

The Kentico E-commerce Solution allows you to define coupon-based **gift cards**. If you create a gift card and generate coupon codes, your [customers](#) can use the codes to reduce or completely cover the final price of their orders. In other words, gift cards can be considered as a form of payment.

When calculating the final price of a shopping cart or order, the system subtracts the value of applied gift cards **after** processing all other types of [discounts](#), [shipping costs](#) and [taxes](#). If the gift card value is greater than the order's price, the remaining amount is saved for the given [coupon code](#) and can be redeemed in future orders.

You can use gift cards to motivate customers to purchase more [products](#). For example, gift cards are particularly suitable:

- During special seasons, for example Christmas
- As a reward for customers who spend a certain amount of money in your store or buy selected products

Gift cards are dynamically configurable, and you can adjust them via rules and conditions according to your needs. These rules are based on properties of the customer's shopping cart and can include the cart content, billing/shipping address, order currency, etc. You can also specify periods of validity.




Limitations

- Kentico does not provide a built-in way to directly sell gift cards as [products](#).
- If you run [multiple sites](#), each site has its own set of gift cards. Therefore, if you want to use the same gift cards across multiple sites, you need to create them on all of the given sites.

Managing gift cards

To create new gift cards or edit existing ones

1. [Open](#) the **Gift cards** application.
2. To add a new gift card, click **New Gift card**. To edit an existing gift card, click **Edit** () next to the given item in the list.
3. Set the gift card's properties:

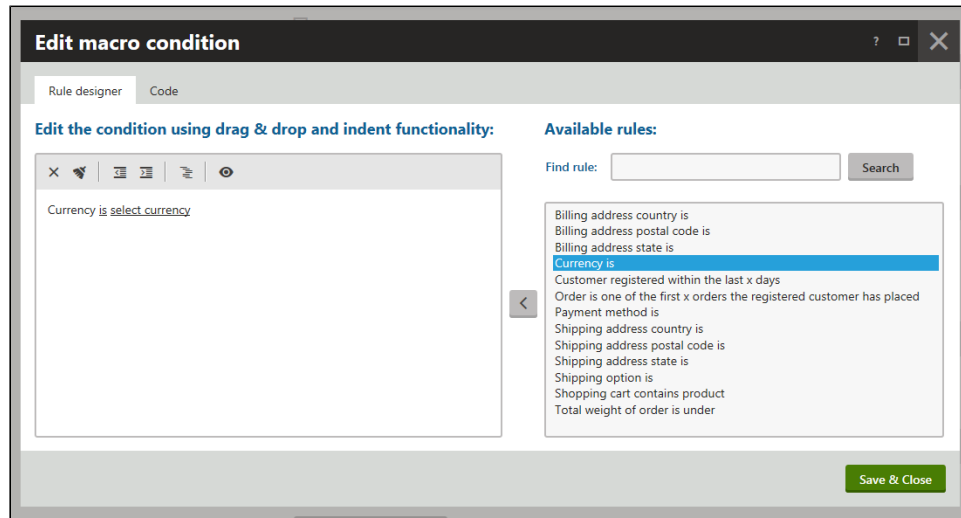
General	
Name	The name of the gift card displayed in the administration interface, shopping cart, invoices, etc.
Code name	The unique identifier of the gift card (for example used by developers in custom code). Unless there is a reason to set a particular value, you can leave the default (<i>automatic</i>) option, and the system generates an appropriate code name automatically.
Description	You can add an explanation about the scenarios for which the gift card is intended. The description does not affect the gift card's functionality, apart from providing information to store managers working in the <i>Gift cards</i> application.
Enabled	Indicates whether the gift card is available for use by customers. If the check box is cleared, the gift card cannot be redeemed.
Gift card value	

Amount	<p>Specifies the exact value of the gift card. The amount must be a valid decimal number greater than 0. Enter the value in the site's main currency.</p> <p>When a customer redeems one of the gift card's coupon codes during checkout, the system subtracts the amount from the total price of the order. If the gift card amount is greater than the order's price, the remainder is stored for the given coupon code, and it can be redeemed in future orders (with a reduced value equal to the remaining amount).</p> <p>Note: You cannot change the <i>Amount</i> value for gift cards that already have coupon codes added on the Coupons tab. In these cases, you need to create a completely new gift card.</p>
Gift card conditions	
Minimum order amount	<p>Specifies the minimum price total that orders must have for the gift card to be applicable. The price is evaluated after the system calculates all discounts and adds shipping costs and taxes to the order total.</p> <p>Set the value in the site's main currency.</p> <p>If you leave an empty value, all orders are eligible for the gift card unless limited by the Further conditions field.</p>

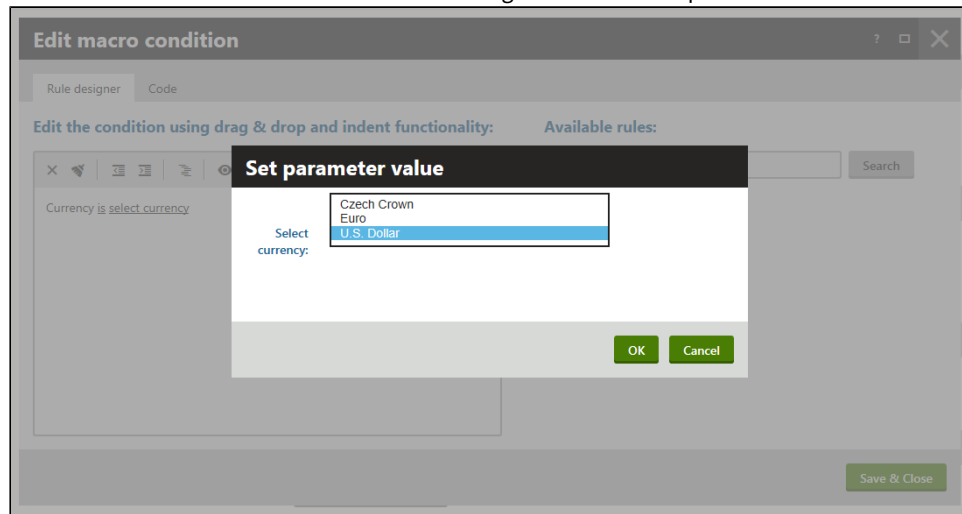
Further conditions

Adds further conditions that must be fulfilled for the system to accept the gift card.

Click **Edit** to open the **Edit macro condition** dialog, which allows you to add conditions through a graphical interface.



- To add rules to the condition, double-click rules on the right or click **Add rule** (↵).
- To remove a rule from the condition, select it and click **Delete** (X). Click **Clear all rules** (🗑️) to remove all added rules.
- Click on underlined words in the rule text to configure various rule options:



If you are a developer, conditions are [Kentico macros](#), and you can edit them directly on the **Code** tab in the **Edit macro condition** dialog.

Duration


Valid from

Specifies the time and date from which the gift card can be redeemed. If you leave the value empty, the gift card is valid at any time before the **Valid to** date.

Valid to

Specifies the time and date until which the gift card can be redeemed. If you leave the value empty, the gift card is valid permanently (starting from the **Valid from** date).

Target customers

Available for	<p>Determines which types of users are able to use the gift card:</p> <ul style="list-style-type: none"> • All visitors – all customers can use the gift card. • Registered users – only registered customers can use the gift card. • Registered users in selected roles – only registered customers in selected roles can use the gift card. To add roles: <ul style="list-style-type: none"> a. Click Select to open the Select roles dialog. b. Select the required roles. c. Click Select. <div style="background-color: #d4edda; padding: 10px; margin-top: 10px;">  You can add registered customers to roles in the Customers application while editing customers on the Roles tab. </div>
---------------	--

4. Click **Save**.

5. Create or generate coupon codes on the **Coupons** tab. For more information, see: [Working with coupon codes](#)

If enabled and valid, the gift card can be redeemed by customers during checkout or applied by store administrators when editing orders. You can now distribute the coupon codes among your customers via any available communication channels.

For gift cards, every coupon code represents a single voucher with a value equal to the gift card's *Amount*. When redeemed, the coupon code is either fully used up, or if the gift card value is greater than the order's price, the remainder is saved for the given coupon code. While the remaining value is greater than zero, the coupon code can be redeemed again in other orders (with a reduced value).

Processing of multiple gift cards

If a customer adds multiple gift card coupon codes to a single order, the system uses them up according to the order in which they were added. For example:

- Order final price: 35 USD
- Gift card A amount: 15 USD
- Gift card B amount: 40 USD


If the customer adds the coupon code for Gift card A first and then B, the system fully uses up the first coupon code and then 20 USD from the second code.

If the customer adds the coupon code for Gift card B first and then A, the system uses up 35 USD from the first coupon code and leaves the second code unused.

Configuring pages to allow customers to enter coupon codes

To allow customers to enter coupon codes for discounts or gift cards on the pages of your on-line store:

1. Open the **Pages** application.
2. Select the required page in the content tree.
3. Switch to the **Edit -> Design** tab.
4. Add the **Coupon codes** [web part](#) to the appropriate web part zone.
5. Configure the web part properties.

 We strongly recommend leaving the web part's **Show Apply button** property enabled. This allows customers to instantly apply coupon codes while viewing the page.

If *Show Apply button* is disabled, the web part only applies the entered coupon code when moving to the next step in the [checkout process](#).

6. Click **Save & Close**.

The *Coupon codes* web part uses a [transformation](#) to display the coupon codes applied to the customer's shopping cart. The transformation allows you to format the coupon codes according to any requirements of your website's design.

When configuring the *Coupon codes* web part, assign the transformation into the **Transformation name** property. You can either **Select** an existing transformation or create a **New** one.



When writing the code of transformations for coupon code items:

- Use the **Code** field to access the coupon code text – `{% Code %}` in Text / XML transformations, `<%# Eval<string>("Code") %>` in ASCX transformations.
- If you apply [hotfix 11.0.29](#) or newer, you can use the **IsApplied** field to evaluate whether the coupon code is valid for the current shopping cart – `{% IsApplied %}` in Text / XML transformations, `<%# Eval<bool>("IsApplied") %>` in ASCX transformations. This allows you to add formatting options for coupon codes that are no longer valid, for example after products are removed from the cart and the total price falls under the value required by the coupon code's discount.
- Call the **GetDiscountCouponCodeRemoveButton** method to add elements that allow customers to remove applied coupon codes. The method generates an `<input>` element (of the *submit* type) that removes the specified coupon code from the shopping cart when clicked. The method has the following parameters:
 - *couponCode* (required) – the text of the removed coupon code.
 - *cssClass* (optional) – the name of the CSS class assigned to the generated input element.
 - *buttonText* (optional) – the text caption of the input element (assigned to the *value* attribute).

Example - Text / XML transformation

```
<div>
  <div class="label">
    <span>{% HTMLEncode(Code) %}</span>
  </div>
  <div class="button">
    {% GetDiscountCouponCodeRemoveButton(Code, "btn") %}
  </div>
</div>
```

Your shopping cart

	Colombia Carlos Imbachí (4 oz) (In stock)	\$8.40
Qty <input type="text" value="2"/>	UPDATE REMOVE	
	Espro Press (2 cups) (In stock)	\$52.42
Qty <input type="text" value="1"/>	UPDATE REMOVE	

Discount coupon

 [APPLY](#)

DG-0AQ70 [REMOVE](#)

GIFT-10 [REMOVE](#)

Subtotal	\$60.82
Order discounts	- \$6.08
Gift cards	- \$10.00
Total (incl. tax estimate)	\$44.74

Review your shopping cart and checkout

[CHECKOUT](#)

After adding the *Coupon codes* web part to your website's checkout pages, you can distribute coupon codes among your customers. Customers can then use these codes to receive discounts or redeem gift cards.



Note: Different coupon codes of a single gift card can be applied together to the same order (unlike other types of [discounts](#) that use coupon codes). For example, if a customer knows 2 different coupon codes for a 10 USD gift card, they can redeem both codes for the same order to reduce the final price by 20 USD.

Displaying gift card values to customers

In [invoices](#) and other [e-commerce notification emails](#), you can show the values of redeemed gift cards by including the `{% OtherPayments%}` or `{%OtherPaymentsSummary%}` macros in the corresponding templates.


To display information about redeemed gift cards on your store's [checkout](#) pages:

- Use the [Shopping cart totals web part](#) with the **Total to display** property set to *Other payments total*.
- If you wish to display detailed information including the names of gift cards, use the **Static text** web part with the `{% EcommerceContext.CurrentShoppingCart.OtherPaymentsSummary%}` macro. See: [Displaying shopping cart details on pages](#)

Handling payments fully covered by gift cards

If you distribute gift cards without setting an appropriate *Minimum order amount*, customers may create orders where the value of a redeemed gift card exceeds the total price. In these cases, the gift card fully covers the payment and the final price of the order is zero. Many [payment methods](#) reject payments whose value is zero (for example [PayPal](#)), which may be confusing for customers.

For these situations, we recommend configuring your site to skip the payment page of your [checkout process](#) and directly display the "Thank you" or "Order completed" page:

1. Open the **Pages** application.
2. Select the root page of your site's checkout process.
3. On the **Design** tab, configure the **Page wizard manager** web part.
4. Click **Edit value** () next to the **Content filter -> Final step URL** property and enter the following macro expression:

```
{% EcommerceContext.CurrentShoppingCart.GrandTotal > 0 ? "~/Store/Payment-page" :  
  "~/Store/Thank-you" %}
```



The macro ensures that the **Final step URL** is `~/Store/Thank-you` if the shopping cart's grand total is zero, but `~/Store/Payment-page` in all other cases. You need to replace the page URLs with the appropriate values of your site's pages.

5. Click **OK** and then **Save & Close**.

For purchases where the final price is zero, the system now creates [orders](#) without displaying the payment page to customers.

Because such orders are not processed by a standard [payment method](#), your store's administrators may need to manually adjust the [status](#) of the orders or mark them as paid in the **Orders** application. Administrators can identify orders paid using gift cards based on their total price, or by checking the details on the **Items** tab.



Tip: You can also add the `{% EcommerceContext.CurrentShoppingCart.GrandTotal > 0 %}` macro condition into the **Visible** property of other checkout web parts that you wish to hide in these cases (for example the **Payment method selection** web part).