

“CreditReporting” REST/JSON service

Interface #9: getAllCompletedOrders

ver. 15.01.2026

Updates:

- 15.01.2026: HTTP status codes not used anymore for application return
 - 20.05.2025: Timestamps now processed as ISO 8601 formatted strings.
- [ver. 22.05.2020]

Purpose: Getting a list of orders completed within specified interval of time.

Interface address : <https://online.igkservice.lv/rscr/getAllCompletedOrders>

Request method : POST

Contents:

1. Request structure
2. Response structure
3. Response status codes
4. Authentication

1. Request structure

1.1. Request parameters:

Parameter name	Type	Required	Explanation
completedFromDate	string*	yes	timestamp the observed interval started
completedToDate	string*	yes	timestamp the observed interval ended

* String in ISO 8601 format "yyyy-MM-dd'T'HH:mm:ss.SSSXXX", e.g. "2025-05-14T14:42:14.000+03:00"

1.2. Request sample:

```
{
  "completedFromDate": "2025-05-01T00:00:00.000+03:00",
  "completedToDate": "2025-05-31T23:59:59.999+03:00 "
}
```

2. Response structure

2.1. Response parameters:

Parameter name	Type	Required	Explanation
getAllCompletedOrdersResult	object	yes	
responseCode	string	yes	see “3.2. Application response codes”
responseText	string	yes	details for “responseCode”, see 3.2.
completedOrders	object	no	
seletedOrderStatus	object []	yes	
providerPartyID	string	yes	always "IGK-LV"
providerOrderReference	string	yes	IGK order reference
consumerOrderReference	object	yes	Customer's order reference
statusCode	object	yes	{FINISHED, FINISHED_NEGATIVE, CANCELLED}
name	string	no	registered name
internalID	string	no	IGK's code
country	string	no	ISO 3166-1 alpha-2
ordered	string*	no	timestamp the order was placed
requestedDelivery	string*	no	timestamp of deadline
delivered	string*	no	timestamp of the first reply

* String in ISO 8601 format "yyyy-MM-dd'T'HH:mm:ss.SSSXXX", e.g. "2025-05-14T14:42:14.000+03:00"

2.2. Response sample

```
{
  "getAllCompletedOrdersResult": {
    "responseCode": "OK",
    "responseText": "OK",
    "completedOrders": {
      "seletedOrderStatus": [{
        "delivered": "2025-05-20T13:36:33.000+03:00",
        "internalID": "018-900-684",
        "ordered": "2025-05-20T13:36:33.000+03:00",
        "providerOrderReference": "I-LV-250520133633.21",
        "providerPartyID": "IGK-LV",
        "statusCode": "FINISHED"
      }, {
        "delivered": "2025-05-15T16:55:48.000+03:00",
        "country": "LV",
        "name": "SAMPLE Company",
        "ordered": "2025-05-14T14:46:23.000+03:00",
        "providerOrderReference": "B-LV-250514144214.19",
        "providerPartyID": "IGK-LV",
        "statusCode": "FINISHED"
      }
    ]
  }
}
```

3. Response status codes

3.1. Http response status codes:

Status code	Description	Explanation
200	ok	Always OK, see "responseCode Text" for details

3.2. Application response codes

responseCode / responseText	Explanation
OK / OK	Successful return
NoOrdersAvailable/ No orders available in period {<from-timestamp>, <to-timestamp>}.	Correct return: nothing was found
BadRequest / - No "authHeader" httpHeader. - Cannot unmarshal "authHeader" to JSON object. - AuthHeader: username is not set. - AuthHeader: password is not set. - "CompletedFromDate" mandatory value is not set. - "CompletedToDate" mandatory value is not set. - "CompletedToDate" has wrong value: "<value-from-request>".	Incorrect request parameters
ServerError / <program exception message>	Unexpected program error

4. Authentication

Authentication should be provided by the client in "authHeader" header of the request - as JSON presentation of AuthHeader object.

```
public class AuthHeader {  
    String username, password;  
    public String toJsonString() {  
        return "{\"username\":\""+username+"\", \"password\":\""+password+"\"}";  
    }  
}
```

Example of the client code:

```
Client client = ClientBuilder.newClient();  
WebTarget webTarget = client.target(BASE_URI);  
WebTarget resource = webTarget.path("getAllCompletedOrders");  
Invocation.Builder builder = resource.request(MediaType.APPLICATION_JSON);  
builder.header("authHeader", _authHeader.toJsonString());
```

.end