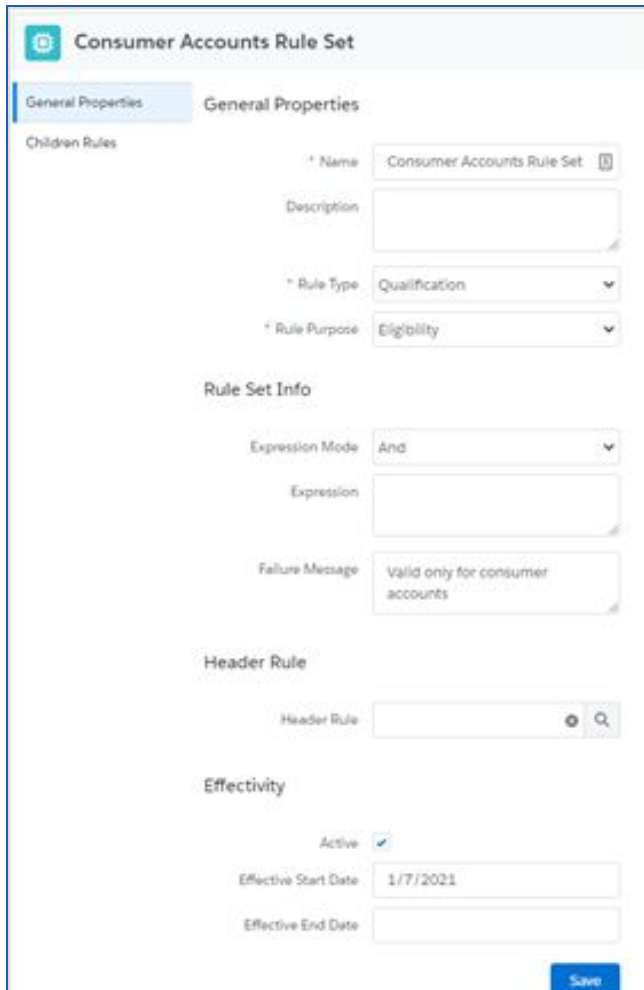
-
8. In the **Product Console** tab navigation bar, click **X** to dismiss the **Non-Consumer Accounts** tab.

Task 5: Create a rule set for consumer accounts

1. In the **Dashboard** under Rules, click the plus sign (+) next to **Rule Set** to create a new rule set.
2. In the **New Rule Set** dialog, enter the following information.

Field	Value	Notes
Name	Consumer Accounts Rule Set	
Rule Type	Qualification	This is a qualification rule set as we are determining the eligibility of the customer for a particular price list.
Rule Purpose	Eligibility	This field currently is not functional but is planned for future use.
Expression Mode	And	
Failure Message	Valid only for consumer accounts	Failure messages do not display for price lists or price list entries. However, since context rules are designed to be reusable, it is good practice to fill this message in anyway for a time in the future when this same rule set may be used to qualify products or promotions (which would allow the message to display on the Disqualified subtab).
Active	✓	
Effective From	[today's date]	

3. Click **Save**.



Consumer Accounts Rule Set

General Properties

Children Rules

* Name: Consumer Accounts Rule Set

Description:

* Rule Type: Qualification

* Rule Purpose: Eligibility

Rule Set Info

Expression Mode: And

Expression:

Failure Message: Valid only for consumer accounts

Header Rule

Header Rule:

Effectivity

Active: ☒

Effective Start Date: 1/7/2021

Effective End Date:

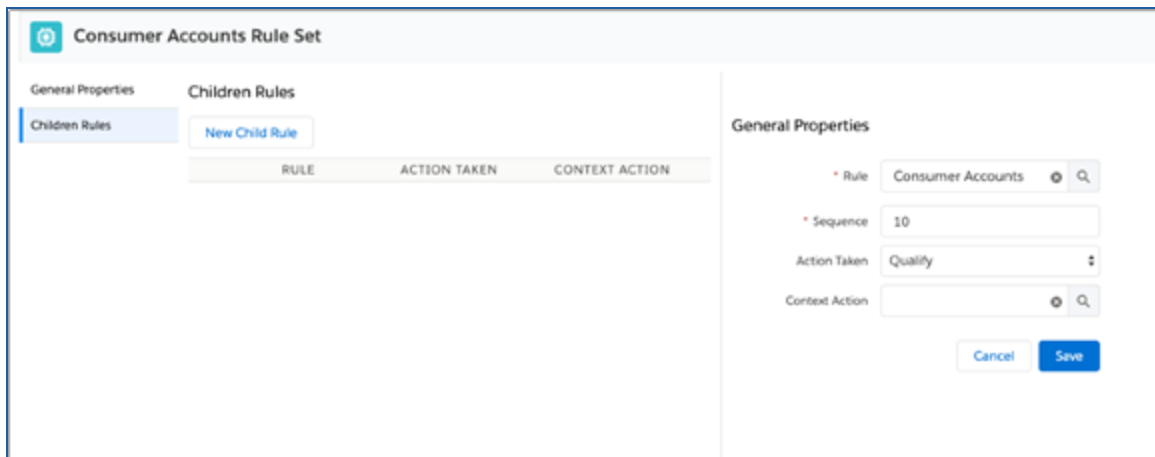
Save

4. In the left sidebar, click the **Children Rules** facet.
5. Click **New Child Rule**.

6. In the **General Properties** dialog on the right, enter the following information.

Field	Value	Notes
Rule	Consumer Accounts	You can select one or multiple rules as child rules of a rule set. If you select multiple child rules, the expression mode for the rule set and other factors determine how those rules are applied.
Sequence	10	
Action Taken	Qualify	This specifies the action to take when this child rule evaluates to true.

7. Click **Save**.



8. In the **Product Console** tab navigation bar, click **X** to dismiss the **Consumer Accounts Rule Set** tab.

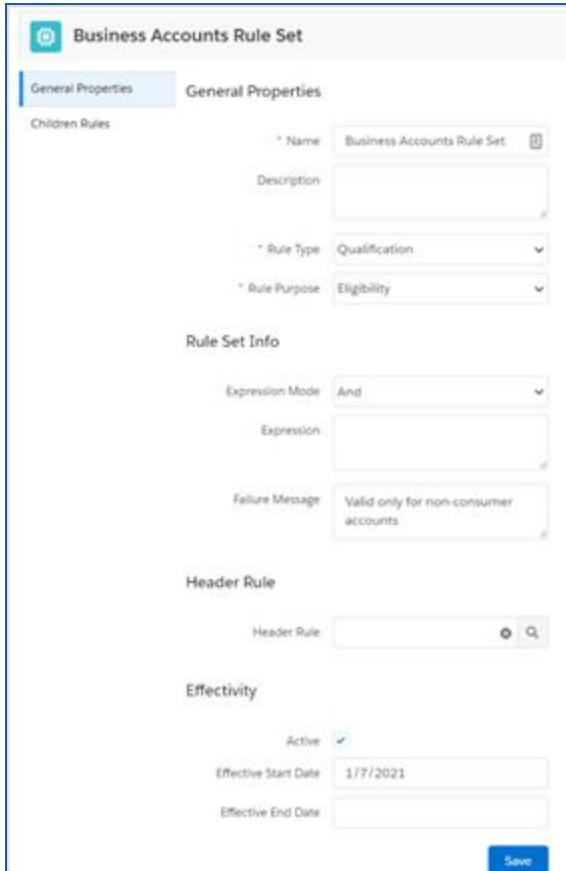
Just as you did with rules, now you need an analogous rule set for the non-consumers. (Otherwise, consumer accounts would see both the B2C and the B2B price lists without restriction.)

Task 6: Create a rule set for business accounts

1. In the **Dashboard** under Rules, click the plus sign (+) next to **Rule Set** to add a rule set for business accounts.
2. In the **New Rule Set** dialog, enter the following information.

Field	Value
Name	Business Accounts Rule Set
Rule Type	Qualification
Rule Purpose	Eligibility
Expression Mode	And
Failure Message	Valid only for non-consumer accounts
Active	✓
Effective Start Date	[today's date]

3. Click **Save**.

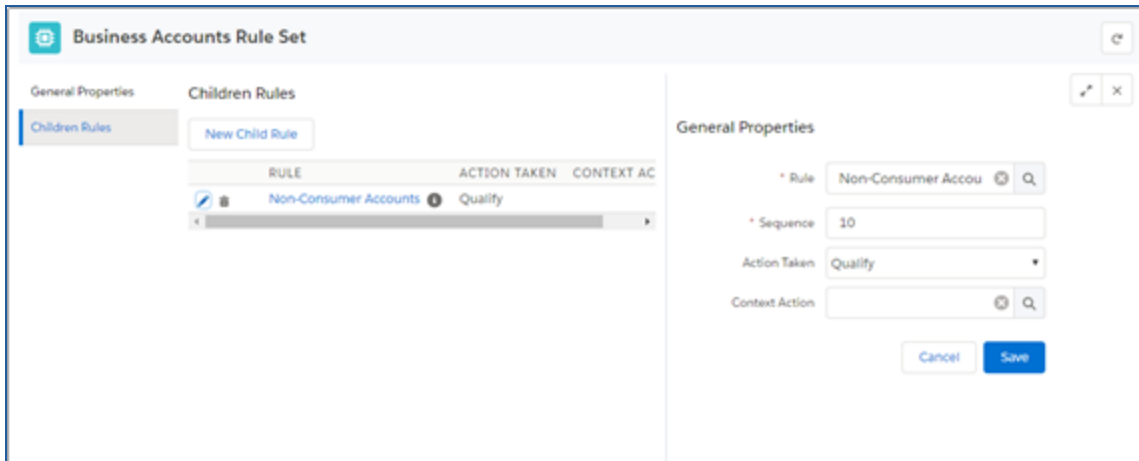


The screenshot shows the 'Business Accounts Rule Set' configuration page. The 'General Properties' tab is active. The 'Children Rules' section is empty. The 'General Properties' section includes fields for Name (Business Accounts Rule Set), Description, Rule Type (Qualification), Rule Purpose (Eligibility), Expression Mode (And), Expression, Failure Message (Valid only for non-consumer accounts), Header Rule, and Effectivity (Active checkbox checked, Effective Start Date 1/7/2021, Effective End Date). A 'Save' button is at the bottom right.

4. In the left sidebar, click the **Children Rules** facet.
5. Click **New Child Rule**.
6. In the **General Properties** dialog on the right, enter the following information.

Field	Value
Rule	Non-Consumer Accounts
Sequence	10
Action Taken	Qualify

7. Click **Save**.






Business Accounts Rule Set



General Properties | Children Rules

Children Rules

New Child Rule



RULE	ACTION TAKEN	CONTEXT AC
  Non-Consumer Accounts 	Qualify	

General Properties

* Rule: Non-Consumer Accou  

* Sequence: 10

Action Taken: Qualify

Context Action:  


Cancel Save


8. In the **Product Console** tab navigation bar, click **X** to dismiss the **Business Accounts Rule Set** tab.

Task 7: Assign the new rules sets to price lists

1. In the **Console Dashboard** under **Pricing** next to **Price List**, click the **search icon**.



2. Next to the **Search Price List...** entry dialog, click the **search icon**.  The list of all price lists is returned.

3. Click the name of **B2B Price List** or click the **Edit icon**  to open the B2B Price List for editing.

4. In the left sidebar, click the **Context Rules** facet.

5. Under **Context Rules**, click **Add Rule Set**.

6. In the **Add Rule Set** dialog, click the **search icon**. 

7. Using the **Vlocity Rule Lookup** dialog, enter `business` and click the **search icon**.

8. Click **Select** next to **Business Accounts Rule Set**.

Vlocity Rule Lookup

Business Accounts

Search Results:

Showing 1 - 2 of 2 results ◀ Page 1 of 1 ▶

	RULE NAME	RULE TYPE	CONTEXT RULE PURPOSE	EXPRESSION EVALUATION
Select	Business Accounts Rule Set	Qualification	Eligibility	And
Select	SAMPLE Business Accounts Rule Set	Qualification		And

9. In the **Add Rule Set** dialog, click **Save**.

B2B Price List

General Properties | Context Rules

Hierarchy


Price List Entries


Standalone Pricing Elements

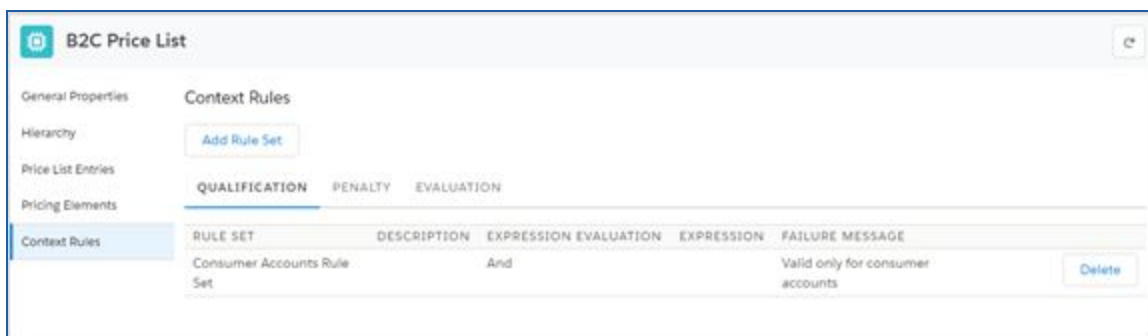
Context Rules

QUALIFICATION		PENALTY	EVALUATION
RULE SET	DESCRIPTION	EXPRESSION EVALUATION	EXPRESSION FAILURE MESSAGE
Business Accounts Rule Set		And	Valid only for non-consumer accounts <input type="button" value="Delete"/>

10. In the Console's tab navigation bar, click the **Search Price List** tab and search for the B2C Price List.

11. Click the name of **B2C Price List** or click the **Edit** icon  to open the B2C Price List for editing.

12. In the left sidebar, click the **Context Rules** facet.
13. Under **Context Rules**, click **Add Rule Set**.
14. In the **Add Rule Set** dialog, click the **search icon**. 
15. Using the **Velocity Rule Lookup** dialog, enter `consumer` and click the **search icon**.
16. Click **Select** next to **Consumer Accounts Rule Set**.
17. In the **Add Rule Set** dialog, click **Save**.



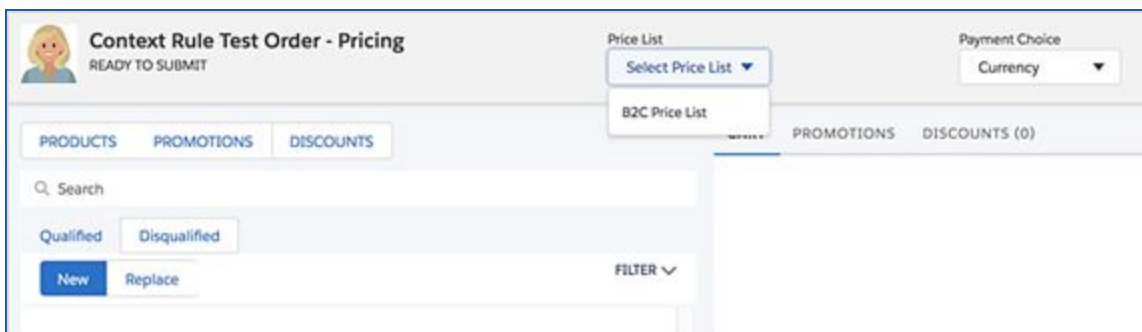
You have now completed applying context rules to your price lists such that only consumer-type customers will have access to the B2C Price List, and non-consumer customers will have access to the B2B Price List. The next step is to test it all out!

Task 8: Test the new rules in the Cart

1. In the Lightning navigation bar, click the **Orders** tab.
2. Click **New**.
3. In the **New Order** dialog, enter the following information.

Field	Value
Order Name	Context Rule Test Order - Pricing
Account Name	White, Carole
Order Start Date	[today's date]

4. Click **Save**.
5. From the Power Launcher, click **Configure Order**. This invokes the Cart.
6. Click the price list dropdown menu in the cart header. Notice that **B2C Price List** is the only price list that displays. Carole qualifies for this price list because of the Consumer Accounts Rule Set, and the B2B Price List does not display because the Business Accounts Rule Set is preventing it.



Troubleshooting Guide

Before you proceed, ensure that the B2C Price List displays for the White, Carole account, and that the B2B Price List does not. If either of these is not the case, your context rules are not working, and it may affect the subsequent exercises. You can troubleshoot by checking the following settings or by comparing your rule with the SAMPLE rule in your training playground.



NOTE:

Context rules, rule conditions and rule sets are cached in the org cache of the CPQPartition platform cache. When troubleshooting, you will need to clear the cache in order for changes to be reflected at run time. To clear the cache, go to the **Vlocity CMT Administration** tab in the Lightning Navigation toolbar then select **Maintenance Jobs**, and click the **Start** button next to **CLEAR MANAGED PACKAGE CACHE**.

- **Context Dimension - Name?** The AccountType context dimension Name should have no spaces, as shown below.

* Name

- **Context Mapping - Source Expression?** The context mapping for the AccountType context dimension should have a Source Expression with no spaces, as shown below.

* Source Expression

- **Context Rule - Code?** The rule Code should have no spaces, as shown below.

* Code

- **Context Rule - Rule Condition Value?** Check to ensure that the rule condition value has no typos.

Value

Review

Confirm your understanding by answering these questions.

1. Why would you want to assign a context rule to a price list?
2. What is a Type In Domain Type? What does it do?
3. Are context rules for price lists weighted?

Exercise 6-4: Creating Context Rules for Price List Entries

Scenario

Ada can create multiple prices for any given product using Industries CPQ. She can create one price for new customers, another price for orders placed over the web, and so on. With all of these prices, she needs to help the Cart know which price to apply when. Eliza uses context rules to define the conditions under which each price list entry applies, and for the cases when accounts are eligible for more than one price list entry, she sets up condition weights to pick the “winner.”

Goal

- Create context rules for price list entries and child price lists
- Understand the TightestMatchInterface
- Use context dimension weights to determine which price is applied to a product

Tasks

1. Review the interface implementation
2. Assign the new accounts rule set to a price list entry
3. Create a rule to apply a child price list
4. Change the condition weight to make the new account pricing the tightest match

Time: 15 mins

How are context rules used for price list entries?

In the last exercise, you created rules to control which price lists display in the Cart. Now, you'll look at how you can apply rules to price list entries, so you can ensure that each customer sees the price that is right for them. You will create a new price list entry and assign a context rule to it. In doing so, you are entering into the world of multiple price list entries for a single product, and in order to determine which price to select, the `TightestMatchInterface` will be called.

What is an interface?

The core of our open architecture is the interface and implementation paradigm. An interface manages the application and its call out to business logic; an *implementation* is that business logic. Salesforce Industries offers multiple implementations for each interface. You can choose the implementation that best fits your needs or you can create your own implementation. When used judiciously, you can define interfaces once and deploy them many times to consistently maintain and apply business logic.

Salesforce Industries provides a variety of interfaces and implementations that support functionality across Salesforce Industries Communications, Media, and Energy & Utilities Clouds and change the default behavior for asset-based ordering, contracts, context rules, advanced rules, and pricing. For example, the `PricingInterface` communicates with the system to ask “how should I price this product?” The implementation responds with the appropriate business logic to evaluate the configuration, price lists, and rules and determine the correct price for the product.

As you know, a product can be assigned multiple prices, and pricing rules determine the prices available to the user. There can be many variables used to determine eligibility—geography, customer segment, promotional discounts—but what if the customer is eligible for multiple prices? In the end, a single price must be set. That's when the `TightestMatchInterface` comes into play.

What is the TighestMatchInterface?

TighestMatchInterface is provided as part of the managed package and is responsible for determining which price list entry should “win” in the event that the customer is eligible for multiple qualified price list entries. This interface includes two implementations in the managed package, the TighestMatchServiceImpl and the FirstMatchImplementation.

- TighestMatchServiceImpl uses condition weights in order to determine the “tightest match” price.
- FirstMatchImplementation, as its name implies, ignores those weights and selects the first match that it finds.

In this exercise, you will use the default implementation, TighestMatchServiceImpl.



NOTE:

The TighestMatchInterface and condition weights are used for context rules assigned to price list entries and child price lists. They are not used for context rules assigned to products or promotions.

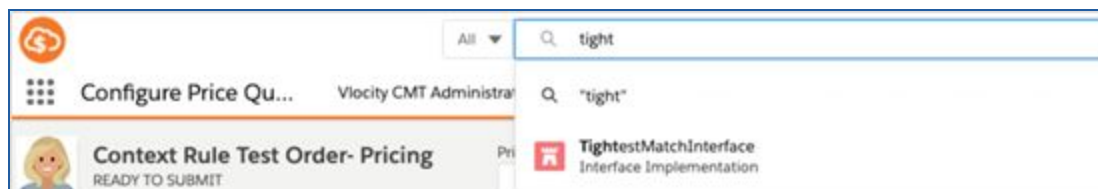
What are condition weights?

Condition weight is a general property of context dimensions, and it is a numeric value ranging from 0 to 60. This value is used by the TighestMatchServiceImpl to calculate the relative weight of a price list entry. It is only used for rules assigned to price list entries or child price lists and not for rules assigned to products and promotions.

- Values from 0-60 are supported.
- 0 or null values will default to a weight of 1.
- Any values greater than 60 will default to a weight of 1.

Task 1: Review the interface implementation

1. In the Lightning search dialog, enter `tight`.
2. In the search results list, click **TightestMatchInterface**.



3. Notice the two interface implementations listed, and that **TightestMatchServiceImplementation** is active.






NOTE:

Only one implementation can be active for each interface in an org.

Now you'll get busy creating a price list entry with a special price for new accounts. Context rules were designed to be reusable, so you can easily reuse the rule set and rule configuration for new accounts that you created in the first exercise. In the first exercise, you applied the rule set to a promotion, but in this exercise, you'll apply it to a price list entry.

Task 2: Assign the new accounts rule set to a price list entry

1. In the Lightning tab navigation bar, click **Vlocity Product Console**.
2. Under **Product Management** next to **Product**, click the **search icon**. 
3. In the **Search Product...** entry dialog, enter `iphone 8`, and click the **search icon**. 
4. Click the name of **Apple iPhone 8** or click the **Edit icon**  to open the product for editing.
5. In the left sidebar, click the **Pricing** facet.
6. Click **New**.
7. In the **PRICE LIST ENTRY General Properties** dialog on the right, enter the following information.

Field	Value
Price List	B2C Price List
Display Text	New account pricing

8. In the **PRICE LIST ENTRY Pricing Variable** dialog, enter the following information.

Field	Value
Charge Type	One-time
Sub-Type	Standard

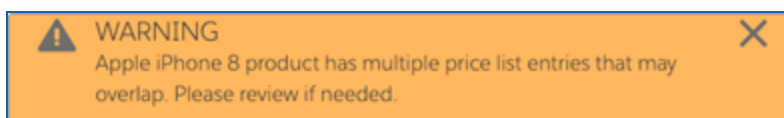
9. In the **PRICE LIST ENTRY Pricing Variable** dialog, click **Search**.
10. Under **Select a Pricing Variable**, click **One Time Std Price**.
11. Under the **Pricing Element** selection, click **\$549.99 OT**.

12. In the **PRICE LIST ENTRY Effectivity** dialog, enter the following information.

Field	Value
Effective From	[today's date]
Active	✓

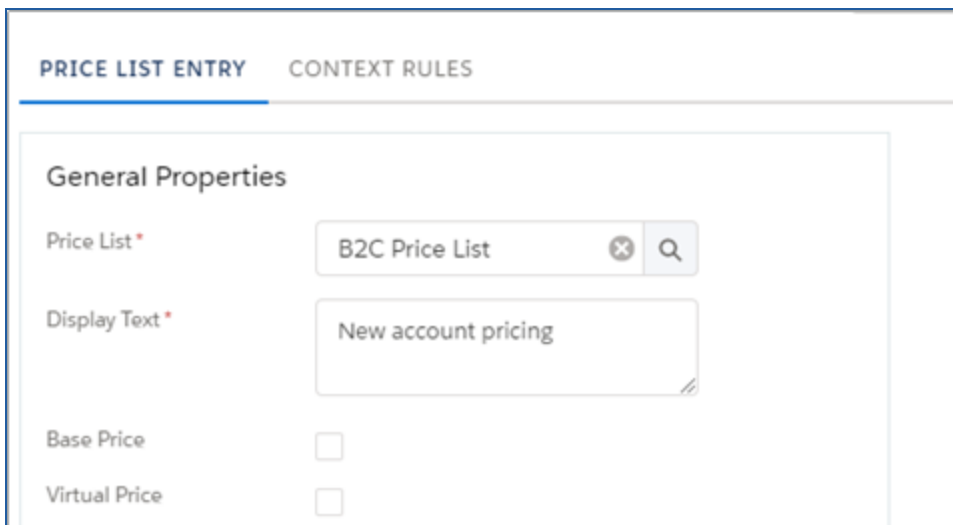
13. Click **Save**.

14. Notice the **WARNING** message, alerting you to the overlapping price list entries for the same price list. If it were an accidental overlap, this message would prompt the user to resolve the overlap. In this case, you can dismiss the message because you'll assign a rule to the new price list entry. Click **X** to dismiss.



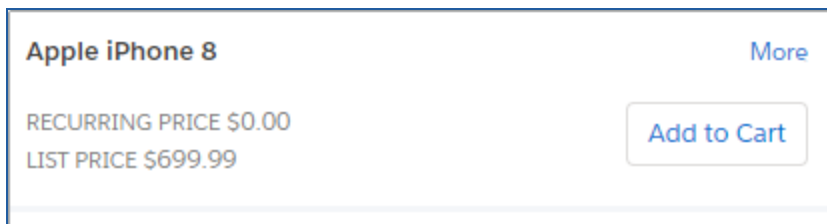
15. Click the edit icon  next to the **New account pricing** entry to open it for editing.

16. Scroll to the top of the price list entry dialog on the right, and click **CONTEXT RULES**.

A screenshot of the "PRICE LIST ENTRY" dialog. The "CONTEXT RULES" tab is selected. Under "General Properties", the "Price List" field is set to "B2C Price List" and the "Display Text" field is set to "New account pricing". There are checkboxes for "Base Price" and "Virtual Price", both of which are currently unchecked.

17. Under **Context Rules**, click **Add Rule Set**.

18. Using the **Add Rule Set** dialogs, select the **New Accounts Rule Set**.
19. In the **Add Rule Set** dialog, click **Save**.
20. Test the rule in the Cart.
 - a. In the Lightning tab navigation bar, click **Orders**.
 - b. Click the order number next to the **Context Rule Test Order - Pricing** order.
 - c. From the Power Launcher, click **Configure Order**.
 - d. Using the price list dropdown menu, select the **B2C Price List**.
 - e. In the product list, find the **Apple iPhone 8** product, and notice the list (base) price that displays.



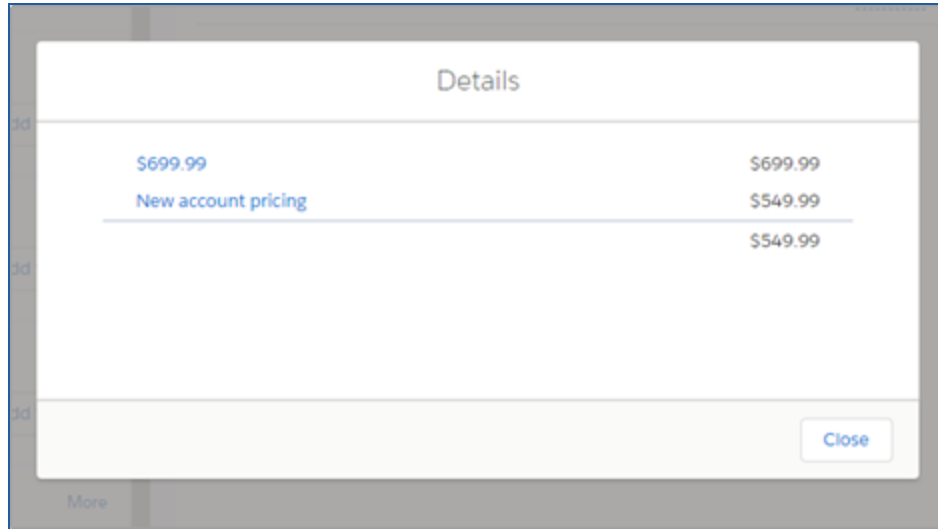
- f. Click **Add to Cart** to add the Apple iPhone 8 to the cart.
- g. Notice that the list price was overridden by the \$549.99 price list entry. Click the green **\$549.99** One Time Charge, and then click the **Price Details** icon.



Item Name	Quantity	Recurring Charge	Recurring Total	One Time Charge	One Time Total	Action	Promotions
Required attribute missing for Apple iPhone 8.							
Apple iPhone 8	1	\$0.00	\$0.00	\$699.99 <u>\$549.99</u>	\$549.99	Add	

- h. In the price **Details** window, notice that there are two prices listed, the base price of \$699.99 and the new account price of \$549.99. In this example, these two price list entries had an equal weight, and so both were returned. However, because the **TightestMatchService** implementation must choose a price, it chose the most recently created price list entry, which was \$549.99.

See the note below for more information on the weighting algorithm.

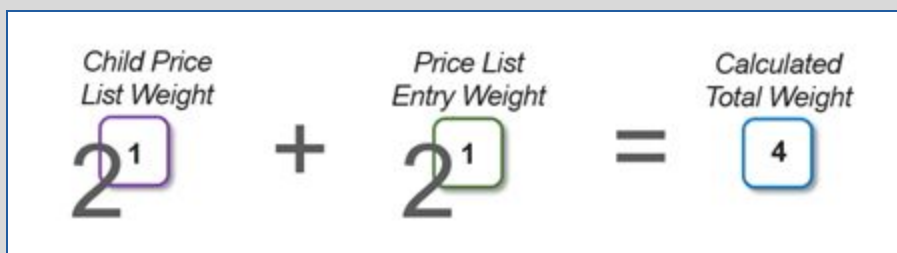


- i. Click **Close** to dismiss the price Details window.

NOTE:

Calculating Condition Weights

When a product is added to the Cart, the rules engine determines all of the qualified child price lists and price list entries based on the rules that have been configured. Then, it assigns a weight to each qualified price list entry, roughly using the formula above. The algorithm is using bitwise operators for maximum processing speed, so the weights are calculated as 2^n .



$$\begin{array}{l} \text{Child Price} \\ \text{List Weight} \end{array} \quad 2^1 \quad + \quad \begin{array}{l} \text{Price List} \\ \text{Entry Weight} \end{array} \quad 2^1 \quad = \quad \begin{array}{l} \text{Calculated} \\ \text{Total Weight} \end{array} \quad 4$$



The rules engine will sort all the price list entries by total weight, and then return the price list entries with the highest total weight. If there are multiple price list entries with the highest weight, the interface will select the most recently created price list entry and apply that price to the line item.

When multiple price list entries are equivalently weighted, the Cart's Price Details window will display all of the valid price list entries with display text.

In this example, you may be wondering how the weights for each of these price list entries were equivalent. In the first exercise, you created the AccountStatus context dimension, which is used in the New Account Rule Set, and you did not set a condition weight. And the list price of \$699.99 comes from the B2C Price List, which has the Consumer Accounts Rule Set applied to it, and its AccountType context dimension was also not given a condition weight.

So, it's easy to understand how these are equivalent, right?

Actually, it's a little more complex than $0=0$. The weighting algorithm discards context rules assigned to parent price lists, but then has a few tricks up its sleeve. It makes allowances for null values in the following way.

- PLE with no context rule = 1 or 2^0
- PLE with condition weight of 0 or null = 1 or 2^0

Given these allowances, the price list entries in this example are

B2C Price List Base Price \$699.99 = New account pricing \$549.99

(PLE with no context rule)

(PLE with null condition weight)

1 or 20

1 or 20

The algorithm makes a few other assumptions, namely

- PLEs with no context rules default to 1
- PLEs with a 0/null weight default to 1
- PLEs with values > 60 default to 1
- Weights for PLEs on Child Price List will be higher than a PLE on a Parent Price List
 - Parent Price Lists with a rule default to 0
 - PLEs on a child PL combine the PLE weight + PL weight

In the last task, you saw how to create a context rule for an individual price list entry. It was easy to reuse an existing rule, and then apply it to a price list entry for a product. However, what if you wanted to apply the rule to a hundred products? Or a thousand? The solution is to use a child price list as a container for the price list entries, then apply the context rule to the child price list, rather than each price list entry.

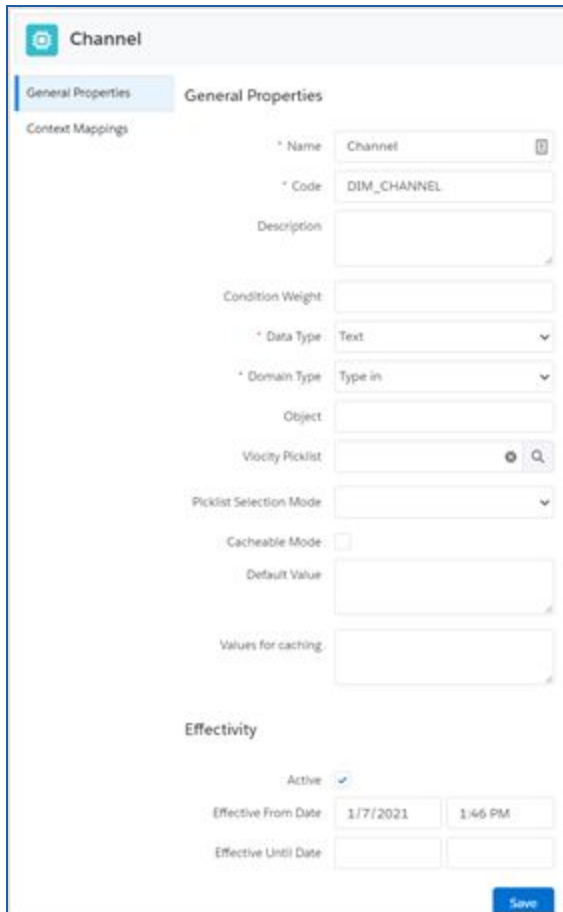
Now you'll create a new context rule for orders received over the web, and then assign the context rule to a Web Channel child price list.

Task 3: Create a rule to apply a child price list

1. In the Lightning tab navigation bar, click **Vlocity Product Console**.
2. Create a context dimension that will allow users to type in values for rule conditions.
 - a. Under Rules, click the plus sign (+) next to **Context Dimension**.
 - b. In the **New Context Dimension** dialog, enter the following information.

Field	Value
Name	Channel
Code	DIM_CHANNEL
Data Type	Text
Domain Type	Type in
Active	✓
Effective From Date	[today's date]

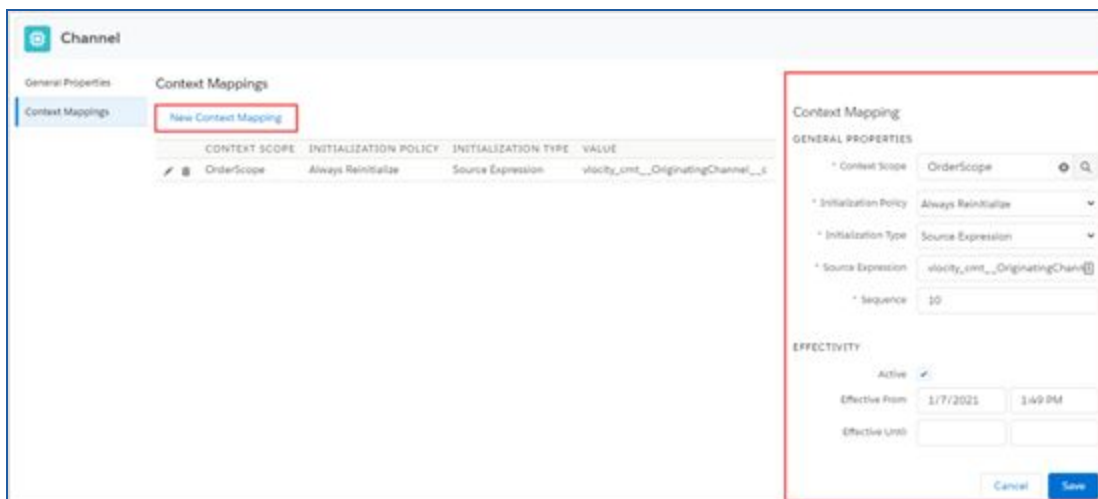
- c. Click **Save**.



3. Create a context mapping to the Originating Channel on the Order sObject.
- On the left sidebar, click the **Context Mappings** facet.
 - Click **New Context Mapping**.
 - In the **Context Mapping** dialog, enter the following information.

Field	Value	Notes
Context Scope	OrderScope	The field that you will map to is only available on the Order sObject.
Initialization Policy	Always Reinitialize	
Initialization Type	Source Expression	
Source Expression	velocity_cmt__OriginatingChannel__c	Make sure you include two underscores after “cmt” and “Channel.”
Sequence	10	
Active	✓	
Effective From	[today's date]	

d. Click **Save**.



The screenshot shows the Salesforce Channel configuration page. The 'Context Mappings' tab is active, displaying a table with one mapping. A 'New Context Mapping' button is highlighted. A modal window on the right shows the configuration for the selected mapping, including fields for Context Scope, Initialization Policy, Initialization Type, Source Expression, Sequence, and Effectivity.

CONTEXT SCOPE	INITIALIZATION POLICY	INITIALIZATION TYPE	VALUE
OrderScope	Always Reinitialize	Source Expression	velocity_cmt__OriginatingChannel__c

Context Mapping: GENERAL PROPERTIES

- Context Scope: OrderScope
- Initialization Policy: Always Reinitialize
- Initialization Type: Source Expression
- Source Expression: velocity_cmt__OriginatingChannel__c
- Sequence: 10

EFFECTIVITY

- Active: ☒
- Effective From: 1/7/2021 3:49 PM
- Effective Until:

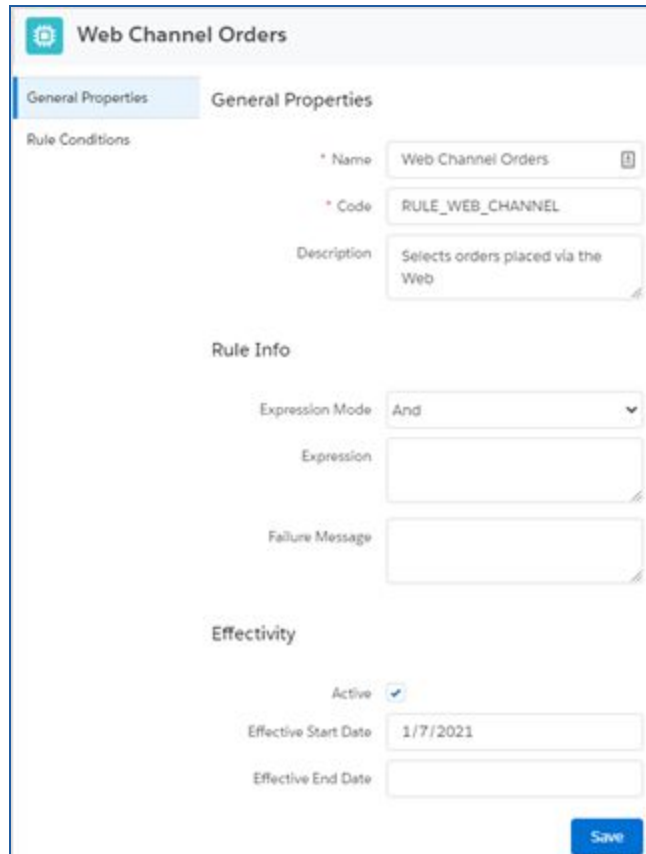
Buttons: Cancel, Save

e. In the **Product Console** tab navigation bar, click **X** to close the **Channel** tab.

4. Create a context rule for consumer accounts.
 - a. In the **Dashboard** under **Rules**, click the plus icon (+) next to **Rule** to create a new context rule.
 - b. In the **New Rule** dialog, enter the following information.

Field	Value
Name	Web Channel Orders
Code	RULE_WEB_CHANNEL
Description	Selects orders placed via the web
Expression Mode	And
Active	✓
Effective Start Date	[today's date]

- c. Click **Save**.



Web Channel Orders

General Properties

Rule Conditions

* Name: Web Channel Orders

* Code: RULE_WEB_CHANNEL

Description: Selects orders placed via the Web

Rule Info

Expression Mode: And

Expression:

Failure Message:

Effectivity

Active: ☒

Effective Start Date: 1/7/2021

Effective End Date:

Save

- d. In the left sidebar, click the **Rule Conditions** facet.
- e. Using the **Add Condition** dropdown menu, select **Simple**.

- f. In the **Rule Condition** window on the right, enter the following information.

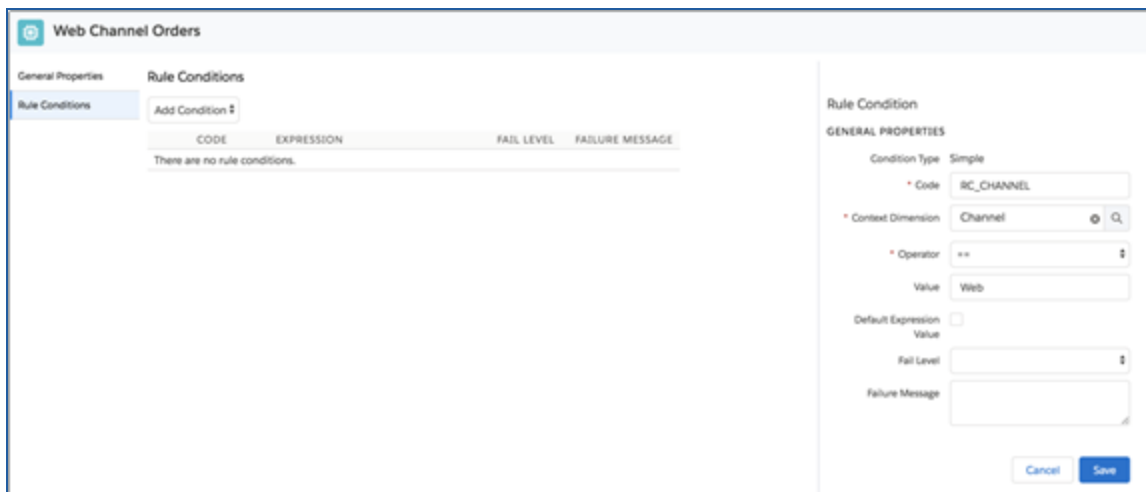
Field	Value	Notes
Code	RC_CHANNEL	
Context Dimension	Channel	This is the context dimension you created earlier in this task.
Operator	==	
Value	Web	Notice that this is a freeform text entry field due to the Domain Type that was set in the context dimension.

- g. Click **Save**



NOTE:

The Default Expression Value is used only with the digital commerce API Caching feature. It is not used with CPQPartition platform cache.

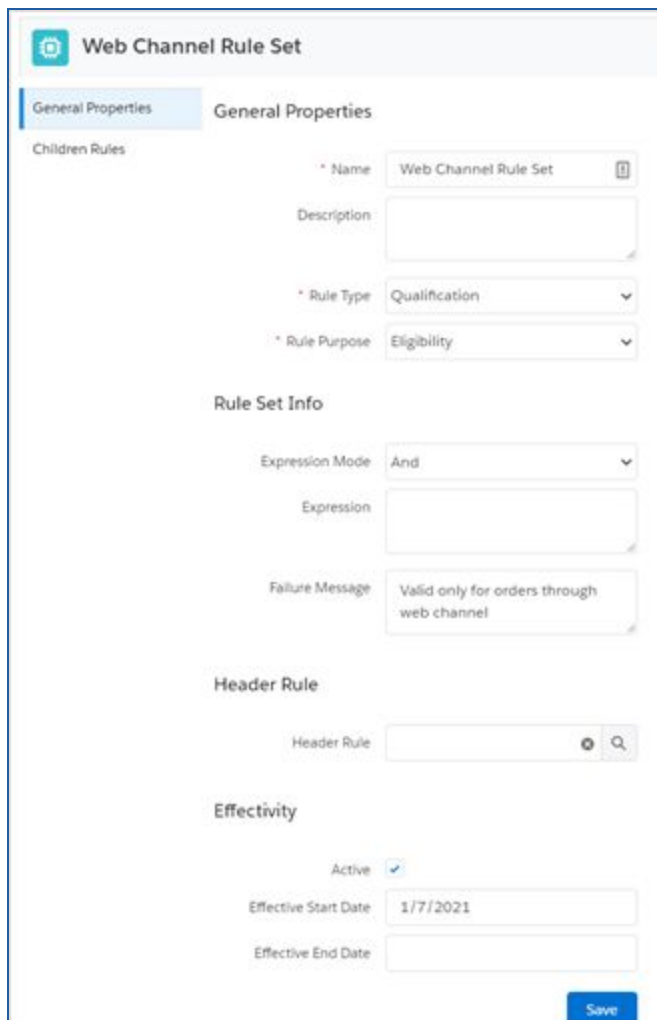


- h. In the **Product Console** tab navigation bar, click **X** to close the **Web Channel Orders** tab.

5. Create a rule set for web channel orders.
 - a. In the **Dashboard** under Rules, click the plus sign (+) next to **Rule Set** to add a new rule set.
 - b. In the **New Rule Set** dialog, enter the following information.

Field	Value
Name	Web Channel Rule Set
Rule Type	Qualification
Rule Purpose	Eligibility
Expression Mode	And
Failure Message	Valid only for web channel
Active	✓
Effective Start Date	[today's date]

c. Click **Save**.



Web Channel Rule Set

General Properties

Children Rules

* Name: Web Channel Rule Set

Description:

* Rule Type: Qualification

* Rule Purpose: Eligibility

Rule Set Info

Expression Mode: And

Expression:

Failure Message: Valid only for orders through web channel

Header Rule

Header Rule:

Effectivity

Active: ☒

Effective Start Date: 1/7/2021

Effective End Date:

Save

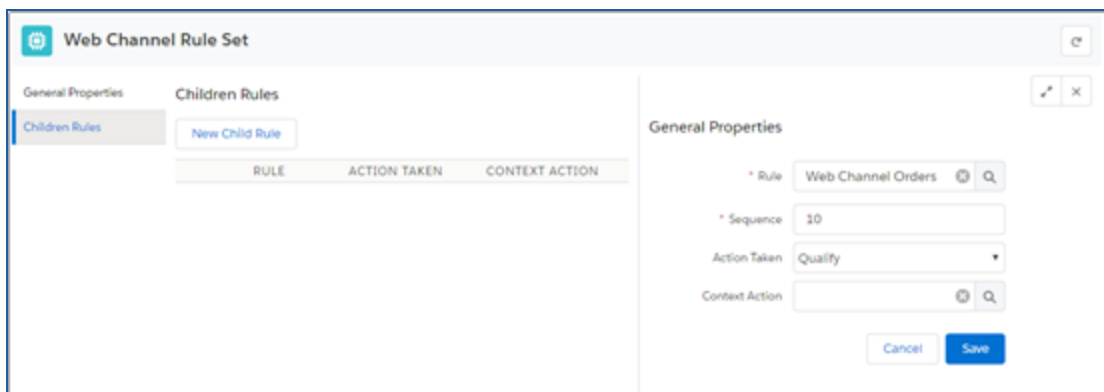
d. In the left sidebar, click the **Children Rules** facet.




e. Click **New Child Rule**.

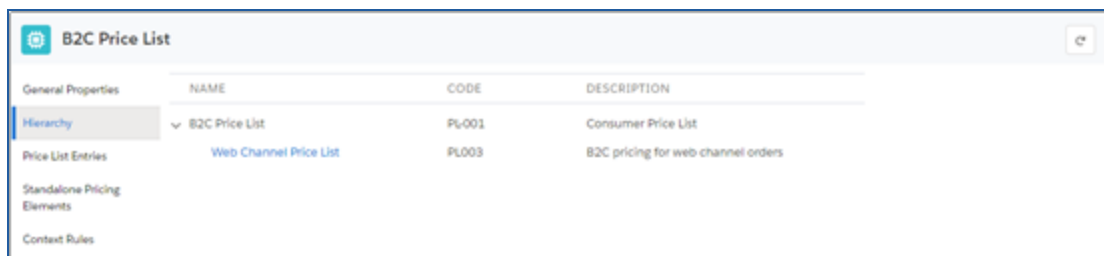
- f. In the **General Properties** dialog on the right, enter the following information.

Field	Value
Rule	Web Channel Orders
Sequence	10
Action Taken	Qualify

- g. Click **Save**.



- h. In the **Product Console** tab navigation bar, click **X** to close the **Web Channel Rule Set** tab.
6. Assign the new rule set to a child price list.
- In the **Console Dashboard** under **Pricing** next to **Price List**, click the **search icon**. 
 - Next to the **Search Price List...** entry dialog, click the **search icon**.  The list of all price lists is returned.
 - Click the name of **B2C Price List** or click the **Edit icon**  to open the B2C Price List for editing.
 - In the left sidebar, click the **Hierarchy** facet. Notice the child price list, **Web Channel Price List**.



- e. Click **Web Channel Price List**.
- f. In the left sidebar, click the **Price List Entries** facet. Notice the price list entries for the various Apple iPhone products.
- g. Click **New**.
- h. In the **General Properties** dialog, enter the following information.

Field	Value
Product	Apple iPhone 8
Display Text	Web channel pricing

- i. In the **PRICE LIST ENTRY Pricing Variable** dialog, enter the following information.

Field	Value
Charge Type	One-time
Sub-Type	Standard

- j. In the **PRICE LIST ENTRY Pricing Variable** dialog, click **Search**.
- k. Under **Select a Pricing Variable**, click **One Time Std Price**.
- l. In the **Pricing Element** section, click **\$499.99 OT**.



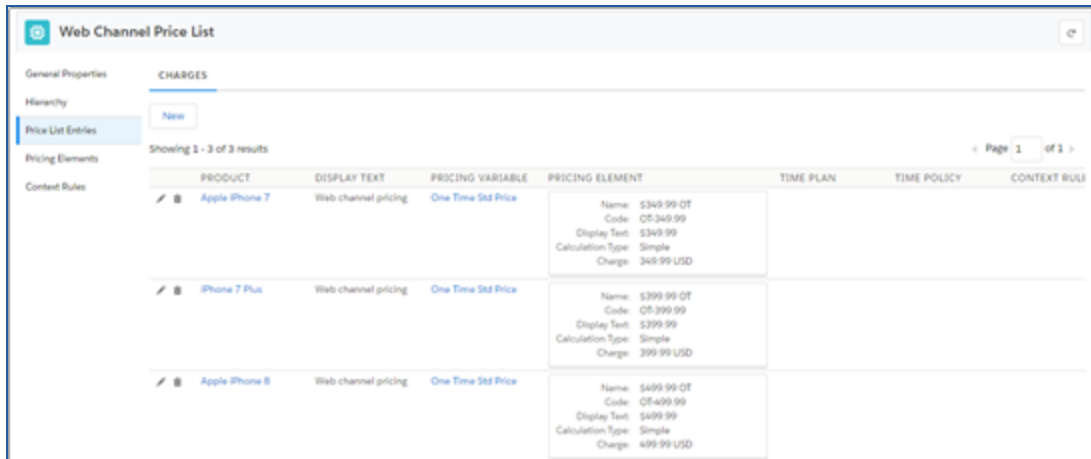
NOTE:

All of these pricing elements are inherited from the parent price list, B2C Price List.

- m. In the **PRICE LIST ENTRY Effectivity** dialog, enter the following information.

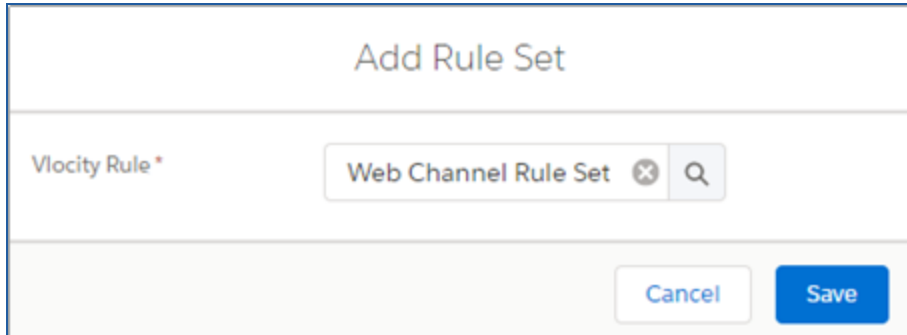
Field	Value
Effective From	[today's date]
Active	✓

n. Click **Save**.



PRODUCT	DISPLAY TEXT	PRICING VARIABLE	PRICING ELEMENT	TIME PLAN	TIME POLICY	CONTEXT RULE
Apple iPhone 7	Web channel pricing	One Time Std Price	Name: \$349.99-OT Code: 05-349.99 Display Text: \$349.99 Calculation Type: Simple Charge: 349.99 USD			
iPhone 7 Plus	Web channel pricing	One Time Std Price	Name: \$399.99-OT Code: 05-399.99 Display Text: \$399.99 Calculation Type: Simple Charge: 399.99 USD			
Apple iPhone 8	Web channel pricing	One Time Std Price	Name: \$499.99-OT Code: 05-499.99 Display Text: \$499.99 Calculation Type: Simple Charge: 499.99 USD			

- o. In the left sidebar, click the **Context Rules** facet.
- p. Under **Context Rules**, click **Add Rule Set**.
- q. In the **Add Rule Set** dialog, click the **search icon**.
- r. Using the **Vlocity Rule Lookup** dialog, enter `web` and click the **search icon**.
- s. Click **Select** next to **Web Channel Rule Set**.
- t. In the **Add Rule Set** dialog, click **Save**.



Now the Web Channel Rule Set will apply to all of the products in the Web Channel Price List child price list.

7. Review the new rule in the Cart.
 - a. In the Lightning navigation bar, click the **Orders** tab.
 - b. Click **New**.
 - c. In the **New Order** dialog, enter the following information.

Field	Value
Order Name	Context Rule Web Order - Pricing
Account Name	White, Carole
Order Start Date	[today's date]
Originating Channel	Web

- d. Click **Save**.
- e. From the Power Launcher, click **Configure Order**. This invokes the Cart.
- f. In the cart header using the price list dropdown, select **B2C Price List**.
- g. In the product list, find the **Apple iPhone 8**, and click **Add to Cart**.
- h. In the cart line items pane, notice the **One Time Charge** is \$499.99.

CART PROMOTIONS DISCOUNTS (0)							
Required attribute missing for Apple iPhone 8.							
AdvancedPricing • Loyalty							
	Quantity	Recurring Charge	Recurring Total	One Time Total	One Time Loyalty Price	One Time Loyalty Total	Action
Required attribute missing for Apple iPhone 8.							
Apple iPhone 8	1	\$0.00	\$0.00	\$499.99	\$499.99	0	Add

- Click the **Price Details** icon.
- Notice that the price list entry from the Web Channel Price List has been applied.

Details	
Web channel pricing	\$499.99
	\$499.99





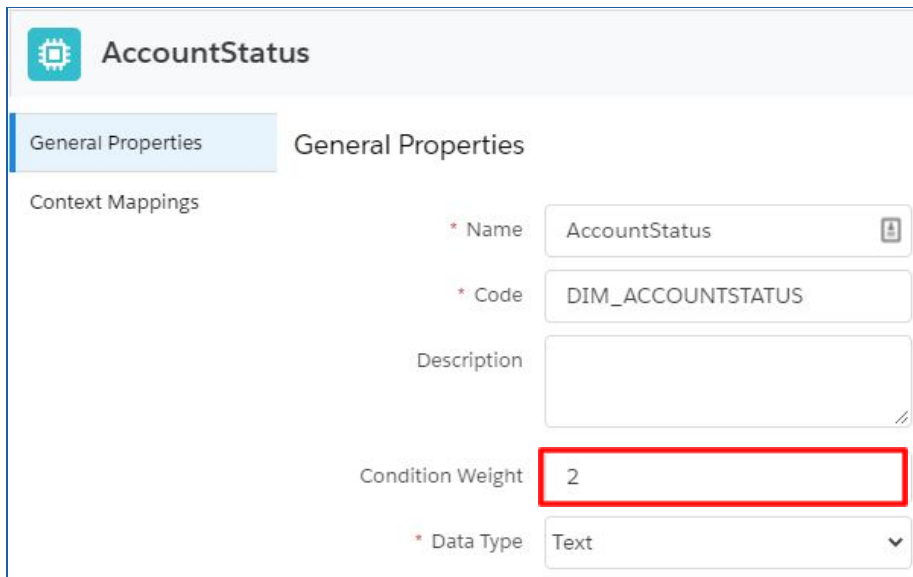
NOTE:

Why did **Web channel pricing** win tightest match when its condition weight was null, just like the condition weights on AccountStatus and AccountType? (Hint: consider the condition weighting formula.)

Although web channel pricing is very important, Eliza has decided that new account pricing should trump web channel pricing. You can make new account pricing “win” the tightest match by changing the condition weight on the AccountStatus context dimension.

Task 4: Change the condition weight to make the new account pricing the tightest match

1. In the Lightning tab navigation bar, click **Vlocity Product Console**.
2. In the **Dashboard** under **Rules**, click the **search icon**  next to **Context Dimension**.
3. In the **Search Context Dimension...** dialog, enter `account`, and then hit **Enter** or click the **search icon**.
4. Click the **edit icon**  next to the **AccountStatus**. This is the context dimension used in the New Accounts Rule Set.
5. Under **General Properties**, enter 2 for the **Condition Weight**, as shown below.



The screenshot shows the configuration page for the 'AccountStatus' context dimension. The 'General Properties' tab is active. The 'Condition Weight' field is highlighted with a red box and contains the value '2'. Other fields include 'Name' (AccountStatus), 'Code' (DIM_ACCOUNTSTATUS), 'Description' (empty), and 'Data Type' (Text).

6. Click **Save**.
7. In the Lightning tab navigation bar, click **Orders**.
8. Click the order number next to the **Context Rules Web Order - Pricing** order.
9. From the Power Launcher, click **Configure Order**.


10. In the product list, find the **Apple iPhone 8**, and click **Add to Cart**.
11. In the cart line items pane, notice that when the new product is added to the cart, the pricing rules are run again, and the updated tightest match price is applied.

[> Messages \(2\)](#)

Advanced Pricing ▼

Item Name	Quantity	Recurring Charge	Recurring Total	One Time Charge	One Time Total	Action	Promotions
Required attribute missing for Apple iPhone 8.							
Apple iPhone 8	1	\$0.00	\$0.00	<u>\$549.99</u>	\$549.99	Add	▼

Item Name	Quantity	Recurring Charge	Recurring Total	One Time Charge	One Time Total	Action	Promotions
Required attribute missing for Apple iPhone 8.							
Apple iPhone 8	1	\$0.00	\$0.00	\$549.99	\$549.99	Add	▼

12. Click the **\$549.99** One Time Price, then click the **Price Details** icon. 
13. Notice that the price list entry from the new account pricing has been applied.
14. Click **Close**.

Details

<u>New account pricing</u>	\$549.99
	\$549.99

Close

Troubleshooting Guide

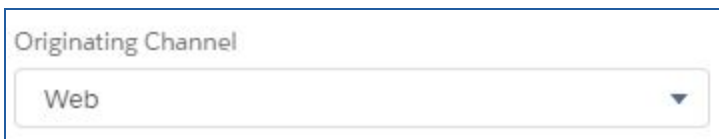
If your context rules are not displaying the desired behavior shown in this exercise, you can troubleshoot by checking the following settings, or by comparing your rule with the **SAMPLE** rule in your training playground.



NOTE:

Context rules, rule conditions and rule sets are cached in the org cache of the CPQPartition platform cache. When troubleshooting, you will need to clear the cache in order for changes to be reflected at run time. To clear the cache, select **Vlocity CMT Administration** from the Lightning toolbar, click on **Maintenance Jobs**, then click on **Start** next to **CLEAR MANAGED PACKAGE CACHE**.

- **Order – Originating Channel?** The **Originating Channel** field on your order should be set to **Web**.



Originating Channel

Web

- **Context Mapping - Source Expression?** The context mapping for the **Channel** context dimension should have a **Source Expression** with no spaces and multiple underscores, as shown below.




* Source Expression vlocity_cmt__Originating

- **Context Rule - Code?** The rule Code should have no spaces, as shown below.



* Code RULE_WEB_CHANNEL

- **Context Rule – Rule Condition Value?** Check to ensure that the rule condition value has no typos.

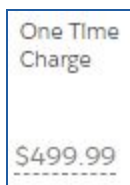


Value Web

- **Context Dimension – Condition Weights?** If you see prices that are crossed out like below, that means that there were multiple “winning” condition weights that were equivalent when evaluated by the TightestMatchInterface. That is the expected behavior at the end of Task 2. If you do not see that, check to make sure that your condition weight on the AccountStatus context dimension is null or 0.



When there is only one price listed, that is the tightest match, i.e., a single “winner.” At the end of Task 3, you should see a single “winner” one time charge of \$499.99 like below, and at the end of Task 4, you should see a one time charge of \$549.99. If not, check your context dimension condition weights.



Review

Confirm your understanding by answering these questions.

1. What is the `TightestMatchServiceImplementation`? What does it do?
2. Where can you set the condition weights used by the `TightestMatchServiceImplementation`?
3. Context rules that use condition weights can only be assigned to which objects?
4. What advantage does a child price list with a context rule have over a price list entry with context rule?

Exercise 6-5: Using Context Rules for Regional Product Launches Challenge

Scenario

Infiwave is jumping on the 5G bandwagon, and product marketing has decided to do a limited release of the new 5G data plan to consumers in high value regions. Your task is to create a new rule that restricts the 5G Data Plan to consumers in the West and Central regions of Canada.

Goal

- Challenge exercises solidify what you learned in the exercises, build your skills, and allow you to make mistakes -- which is how you will master this application.

Tasks

1. Determine where the regional data is stored
2. Determine which context scope you will use
3. Create a “Region” context dimensions and context mapping to the regional data
4. Create a “High Value Regions” context rule using the Region context dimension
5. Create a “High Value Consumer Regions” rule set
6. Assign the High Value Consumer Regions rule set to the 5G Data Plan product
7. Test the new rule in the Cart
8. Tune the rule set for best performance

Time: 45 mins

Task 1: Determine where the regional data is stored

Decide which object and field you want to evaluate in your rule. Remember the rule must evaluate consumers from West and Central regions of Canada. In your training playground, use Carole White as an example.

Task 2: Determine which context scope you will use

Should this rule be used in opportunities? quotes? orders? Which context scope will deliver the best performance for this use case?

Task 3: Create a “Region” context dimensions and context mapping to the regional data

When creating your context mapping, use the object and field that you selected in the first task.

Task 4: Create a “High Value Regions” context rule using the Region context dimension

Your context rule should qualify both West and Central regions. How can you do that using one context rule?

Task 5: Create a “High Value Consumer Regions” rule set

When you create your rule set, consider how to evaluate the region and the account type in one rule set. What can you re-use?

Task 6: Assign the High Value Consumer Regions rule set to the 5G Data Plan product

This is the easy part. Go for it!

Task 7: Test the new rule in the Cart

Using Carole White's account, test your new rule. Does the 5G Data Plan product appear in the list of qualified products? If not, review the troubleshooting guide in 6-2.

Task 8: Tune the rule set for best performance

What is one change you could make to your rule set to ensure best performance?

Hint: it's something in the Rule Set General Properties facet.



Yay! All done!