

Examples of popular promotions

Screenshot_2020-04-29_at_19.1

Getting started

In this article, you will find an example of the promotions that are used most frequently on multiple Yves Rocher websites. This article will only show you possible combinations of conditions and actions without the details, of how to create such promotions. You can find articles about condition, action, and rule creation with all their possibilities in the following articles:

- [Promotion module introduction](#)
- [Conditions](#)
- [Actions](#)
- [Rules](#)

Table of contents:

1. [Gift of choice after set cart value is exceeded](#)
2. [Discount on 2 most expensive products when discount code is added to the cart](#)
3. [Free delivery over set cart value](#)
4. [Two products for one when selected products are in the cart](#)
5. [Third product from the list for free when selected discount code is used](#)
6. [Additional FIDEN2 loyalty points when customer exceeds set cart value](#)

Screenshot_2020-04-29_at_19.1

Gift of choice after set cart value

is exceeded

Gift of choice is a list of gifts that customers can choose from. In this example, the customer will be able to do that after his cart value is over 290Kc

To create such a promotion, you have to create an **Order Value** condition. In this example, we used the following JSON:

```
{"prices": "promotion", "include_paid_gifts": "TRUE", "use_default_excluded_list": "FALSE", "consider
```

Additionally, you have to create a **Gift of choice** action with example JSON:

```
{"sku_list": ["F31623", "F94615", "F01989", "250PT", "NOGIFT"], "min_cart_value_for_description": "290
```

For a promotion to work, you have to assign the above condition and action to the rule.

POWERPNT_Dvh4jgJLQ4.png

Visualization of the rule configuration of the promotion.

For an **Order value** condition to activate above 290Kc, we have to set the logic operator to **>= (1)** and the value to **290 (2)**. The action doesn't require any operator, or value, as it doesn't use any parameters except the ones written in the JSON.

Below, you can see how a customer sees the **Gift of choice (3)** promotion, whenever he has more than 290Kc of products in his cart.

POWERPNT_jjERXWkywh.png

Visualization of the gift of choice promotion on the website

Screenshot_2020-04-29_at_19.1

Discount on 2 most expensive

products when discount code is added to the cart

One of the most common promotion types uses discount codes as a condition with various actions. In this case, when the customer types the discount code to the cart, his two most expensive products will get discounted by -40%.

Such promotion uses the **Discount code in cart** condition. This condition doesn't use any parameters.

To discount the most expensive products in the cart, we have to create a **discount most expensive product** action. In this example, we will use the following JSON parameters:

```
{"type": "percentage", "prices": "catalog", "exclude_products": ["green_point_flag"], "quantity_limit": 2}
```

For a promotion to work, you have to assign the above condition and action to the rule.

POWERPNT_fMbHqUcB6q.png

Visualization of the rule configuration of the promotion.

For a **Discount code in cart** condition to activate, we have to set the logic operator to = **(1)** and value to desired promotion code **(2)** - in this case, we will use **MAILING11_22points** one. In the action field, we have to specify the percentage discount customers will receive on the two most expensive products, as the type parameter in the action is set to percentage. In this example, we set it to **40 (3)**.

Below you can see how the customer can see this discount in his cart **(4)**.

POWERPNT_PXCfy2CRNd.png

Visualization of the applied promotion

Screenshot_2020-04-29_at_19.1

Free delivery over set cart value

One of the most common promotions in the e-commerce platforms is free delivery. In TauCeti panel, we can set it up in two different ways:

1. Changing the price range for set delivery methods in **Altshops > delivery costs / list**
2. Creating a promotion with a free delivery action

The first method can be read about in the [Delivery costs / list - browsing and managing delivery costs](#) article.

The second method requires creating a rule that contains a selected condition and a free delivery rule. In this example, we will use an **order value** condition with JSON parameters:

```
{"prices":"promotion","consider_discounts":"TRUE","exclude_products":["green_point_flag"]}
```

For an action, we have to create a new action with a **Free delivery** type. This action type doesn't require any JSON parameters.

For a promotion to work, you have to assign the above condition and action to the rule.

POWERPNT_hAEPiQ9ZXV.png

Visualization of the rule configuration of the promotion.

For an **Order value** condition to activate above, for example, 499Kc, we have to set the logic operator to **>= (1)** and value to **499 (2)**. **Free delivery** action doesn't require any operator, or value, as it doesn't use any parameters except the ones written in the JSON.

Below, you can see how customers can see this discount in their cart **(3)**.

POWERPNT_7wq6ddVKJ8.png

Visualization of the applied promotion

Screenshot_2020-04-29_at_19.1

Two products for one when selected products are in the cart

One of the most often promotions used are two for one promotions. Such promotion can be easily configured using the **two products for one** action.

In this example, the promotion will work only when selected products are in the cart. For this, we need to create a condition with **check if SKU(s) exist** condition type. We are using the following JSON:

```
{"sku":["55794","55665","55692","55751","55724","81503"],"required_quantity":"2","consider_quant
```

For an action, we have to create a **two products for one action** with the following JSON:

```
{"allowed_products":["55794","55665","55692","55751","55724","81503"]}
```

In order for a promotion to work, you have to assign the above condition and action to the rule.

POWERPNT_nOBV4peV6a.png

Visualization of the rule configuration of the promotion.

For a **check if SKU(s) exist** condition, we have to set the logic operator to = **(1)** and value to **TRUE (2)**. **Two products for one** action doesn't require any operator, or value, as it doesn't use any parameters except the ones written in the JSON.

Below, you can see how customers can see this discount in their cart **(4)**.

POWERPNT_Pvaq7DBzMM.png

Visualization of the applied promotion

Screenshot_2020-04-29_at_19.1

Third product from the list for free when selected discount code is used

This promotion allows customers to select three products from the list and one of them (usually the cheapest one) will be for free. In order to activate such promotion customers will be required to apply a discount code in the cart.

For conditions, we need a simple **Discount code in cart** condition. This condition doesn't use any parameters.

In order for promotion to apply 3rd product for free to a certain list of products we need a **Set of products** action. Set of products can be used in various different ways according to their parameters. In the case of this promotion, we need the action with the following JSON parameters:

```
{"allow_same_sku": "TRUE", "allowed_products": ["55794", "55665", "55692", "55751", "55724", "81503"], "s
```

In order for a promotion to work, you have to assign the above condition and action to the rule.

POWERPNT_UisQ8C2ZoY.png

Visualization of the rule configuration of the promotion.

For a **Discount code in cart** condition to activate, we have to set the logic operator to = **(1)** and value to desired promotion code **(2)** - in this case, we will use **Discountcode** one. In the action field, we only have to specify a newly created action, as it doesn't require additional values.

POWERPNT_tljaB3mPri.png

Visualization of the applied promotion

Screenshot_2020-04-29_at_19.1

Additional FIDEN2 loyalty points when customer exceeds set cart value

Sometimes it is a good idea to reward customers with more FIDEN2 points whenever they spend more money on their order. In this case, it is possible to use a promotion module to automatically apply such promotion.

For this type of promotion, we require a condition that will activate an action whenever the cart value exceeds the set price point. For this, we can use **Order value** condition with the following JSON parameters:

```
{"prices": "promotion", "consider_discounts": "TRUE", "exclude_products": ["green_point_flag"]}
```

In the above case promotion price of products and other promotional discounts will be counted.

For an Action, we have to use **Add or multiply loyalty points** action. This action requires to set if we want to add or multiply by percentage the loyalty points, so we can do that like in the example below:

```
{"type": "amount"}
```

In this case, we will add a set amount of additional FIDEN2 points to the customer.

For a promotion to work, you have to assign the above condition and action to the rule.

POWERPNT_zdCPCkg3aD.png

Visualization of the rule configuration of the promotion.

For an **Order value** condition to activate above, for example 499Kc, we have to set the logic operator to **>= (1)** and value to **499 (2)**. **Add or multiply loyalty points** action requires to specify the amount of points we want to add to the order. In our case as the action is set to the amount of points if we type for example **500** points, the customer will receive additional 500 FIDEN2 points with their order.

Revision #2

Created 27 November 2024 21:06:18 by Tau Ceti

Updated 27 November 2024 21:26:26 by Tau Ceti