

The system stores payment results in XML format. In the API, the results are managed using *CMS.Ecommerce.PaymentResultInfo* objects. Every XML node of a payment result is represented by a single payment result item, which can be managed using the *CMS.Ecommerce.PaymentResultItemInfo* object.

By default, the system provides the following payment result items:

- **Payment date** – the date and time when the payment result was last updated.
- **Payment method** – the [payment method](#) that was used for the transaction.
- **Payment is completed** – indicates whether the payment was completed successfully.
- **Payment is authorized** – indicates whether the payment was successfully authorized for later capture (for payment gateways that use delayed funds capture).
- **Payment is failed** – indicates whether the payment transaction failed (cannot be set together with the *Payment is completed* result item).
- **Payment approval URL** – the URL of a page (typically external) where customers provide additional payment details and finalize the transaction. If set, the system automatically redirects customers to this URL after they submit the related payment form.
- **Payment status** – the status of the payment, e.g. *Completed*, *Failed*, etc. (your custom status).
- **Payment transaction ID** – a unique identifier for the payment transaction generated by the payment gateway.
- **Payment authorization ID** – a unique identifier for authorization transactions generated by the payment gateway (for payment providers that use transactions with delayed capture of funds).
- **Payment description** – text describing the result of the payment transaction.

Payment result items have the following properties:

- **Name** – represents a unique identifier of the payment result item.
- **Header** – represents the name of the payment result item visible to users in the Orders application (simple text or [localizable string](#)).
- **Text** – determines the outer representation of the payment result item value visible to users in the Orders application (simple text or localizable string).
- **Value** – determines the inner representation of the payment result item value used by developers.

Example - Order payment result XML definition

The following example shows an XML definition of an order payment result extended by the *authorizationcode* item used by Authorize.NET:

```
<result>
  <item header="{ $PaymentGateway.Result.Date$}" name="date" value="7/28/2017 3:06:16 PM"></item>
  <item header="{ $PaymentGateway.Result.PaymentMethod$}" name="method" text="Credit Card - Authorize.NET" value="5"></item>
  <item header="{ $PaymentGateway.Result.IsCompleted$}" name="completed" text="{ $PaymentGateway.Result.PaymentCompleted$}" value="1"></item>
  <item header="{ $PaymentGateway.Result.Status$}" name="status" text="{ $PaymentGateway.Result.Status.Completed$}" value="completed"></item>
  <item header="{ $PaymentGateway.Result.TransactionID$}" name="transactionid" value="0"></item>
  <item header="{ $PaymentGateway.Result.Description$}" name="description" value="I00001: Successful.1: This transaction has been approved."></item>
  <item header="{ $AuthorizeNet.AuthorizationCode$}" name="authorizationcode" value="000000"></item>
</result>
```

The system stores the payment result XML values in the *OrderPaymentResult* column of order records (*COM_Order* database table).

Example - Order payment result in the Orders application

The following example shows a payment result visible to your on-line store administrators in the **Orders** application while editing a selected [order](#) on the **Billing** tab:

Date: 7/28/2017 3:06:16 PM

Method: Credit Card - Authorize.NET

Is completed: YES

Status: Completed

Transaction ID: 0

Authorization code: 000000



The payment result is unavailable (N/A) until the payment gateway processor updates it.

You do not need to specify both item value and item text if they are identical. The system automatically renders payments results as follows:

1. Render the item text if available.
2. If not, render the item value.

Customizing payment results

To manage the default payment result items, use the corresponding properties of **PaymentResultInfo** objects:

- `PaymentDate` (DateTime)
- `PaymentMethodID` (int)
- `PaymentMethodName` (string)
- `PaymentIsCompleted` (bool)
- `PaymentIsAuthorized` (bool)
- `PaymentIsFailed` (bool)
- `PaymentApprovalUrl` (string)
- `PaymentStatusName` (string)
- `PaymentStatusValue` (string)
- `PaymentTransactionID` (string)
- `PaymentAuthorizationID` (string)
- `PaymentDescription` (string)

To get or set custom payment result items, call the **GetPaymentResultItemInfo**(string itemName) and **SetPaymentResultItemInfo**(PaymentResultItemInfo itemObj) public methods.

The following examples demonstrate how to work with a custom payment result item (storing an authorization code) in the code of a [custom gateway provider class](#):



Setting the authorization code

```
using CMS.Ecommerce;

...

// Prepares a PaymentResultItemInfo object for the authorization code
PaymentResultItemInfo item = new PaymentResultItemInfo();
item.Header = "{$AuthorizeNet.AuthorizationCode$}";
item.Name = "authorizationcode";
item.Value = "00000";

// Saves the custom item into the PaymentResultInfo object processed by the gateway
// provider
PaymentResult.SetPaymentResultItemInfo(item);
```

Getting the authorization code

```
using CMS.Ecommerce;

...

// Gets the custom item from the PaymentResultInfo object processed by the gateway
// provider
PaymentResultItemInfo item = PaymentResult.GetPaymentResultItemInfo
("authorizationcode");
```