

Conditions

Screenshot_2020-04-29_at_19.1 Getting started

This article will teach you about conditions, their types, and how to configure them. Conditions are used to determine whether a customer is eligible to get a promotion.

Conditions can be found in the

Promotions > Conditions / list

At the **Conditions / list** window **Filters (1)** can be found which will help search for a specific type of conditions, and the list of **Conditions (2)** itself, which allows for searching conditions by name, editing, deleting, and adding new ones.

Screenshot_2022-09-14_154342.png

Visualization of the Promotions, Conditions / list

info2.svg **Remember**

Conditions and **Actions** are integral to **Promotion Rules**. All three can't be used individually. You can learn more about actions and rules in the [Actions](#) and [Rules](#) articles

Table of contents:

1. [Conditions / list](#)
 1. [Filters](#)
 2. [Conditions](#)
2. [Creating new condition](#)
3. [Condition types](#)

1. [Discount code in cart](#)
 2. [Discount code is empty in cart](#)
 3. [Order value](#)
 4. [Check if SKU\(s\) exist](#)
 5. [Active alt shop](#)
 6. [Sum of quantities from SKU list](#)
 7. [Prima port ID in cart](#)
 8. [Count of orders](#)
 9. [Days from last order](#)
 10. [Voucher in cart](#)
 11. [Empty gift of choice in the cart](#)
 12. [Delivery method](#)
 13. [Payment method](#)
 14. [Customer title](#)
 15. [Regular account](#)
 16. [Order items count](#)
 17. [Promotion rules active in cart](#)
-

Screenshot_2020-04-29_at_19.1

Conditions / list

Filters

Filters are the first tab in **Conditions / list**. It is used to narrow the results in the **Conditions** list.

(1) Is active? - Shows only active or inactive conditions.

(2) Type - Shows only active or inactive conditions.

(3) Two buttons:

- **Apply filters** - Applies chosen filters to **Conditions**

- **Clear filters** - Clears chosen filters and refreshes **Conditions**

Screenshot_2022-09-14_161027.png

Visualization of Filters, Conditions / list

Conditions

The conditions list is the main tab in this module. Here new conditions can be added **(1)**, below there is a search field **(2)** that will filter out conditions names to typed letters. The table list itself **(3)** contains several columns, these are:

- **ID** - The number given after the creation of the condition, it cannot be changed. Used only for database listing.
- **Type** - Type of condition that determines its behavior (more about types in this section [\[LINK\]](#))
- **Name** - Name of the condition, used as the main identification. Its name is used for rules and in the search bar.
- **Is active?** - Mark that shows whether a condition is active or inactive
- **Actions** - Four buttons:
 - **Edit** - Edit condition, its name, description, whether it's active, and parameters.
 - **Duplicate** - Duplicates condition with exact same parameters, the name will be edited by gaining (copy) as a suffix and changing its activation status to inactive.
 - **Show** - Shows details of the condition without permission to edit.
 - **Delete** - Erase condition, it won't be recoverable.

Screenshot_2022-09-14_163202.png

Visualization of Filters, Conditions / list

Screenshot_2020-04-29_at_19.1

Creating new condition

The condition has three fixed options that will stay the same **(1)**, these are:

- **Name** - Name of the condition, used as the main identification. Its name is used for rules and in the search bar.
- **Description** - Description serves as an additional source of information about the condition and its mechanics
- **Is active?** - ON/OFF - switch to turn the condition active or inactive.

There are also two **(2)** configurable options that determine its behavior, these are:

- **Type** - Determines the main target of verification. Types are:
 - **Discount code in cart (link do każdego, scrollujący do danego typu)**
 - **Discount code is empty in cart**
 - **Order value**
 - **Check if SKU(s) exist**
 - **Active alt shop**
 - **Sum of quantities from SKU list**
 - **Prima port ID in cart**
 - **Count of orders**
 - **Days from last order**
 - **Voucher in the cart**
 - **Empty gift of choice in the cart**
 - **Delivery method**
 - **Payment method**
 - **Customer title**
 - **Regular account**
 - **Order items count**
 - **Promotion rules active in cart**
- **Parameters** - JSON code used for parametrization of the condition.

Screenshot_2022-09-15_152600.png

Visualization of Filters, Conditions / list

info2.svg **Remember**

JSON is a light-weighted format to exchange data. More about this format you can learn from [here](#).

Screenshot_2020-04-29_at_19.1

Condition types

The type of condition defines what it's supposed to look for when calculating cart items and values. Here are described all types of conditions and their parameters

Discount code in cart

This condition checks if there is a discount code in the cart (more about discount codes can be found in the [Discount codes](#) article).

Parameters:

- **None**

Logic operators:

- =
- !=

Logic values:

- **Discount code name**

info2.svg **Remember**

Since this condition type is so general and lacks any parameters. There is usually one of this type in the environment.

Discount code is empty in cart

This condition checks if there isn't a discount code in the cart.

Parameters:

- **Discount code name** - it is not used in the condition itself since adding this type of condition to the rule will require naming it.

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

info2.svg **Remember**

Since this condition type is so general and lacks any parameters. There is usually one of this type in the environment.

Order value

This condition checks the current cart value.

Parameters:

- **prices:**
 - **promotion** - used by default, counts the value of products by promotion prices
 - **catalog** - counts values of products by their standard prices
- **consider_discounts:**
 - **FALSE** - used by default, counts the value before all discounts
 - **TRUE** - calculate order value after all discounts
- **use_default_excluded_list:**
 - **TRUE** - used by default, uses global excluded SKU list
 - **FALSE** - excluded SKUs will also be counted toward the cart value
- **include_payed_gifts:**
 - **TRUE** - consider the value of payed gifts
 - **FALSE** - used by default, do not count payed gifts
- **include_opensets:**
 - **TRUE** - opensets count into the cart value
 - **FALSE** - used by default, do not count value of opensets
- **exclude_products:**
 - **Array** - SKU numbers, these will be excluded from the order value. This parameter is above and will exclude products regardless of **use_default_excluded_list**

parameter.

- **exclude_fid_discounted_products:**
 - **TRUE** - Products discounted by the fidelity program will not be counted.
 - **FALSE** - used by default, do not count value of products discounted by fidelity points

Logic operators:

- =
- !=
- >
- <
- >=
- <=

Logic values:

- **Number** - cart value

Example:

- {"prices":"catalog","consider_discounts":"TRUE","exclude_products":["12345","54321"]}

Check if SKU(s) exist

This condition checks if the product SKUs from the selected array are present in the cart.

Parameters:

- **SKU:**
 - **Array** - SKU numbers, without this parameter the condition won't work.
- **required_quantity:**
 - **all** - All SKUs from the **SKU** parameter have to be present in the cart
 - **number** - Number of SKUs from the SKU parameter that has to be present.
- **consider_quantity:**
 - **TRUE** - The number of pieces counts towards the **SKU** parameter (for example if the **SKU** parameter has 3 SKU's in the list then one of these SKU's added 3 times to the cart will return **TRUE** value in the promotion rule)
 - **FALSE** - Used by default, doesn't count the number of pieces

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Example:

- {"SKU":["12345","54321"],"required_quantity":"5"}

Active alt shop

This condition checks the altshop. This type is deprecated and unused. Promotion rules already are able to filter altshops.

Sum of quantities from SKU list

This type sums up all pieces from the SKU list set in the parameters.

Parameters:

- **SKU:**
 - **Array** - SKU numbers, without this parameter the condition won't work.

Logic operators:

- =
- !=
- >
- <
- >=
- <=

Logic values:

- **Number** - Quantity of products

Example:

- {"SKU":["54321","12345"]}

Prima port ID in cart

This type of condition is no longer supported since Prima port is no longer used.

Count of orders

This type count the number of orders in the customer account.

Parameters:

- **orders:**
 - **all** - All orders from the customer account will be counted (still regulated by the rest of the parameters)
 - **all_mc** - All orders are counted but since the date of gaining a title. You can learn more about customer titles in the [Campaigns and titles](#) article.
 - **current_title** - Orders are counted since the date of gaining the current customer title.
 - **active_rule** - Orders are counted only if they used certain promotion rules (list from parameter **rule id**)
 - **campaigns:**
 - **Array** - List of campaigns short names (more about campaigns in this article **[UNDER CONSTRUCTION]**)
 - **date_from:**
 - **date** - The format for the date is YYYY-MM-DD (for example 2018-12-31). Orders will be counted from this date onwards.
 - **date_to:**
 - **date** - The format for the date is YYYY-MM-DD (for example 2018-12-31). Counted will be orders placed only before this date.
 - **titles:**
 - **Array** - List of title short names. Orders will be counted only if they were placed while the customer had these titles. Only used when parameter **"orders":"current_title"**
 - **rule_id:**
 - **Array** - List of **Rules** id's. This parameter only works with **"orders":"active_rule"**
 - **only_paid:**
 - **TRUE** - Only COD or fully paid orders are counted.
 - **FALSE** - Orders with pending payment are also counted
 - **altshop_id:**
 - id of altshop only from which orders are considered. The parameter value can only be one altshop/one ID. You can't use an array of altshop ID's.
- Syntax example:**
- {"altshop_id":"122"}

Logic operators:

- =
- !=
- >

- <
- >=
- <=

Logic values:

- **Number** - Orders

Example:

- {"orders":"all_mc","campaigns":["Campaign1","Campaign2"],"date_from":"2023-01-12","date_to":"2023-02-22"}

Days from last order

This condition count passed days from the last order.

Parameters:

- **orders:**
 - **all** - All orders from the customer account will be counted (still regulated by the rest of the parameters)
 - **all_mc** - All orders are counted but since the date of gaining a title. You can learn more about customer titles in the [Campaigns and titles](#) article.
 - **current_title** - Orders are counted since the date of gaining the current customer title.
 - **active_rule** - Orders are counted only if they used certain promotion rules (list from parameter **rule id**)
- **campaigns:**
 - **Array** - List of campaigns short names (more about campaigns in this article **[UNDER CONSTRUCTION]**)
- **date_from:**
 - **date** - The format for the date is YYYY-MM-DD (for example 2018-12-31). Orders will be counted from this date onwards.
- **date_to:**
 - **date** - The format for the date is YYYY-MM-DD (for example 2018-12-31). Counted will be orders placed only before this date.

Logic operators:

- =
- !=
- >
- <
- >=
- <=

Logic values:

- **Number** - Orders

Example:

- {"orders":"all_mc","campaigns":["Campaign1","Campaign2"],"date_from":"2023-01-12","date_to":"2023-02-22"}

caution.svg **Warning**

If there are no order-matching rules defined by parameters, the condition will return 999 days. We strongly recommend pairing this condition with "Count of closed orders > 0"

Voucher in the cart

This condition verifies whether there is applied promotion code in the cart.

Parameters:

- **type:**
 - **1** - Fiden2 code, this type is set as default.
 - **2** - Promotion code from **Promotions > Discount codes / list**. You can learn more about discount codes in the [Discount codes](#) article.

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Example:

- {"type":"2"}

Empty gift of choice in cart

This condition will check if the customer chooses a certain, empty gift from the **Gift of choice** action type. You can learn more about actions in the [Actions](#) article.

Parameters:

- **rule_id:**
 - **1** - Fiden2 code, this type is set as default.
 - **2** - Promotion code from **Promotions > Discount codes / list** (More about codes in this article[**UNDER CONSTRUCTION**])

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Example:

- {"rule_id":"2"}

Delivery method

This type checks chosen delivery method.

Parameters:

- **name:**
 - **Array** - List of delivery method names.

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Example:

- {"name":["Delivery1","Delivery2"]}

Payment method

This type checks chosen payment method.

Parameters:

- **name:**
 - **Array** - List of delivery method names.

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Example:

- {"name":["Payment1","Payment2"]}

Customer title

This type checks current customer title.

Parameters:

- **include_title:**
 - **Array** - List of **shortcut title names** that will count into the **TRUE** value during calculation.
- **exclude_title:**
 - **Array** - List of **shortcut title names** that will count into the **FALSE** value during calculation.

Logic operators:

- =
- !=
-

Logic values:

- **TRUE**
- **FALSE**

Example:

- {"include_title":["T0","TS"]}

Regular account

This type checks if a customer is logged in to a registered account (not one-time).

Parameters:

- **None**

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Orders item count

This type checks the number of items in the cart.

Parameters:

- **consider_quantity:**
 - **TRUE** - Set by default, every piece in each row will be counted.
 - **FALSE** - Only unique SKUs will be counted.
- **consider_gifts:**
 - **TRUE** - Gifts will be counted to this condition.
 - **FALSE** - Set by default, gifts won't be counted to this condition.
- **consider_sets:**
 - **TRUE** - All sets will be counted.
 - **FALSE** - Set by default, sets won't be counted.
- **consider_outlet:**
 - **TRUE** - Discounted products with the flag outlet will count toward this condition.
 - **FALSE** - Set by default, products with the flag outlet won't be counted.
- **consider_greenpoint:**
 - **TRUE** - Products with the flag greenpoint will count toward this condition.
 - **FALSE** - Set by default, products with the flag greenpoint won't be counted.

Logic operators:

- =
- !=
- >
- <
- >=

- <=

Logic values:

- **Number** - Products in a cart

Example:

- {"consider_quantity":"FALSE","consider_greenpoint":"TRUE"}

Promotion rules active in cart

This type checks if a set of rules is active in a cart. A rule is considered an active one when its conditions are met. A rule with no conditions is always considered as active. It is important to remember to set a sort order in a rule with this condition to a high value, so it is read last by the promotion module.

Parameters:

- **rule_ids:**
 - **Array**- list of rules' IDs that are checked for activity. Set by default, every piece in each row will be counted.
- **mode (Default "all"):**
 - **"all"** means *all of the rules must be active*,
 - **"any"** means *any of the rules must be active*.

Logic operators:

- =
- !=

Logic values:

- **TRUE**
- **FALSE**

Examples:

- **Rule 123 is active:**
 - {"rule_ids": [123]}
- **All the rules 123 and 321 are active:**
 - {"rule_ids": [123, 321]}
- **Any of the rules 123 and 321 is active:**
 - {"rule_ids": [123, 321], "mode": "any"}

Revision #2

Created 27 November 2024 21:06:21 by Tau Ceti

Updated 27 November 2024 21:28:23 by Tau Ceti