

YAPI KREDİ BANKASI  
VIRTUAL POS (POSNET)  
XML Services  
INTEGRATION DOCUMENT

## Content

Content.....	2
Introduction .....	4
General Structure of the Service .....	4
Credit Card Transactions.....	5
Sales (Direct Sales) & Sales in Installments at Cash Price .....	5
Marking of Sales (Direct Sales) & Sales in Installments at Cash Price as Mail Order.....	8
Provision & Provision in Installment at Cash Price .....	9
Financialization & Financialization in Installment at Cash Price.....	12
Cancellation transaction (sale, provision, financialization, etc.).....	14
Return transaction .....	16
Unconjugated Returns Transaction.....	18
Transactions of Points .....	19
Points Inquiry .....	19
Points Usage .....	20
Points Return .....	22
Cancellation of Points Usage.....	23
Mixed Transactions.....	25
Cancellation of Mixed Transaction.....	27
Return of Mixed Transaction.....	27
Transactions with Delay Interest (VFT).....	27
Delay Interest Inquiry .....	28
Sales in Installment with Delay Interest.....	29
Return of Sales with Delay Interest.....	31
Personalized - Joker Vadaa Transactions .....	33
Personalized Transaction Inquiry.....	33
Trio Transactions.....	35
Single Payment.....	36
Fixed payment .....	37
Multi-Payment .....	38
Return.....	39
Limit Inquiry.....	40
Reconciliation Services .....	41
Transaction Status Inquiry (Agreement).....	41
Daily Transaction Report Inquiry.....	43
New Generation Payment.....	46

Transaction conducted with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card.....	49
Provision and provision financialization with Turkish Identity Number / Taxpayer Information and card first 6 last 4 information .....	56
Points inquiry with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card .....	59
Points usage with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card.....	60
Error codes .....	62
Steps for Going Live .....	69
History.....	71

## Introduction

This document describes how to integrate into the POSNET system. Shared service URLs are for the testing environment. The procedures required to move to the production environment are provided at the end of the document. After completing your tests in the test environment, you must send your request to go live to [possup@yapikredi.com.tr](mailto:possup@yapikredi.com.tr) In the mail attachment you will send, you need to include distinctive information (MERCHANT\_ID, TERMINAL\_ID, POSNET\_ID, SOURCE\_IP, ORDER\_NO, TRANSACTION\_DATE, etc.) and the date of the transaction.

Merchants that will use the POSNET system are required to provide Static IP addresses to the bank for both testing and live environments.

In order to provide the best experience to the merchants integrated to the system, when the card number is entered on the screens, the Joker Vadaa inquiry that provides personal discounts should be checked.

It is recommended that the merchant provides the opportunity to use World Points for payment to ensure the satisfaction of its customers.

It is recommended that the merchant make 3D Secure (3-dimensional security) payment integration both in order to reduce its own risk and to ensure the security of customer information. Cancellation, Refund, Points Transactions, Personal - Joker Vadaa Transactions, Delay Interest Transactions in this document constitute the continuation of 3D Secure integration document.

## General Structure of the Service

Posnet XML service is a service that enables Posnet merchants to make Posnet transactions by sending XML documents. The merchants should POST the xml document after encoding the xml document they create with UTF-8 URL Encode to `<%XML_SERVICE_URL%>` address (testing environment: <https://setmpos.ykb.com/PosnetWebService/XML>) as the environment variable at “xmldata” parameters with Content-Type=`application/x-www-form-urlencoded; charset=utf-8`. The result is returned to the merchant as an XML document.

Example URL:

<https://setmpos.ykb.com/PosnetWebService/XML?xmldata=%3CposnetRequest%3E%0D%0A++%3Cmid%3E...>

The following information included in the service integration is communicated to merchants by mail and this information varies between test and live environments. It is recommended that this information should not be embedded in the code, but should be defined and used as environment variable.

Key	Type	Description	Sample Data
MERCHANT_ID	String	10 digit YKB (Yapı Kredi Bank) merchant number	6706598320
TERMINAL_ID	String	8 digits YKB merchant terminal number	67005551
POSNET_ID	String	Up to 16 digits, YKB merchant POSNET number. It is used in 3D Secure encryption transactions.	9644
XML_SERVICE_URL	String	Bank integration service address	<a href="https://setmpos.ykb.com/PosnetWebService/XML">https://setmpos.ykb.com/PosnetWebService/XML</a>

The MERCHANT\_ID, TERMINAL\_ID, POSNET\_ID information can also be found on the Merchant information page on the Merchant Admin Screens.

### NOTES:

- For each service request integration, following information shall be added to Request Header: X-MERCHANT-ID, X-TERMINAL-ID, X-POSNET-ID, X-CORRELATION-ID. (CorrelationId: Unique value of the transaction to be set by the merchant, and will allow a quick return of Posnet support team when a problem is reported. Order number (XID) can be set. If more than one service call is created for the same order, it can be separated by the characters (max 24) to be added to the end of the order number)
- If the order id value is desired to be sent between 1-24 characters instead of 24 characters or if the same order id will be used on different dates, the activation of the order id parameter for the merchant should be requested from the Posnet Support team. If the order id parameter is active, transactions can be sent with the same order id on different days. In financialization, cancellation, return and reconciliation transactions where the order id parameter is active and the order id is called, the order date of the original transaction must be sent in the orderDate field. Detailed information is given in the titles of the related transactions.
- In order to prevent the data to be sent to the service to disrupt the xml structure, xml escape characters must be sent after being encoded.
- UTF-8 encoding is supported in bank systems. The request's content must be set to charset = UTF-8, and the request content must be encoded as UTF-8.

## Credit Card Transactions

### Sales (Direct Sales) & Sales in Installments at Cash Price

The sales transaction (also known as Direct Sales) enables customers to withdraw the specified amount from their card and deposit it in the merchant account. The number of installments can also be specified in the sales transaction. In this case, the amount is reflected to the card in installments (without delay interest). The transactions made in this way are called Sales in Instalment at Cash Price

If the sale is created in instalment at cash price, no delay interest is calculated. The specified amount is divided into the specified installments, and the debit is charged to the customer's card account whenever the installment is due.

Posnet merchants wishing to add a delay interest to their sales in installments must first calculate this delay interest and then add it to the original amount. For example, if a 100 TL transaction is divided into 2 installments, if the 10% delay interest is to be added, Posnet merchant should send 110 as the amount and 2 as installment. Thus, 55 TL will be withdrawn from the customer's card for 2 months and deposited into the merchant account. Besides this merchant can create a [Sales with Delay Interest](#) transaction.

Since the sales transaction gains financial value when the group is closed, it can be canceled until the group is closed. If a group-closed transaction wants to be cancelled, [0211](#) error code is received. In this case it must be returned. Sales transactions that are not canceled until the end of the day become financial.

### Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>670000067</mid>
4.   <tid>67000067</tid>
5.   <tranDateRequired>1</tranDateRequired>
6.   <sale>
7.     <amount>2451</amount>
8.     <ccno>4506349116608409</ccno>
9.     <currencyCode>TL</currencyCode>
10.    <cvc>000</cvc>
11.    <expDate>0703</expDate>
12.    <orderID>1s3456z8901234567890123</orderID>
13.    <installment>02</installment>
14.    <!--
15.     <koiCode>1</koiCode>
16.    -->
17.    <!--
18.     <subMrcId></subMrcId><mrcPfId></mrcPfId><mcc></mcc>
19.    -->
20.    <!--
21.     <tckn></tckn><vkn></vkn><subDealerCode></subDealerCode>
22.    -->
23.   </sale>
```

<b>posnetRequest – sale</b>	
It is used for direct sales transaction.	
<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.
<b>sale</b>	
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>ccno</b>	Credit Card Number
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>cvc</b>	Credit card security number - CVV2
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>orderID</b>	Order number. 24-digits alphanumeric Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>mailorderflag</b>	It should be used if the transactions to be performed as Ecom via online integration are desired to be realized as Mail Order. "Y" must be set for Mail Order. It can only be set in Sales and Sales in Installments at Cash Price.
<b>koiCode</b>	Joker Vadaa campaign type (customized transaction code). It should be used by querying the personal transaction list by questioning by card number. It is optional. <b>If a value is not set, it should not be contained in xml.</b> 1: Additional Installment 2: Postponing Installments 3: Extra Points 4: Gaining minutes 5: Postponing Statement 6: Special Delay Interest
<b>subMrclId</b>	If used by a Posnet payment facilitator, the payment facilitator must send to the Posnet system the information it identifies with its customers through these 3 fields. <b>Standard merchants without payment facilitators should not include these fields in xml.</b>
<b>mrcPfld</b>	
<b>Mcc</b>	
<b>tckn</b>	If sub-dealer transaction will not be made, TR Identity Number / Taxpayer Identity Number / SubdealerCode fields should not be sent.
<b>vkcn</b>	

**subDealerCode**

In order for a main dealer to send a sub-dealer transaction, the sub-dealer must send at least one of the registered TR ID Number and Taxpayer Identity information.

When the transaction is sent with TR Identity Number / Taxpayer Identity Number:

- If there is only one sub-dealer registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the sub-dealer.
- If there are no sub-dealers registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the main dealer.



- There can be more than one sub-dealer registered with this Identity Number/Taxpayer Identity Number. In this case, we expect the subdealerCode to be sent. In case the subDealerCode is not specified, it will not be known which sub-dealer will be used, so the transaction will take place from the main dealer.

### Response Example

```

1. <?xml version='1.0' encoding='iso-8859-9'?>
2. <posnetResponse>
3.   <approved>1</approved>
4.   <hostlogkey>019676067890000191</hostlogkey>
5.   <authCode>760678</authCode>
6.   <tranDate>190519161445</tranDate>
7.   <instInfo>
8.     <inst1>00
9.     <amnt1>0000000000</amnt1>
10.  </instInfo>
11.  <pointInfo>
12.    <point>00000000</point>
13.    <pointAmount>0000000000</pointAmount>
14.    <totalPoint>09840134</totalPoint>
15.    <totalPointAmount>000004920067</totalPointAmount>
16.  </pointInfo>
17. </posnetResponse>

```

<b>posnetResponse – instInfo – pointInfo</b>	
Hostlogkey must be recorded for being used in transactions such as cancellation/refund.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful 2: <a href="#">Previously performed</a>
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>Hostlogkey</b>	The unique Id on the system. It should be kept for refund/cancellation transactions.
<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is conducted, it must be kept for refund and cancellation transactions.
<b>tranDate</b>	It is the actual date of operation according to the system setting and the date that will appear on the merchant administrator screens. If tranDateRequired = 1 is entered in the request, it is received. The format is as following: YYMMDDHHMMSS
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	

<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

#### Response Example (Incorrect)

```

1. <posnetResponse>
2.   <approved>0</approved>
3.   <respCode>0014</respCode>
4.   <respText>RED -HATALI KART 0014</respText>
5.   <yourIP>10.105.182.74</yourIP>
6. </posnetResponse>

```

#### Response Example (Previously performed)

For transactions made using OrderId (Sales, Provision, Points Usage, Mixed, etc.), uniqueness control is made over OrderId in order to receive repetitive payment in the Posnet systems and to prevent potential customer dissatisfaction. This singularity check can happen in 2 scenarios:

1. If the OrderID parameter is active: In this scenario, singularity control is carried out with OrderId + order date. The same OrderId can be used as long as it is on different dates.
2. If the OrderID parameter is passive: In this scenario, the singularity check is carried out via OrderID. When a request is made to the service again with a previously used OrderID, the response will be as follows. If the transaction result could not be obtained due to reasons such as network in the first request, the order can be completed with approved: 2 and respCode: 127 in the second request.

```

1. <posnetResponse>
2.   <approved>2</approved>
3.   <respCode>0127</respCode>
4.   <respText>ORDERID DAHA ONCE KULLANILMIS 0127</respText>
5.   <hostlogkey>020527337090000191</hostlogkey>
6.   <authCode>273370</authCode>
7.   <tranDate>190703093340</tranDate>
8.   <instInfo>
9.     <inst1>00
10.    <amnt1>000000000000</amnt1>
11.  </instInfo>
12.  <pointInfo>
13.    <point>00000000</point>
14.    <pointAmount>000000000000</pointAmount>
15.    <totalPoint>00000000</totalPoint>
16.    <totalPointAmount>000000000000</totalPointAmount>
17.  </pointInfo>
18. </posnetResponse>

```

#### Marking of Sales (Direct Sales) & Sales in Installments at Cash Price as Mail Order

Sales / Sales in Installments at Cash are sent as an e-commerce transaction (E-Commerce / Ecom) unless a change is performed on the online web services, and the e-commerce authorizations of the card made are controlled.

In Mail Order (MO) transactions that can be performed with file transfer (FTP), the Mail Order (MO) authorization of the processed card is checked. In the classical method, MO operations can be performed with sending files via FTP, and they can be performed with additional parameters with web services. In this case, `<mailorderflag>Y</mailorderflag>` should be included in the xml pattern.

In order for this process to be realized as an end-to-end Mail Order, the Workplace Number sent in the Mid parameter must be defined as a Mail Order workplace in the bank system.

Integration parameters of an online transaction that is required to be realized as a Mail Order can be completed by adding only the mailorderflag parameter to a Standard Sale/Sales in Installments at Cash Price. Sample request package is as follows;

## Request Örneği

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6700000067</mid>
4.   <tid>67000067</tid>
5.   <tranDateRequired>1</tranDateRequired>
6.   <sale>
7.     <amount>2451</amount>
8.     <ccno>4506349116608409</ccno>
9.     <currencyCode>TL</currencyCode>
10.    <cvc>000</cvc>
11.    <expDate>0703</expDate>
12.    <orderId>1s3456z8901234567890123</orderId>
13.    <installment>02</installment>
14.    <mailorderflag>Y</mailorderflag>
15.    <!--
16.    <koiCode>1</koiCode>
17.    -->
18.    <!--
19.    <subMrcId></subMrcId><mrcPfId></mrcPfId><mcc></mcc>
20.    -->
21.    <!--
22.    <tckn></tckn><vkn></vkn><subDealerCode></subDealerCode>
23.    -->
24.  </sale>
25. </posnetRequest>
```

The response packages of the transactions in which the Mail Order Flag parameter is set are exactly the same as the response packages of the [Sales and Sales in Installments at Cash Price](#) transactions.

### Provision & Provision in Installment at Cash Price

The provisioning process is used to check if a credit card is eligible for shopping and, if appropriate, to block the requested amount on the card. This does not appear on the cardholder's statement, but reduces the card's limit.

The purpose of the provision process is to place a certain amount of blockage on the card, actual financialization of the product after the actual delivery of the product. If financialization is to be made immediately, not after product delivery, or if the product sold is a product that can be delivered to the customer immediately, it is recommended to use Sales transaction instead of Provision.

In Posnet transactions, which include the provision and financialization steps, the provision transaction is the prioritized transaction to be completed. Financialization cannot be realized without provision; that means, the desired amount cannot be withdrawn from the customer's card.

If the provision is not financialized within a certain period of time, the bank of the cardholder removes the freeze automatically after a few days. The merchant can also cancel the provision without waiting for the expiry of this period in order to remove the freeze placed on the customer's card. The freeze on the customer account will be removed and customer satisfaction will be increased. In addition, non-financialized transactions can be financialized later in this period, but in this case the freeze will be removed and it is possible to get errors such as insufficient limits. The customer in card online transactions, not in the card statement, can view the provision transactions that are not financialized.

### Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
```

```

3.     <mid>6706598320</mid>
4.     <tid>67005551</tid>
5.     <tranDateRequired>1</tranDateRequired>
6.     <auth>
7.         <amount>175</amount>
8.         <ccno>4048097006508842</ccno>
9.         <currencyCode>TL</currencyCode>
10.        <cvc>000</cvc>
11.        <expDate>2002</expDate>
12.        <orderID>YKB_TST_1905210122001234</orderID>
13.        <installment>00</installment>
14.        <!--
15.        <koiCode>1</koiCode>
16.        -->
17.        <!--
18.        <subMrcId></subMrcId><mrcPfid></mrcPfid><mcc></mcc>
19.        -->
20.        <!--
21.        <tckn></tckn><vkn></vkn><subDealerCode></subDealerCode>
22.        -->
23.    </auth>
24. </posnetRequest>

```

#### posnetRequest – auth

It is used for the provision transaction.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.

#### auth

<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>ccno</b>	Credit Card Number
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>cvc</b>	Credit card security number - CVV2
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>orderID</b>	Order number. 24-digits alphanumeric Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>koiCode</b>	Joker Vadaa campaign type (customized transaction code). It should be used by querying the personal transaction list by questioning by card number. It is optional. <b>If a value is not set, it should not be contained in xml.</b> 1: Additional Installment 2: Postponing Installments 3: Extra Points 4: Gaining minutes 5: Postponing Statement 6: Special Delay Interest
<b>subMrcId</b>	If used by a Posnet payment facilitator, the payment facilitator must send to the Posnet system the information it identifies with its customers through
<b>mrcPfid</b>	

**mcc**

these 3 fields. Standard merchants without payment facilitators should not include these fields in xml.

<b>tckn</b>	<p>If sub-dealer transaction will not be made, TR Identity Number / Taxpayer Identity Number / SubdealerCode fields should not be sent.</p> <p>In order for a main dealer to send a sub-dealer transaction, the sub-dealer must send at least one of the registered TR ID Number and Taxpayer Identity information.</p> <p>When the transaction is sent with TR Identity Number / Taxpayer Identity Number:</p> <ul style="list-style-type: none"> <li>• If there is only one sub-dealer registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the sub-dealer.</li> <li>• If there are no sub-dealers registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the main dealer.</li> <li>• There can be more than one sub-dealer registered with this Identity Number/Taxpayer Identity Number. In this case, we expect the subdealerCode to be sent. In case the subDealerCode is not specified, it will not be known which sub-dealer will be used, so the transaction will take place from the main dealer.</li> </ul>
<b>vkcn</b>	
<b>subDealerCode</b>	

#### Response Example

```

1. <?xml version='1.0' encoding='iso-8859-9'?>
2. <posnetResponse>
3.   <approved>1</approved>
4.   <hostlogkey>019676067890000191</hostlogkey>
5.   <authCode>760678</authCode>
6.   <tranDate>190519161445</tranDate>
7.   <instInfo>
8.     <inst1>00
9.     <amnt1>000000000000</amnt1>
10.  </instInfo>
11.  <pointInfo>
12.    <point>00000000</point>
13.    <pointAmount>0000000000</pointAmount>
14.    <totalPoint>09840134</totalPoint>
15.    <totalPointAmount>000004920067</totalPointAmount>
16.  </pointInfo>
17. </posnetResponse>

```

#### posnetResponse – instInfo – pointInfo

Hostlogkey must be recorded for being used in transactions such as cancellation/refund. (If the hostlogkey is not to be used in cases where the merchant OrderID parameter is active, the order id + order date should be saved for later use in transactions such as cancellation / return.)

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system. It should be kept for refund/cancellation and financialization transactions.

<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is conducted, it must be kept for refund and cancellation transactions.
-----------------	--



<b>tranDate</b>	It is the actual date of operation according to the system setting and the date that will appear on the merchant administrator screens. If tranDateRequired = 1 is entered in the request, it is received. The format is as following: YYMMDDHHMMSS
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	
<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

### Financialization & Financialization in Installment at Cash Price

Financialization means that the provision becomes financial, that means the amount is withdrawn from the customer's card account and deposited in the merchant account. In order to complete this transaction, it is necessary to make the provision. This transaction appears on the card statement and the financial records of the merchant. This transaction must be performed in order to become financial.

The amount (financialization amount) submitted in the financialization process may not exceed the provision amount, but may be less than the provision amount. For example, if a provision of TL 10 is placed, a maximum of 10 TL can be financialized. If in some cases it is necessary to exceed the provision amount, it is necessary to contact [possupp@yapikredi.com.tr](mailto:possupp@yapikredi.com.tr). In this case, it may be possible to financialize an amount exceeding the provision by defining a percentage of amounts that may exceed the provision. A company with a 10% percentage that can exceed the provision can financialize a 100 TL provision for a maximum of 110 TL.

A provision that cannot be sent in installments cannot be split into installments during the financialization. The number of installments that can be found in the financialization transaction of a provision sent in installments is equal to the number in the provision at the maximum. For example; a provision transaction sent in 4 installments will not be financialized in 5 installments, financialization in 3 installments will be approved.

Financialization can be realized after the provision is automatically removed due to non-financialization; however, in this case, it would be possible to receive an error of insufficient limit since the freeze on the card will be removed.

According to the financialization date, cancellation can be completed on the same day while refund transaction can be completed on the following days. In partial refunds, a cancellation request cannot be sent to a transaction with a refund. For example, a cancellation request cannot be sent after a refund of 5 TL is sent to the provision transaction, which is sent as 10 TL.

Workplaces with active Order ID parameters cannot make provisions with the same Order ID on the same day, otherwise the error "CANNOT FINANCIALIZE AGAIN SINCE FINANCIALIZATION HAS BEEN DONE TODAY WITH THIS ORDER ID" will be received. This is to avoid the occurrence of two financialization transactions with the same OrderId and the same date.

#### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <capt>

```

```
6.      <amount>175</amount>
7.      <currencyCode>TL</currencyCode>
8.      <hostLogKey>019799151790000191</hostLogKey>
9.      <installment>00</installment>
10.     </capt>
11.     </posnetRequest>
```

<b>posnetRequest – capt</b>	
It is used to financialize the provision.	
<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>capt</b>	
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>hostLogKey</b>	The unique Id on the system. The return value from the provision service must be used.
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>orderID</b>	In cases where the OrderID parameter of the merchant is activated, if the hostlogkey is not sent, the order number of the original transaction should be sent.
<b>orderDate</b>	In cases where the OrderID parameter of the merchant is activated, the order date of the original transaction should be sent in this field for the singularity control of the transaction to be financialized with orderID. The format must be YYYYMMDD.

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019799151790000191</hostlogkey>
4.   <authCode>991517</authCode>
5.   <instInfo>
6.     <inst1>00
7.     <amnt1>000000000000</amnt1>
8.   </instInfo>
9.   <pointInfo>
10.    <point>00000000</point>
11.    <pointAmount>0000000000</pointAmount>
12.    <totalPoint>04860169</totalPoint>
13.    <totalPointAmount>000002430084</totalPointAmount>
14.  </pointInfo>
15. </posnetResponse>

```

<b>posnetResponse – instInfo – pointInfo</b>	
At the of the financialization process, it is possible to access the number of instalments and points information.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system. It should be kept for refund/cancellation and financialization transactions.

<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is conducted, it must be kept for refund and cancellation transactions.
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00

	or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	
<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

### Cancellation transaction (sale, provision, financialization, etc.)

It is used to cancel a transaction made during the day. Canceled transactions are not financial and do not appear in the customer statement in any way. After the cancellation, the limit of the customer's credit card is increased by the transaction amount at the end of the day at the latest.

At the end of the day, the sales transactions become financial and cancellation of the financialized transactions is only possible with the return transaction. For this reason, cancellation option is not displayed on Posnet Admin Screens for sales realized on the previous days and cancellation service cannot be used.

If the sales cancellation is approved, no record of the sales transaction will be shown on the customer statement. Therefore, this is the most preferred method of canceling a sale. This transaction can be done from the transaction detail by using the interface created by the merchant or by displaying the sales transaction in Posnet Admin Screens.

The cancellation of the provision is to remove the freeze placed on the card. Like the provision transaction, this transaction does not appear on the card statement.

Cancellation of the financialization process prevents the financialization from being reflected on the card statement. However, it does not remove the freeze placed on the card. For this, it is necessary to remove the provision. In case of cancellation of the financialization realized on the previous day, "RED-GECERSIZ ISLEM 0229" error is received. In this case, it is necessary to perform a return transaction.

The cancellation of the point usage process is shown separately to the customer in the "WorldPoint Information" section of the card statement.

If there is a refund on a transaction, the Cancellation transaction will yield an error. For example, if a refund of 3 TL is made within a sales transaction of 10 TL, canceling the 10 TL transaction will leave the 3 TL return transaction unallocated, so this transaction is not allowed.

If you want to withdraw the entire 10 TL transaction with a 3 TL refund, you can make another 7 TL refund.

Or, after canceling the 3 TL refund, you can cancel the original amount of 10 TL. When making a plan in this regard, the issue to be considered is that the cancellation can only be made on the same day.

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <reverse>
6.     <transaction>sale</transaction>
7.     <hostLogKey>050215105426770842</hostLogKey>
8.     <!--
9.     <orderID></orderID>
10.    -->

```

```

11.      <!--
12.      <authCode></authCode>
13.      -->
14.      </reverse>
15. </posnetRequest>

```

#### posnetRequest – reverse

It is used for cancellation (reverse) within the day of the transaction.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>reverse</b>	
<b>transaction</b>	The type of operation to be canceled is set in this field. Can be used as cancellation of sales, cancellation of provision, cancellation of financialization, cancellation of point usage, cancellation of VFT transaction, cancellation of refund. Sales: sale Provision: auth Financialization: capt Points Usage: pointUsage VFT Transaction: vftTransaction Return Transaction: return
<b>hostLogKey</b>	The unique Id on the system. For the relevant transaction, it is used as returned from the service.
<b>orderID</b>	Order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. However, this method is worse in terms of performance than the use of hostlogkey. <b>If Hostlogkey is used, this field should not be included in xml.</b> In cases where the OrderID parameter of the merchant is activated, minimum 1 and maximum 24 digits must be sent.
<b>orderDate</b>	In cases where the OrderID parameter of the merchant is activated, the order date of the original transaction should be sent in this field for the singularity control of the transaction to be financialized with orderID. The format must be YYYYMMDD.
<b>authCode</b>	It is the authorization code generated based on the system authorization. Transaction with delay interest (VFT) is used as returned from the service. <b>This field is mandatory for VFT cancellation; other cancellations should not be included in xml.</b>

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019799159990000191</hostlogkey>
4.   <authCode>000000</authCode>
5. </posnetResponse>

```

The result of the cancellation is reached.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.

<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization. The default value is 000000 for transactions other than the cancellation of transaction with delay interest.

### Response Example (Incorrect)

```

1. <?xml version='1.0' encoding='iso-8859-9'?>
2. <posnetResponse>
3.   <approved>0</approved>
4.   <respCode>0220</respCode>
5.   <respText>IPTAL ISLEMI YAPILMIS           0220</respText>
6.   <yourIP>10.105.182.74</yourIP>
7. </posnetResponse>

```

### Return transaction

This transaction can completely cancel any sales or financialisations made in the past or only refund a certain amount. It appears as a separate transaction in the card statement and does not delete the canceled transaction's record in the statement.

In refund transactions, the amount to be refunded cannot exceed the original transaction amount and the sum of the refunds previously made on the same original transaction. For example, if you returned 3 TL on the first refund of a 10 TL transaction, you may return a maximum of 7 TL on the second refund. Refunds can be made as long as the total amount of the refunds of a transaction does not reach the amount of the original transaction.

It is also possible to return transactions on the day of the transaction. Thus, unlike cancellation, only part of the transaction can be returned. The refund of the transactions with delay interest is separately evaluated and there is a special return service. Please see: [Transactions with Delay Interest \(VFT\)](#) > [Sales Return with Delay Interest](#)

In the return process, it is necessary to come with the information of the original transaction to be returned; For example, in the case of a partial refund after the sale, the second partial refund should be made with the information returned from the sale, not the information returned from the first partial refund.

In case the order id parameter is active, the hostlogkey of the transaction to be returned or order id + order date should be used. For example, after the pre-authorization and financialization process is done, the hostlogkey or order id + order date of the financialization should be used for the return. When it is desired to make a return again after the partial return of the sales transaction, the return should be made with the hostlogkey returned from the sale or the order id + order date with which the sale was made.

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <tranDateRequired>1</tranDateRequired>
6.   <return>
7.     <amount>100</amount>
8.     <currencyCode>TL</currencyCode>
9.     <hostLogKey>019676067890000191</hostLogKey>
10.    <!--
11.     <orderID></orderID>
12.    -->
13.   </return>
14. </posnetRequest>

```



**posnetRequest – return**

Is used to return the transaction.

**posnetRequest**

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.

return	
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>hostlogkey</b>	The unique Id on the system. For the relevant transaction, it is used as returned from the service.
<b>orderID</b>	Order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. If a financialized transaction with 3D Secure payment management is being refunded, it is necessary to complete 24 digits by putting "TDSC" in front of the orderId with 20 digits. Example: TDSCYKB_0000190526121122 This method is worse in terms of performance than the use of hostlogkey. <b>If Hostlogkey is used, this field should not be included in xml.</b> In cases where the OrderID parameter of the merchant is activated, minimum 1 and maximum 24 digits must be sent.
<b>orderDate</b>	In cases where the OrderID parameter of the merchant is activated, the order date of the original transaction should be sent in this field for the singularity control of the transaction to be financialized with orderID. The format must be YYYYMMDD.

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019799179990000191</hostlogkey>
4.   <authCode>991799</authCode>
5.   <instInfo>
6.     <inst1>00
7.     <amnt1>0000000000</amnt1>
8.   </instInfo>
9.   <pointInfo>
10.    <point>0000000</point>
11.    <pointAmount>0000000000</pointAmount>
12.    <totalPoint>09856296</totalPoint>
13.    <totalPointAmount>00004928148</totalPointAmount>
14.  </pointInfo>
15. </posnetResponse>

```

#### posnetResponse – instInfo – pointInfo

At the end of return transaction, it is possible to access the number of instalments and point information.

posnetResponse	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.
instInfo	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.

<b>pointInfo</b>	
<b>point</b>	
<b>pointAmount</b>	
<b>totalPoint</b>	Total world points of the card

**totalPointAmount**

Total world points equivalent of TL

## Unconjugated Returns Transaction

This type of transaction allows the amount in it to be returned to the cardholder to whom the transaction was sent and withdrawn from the account of the merchant. It does not require a previous order number or reference number and does not need to be consistent with a previous transaction.

Merchants with active OrderId parameters should write return in the transaction field when canceling a return transaction with OrderId and OrderDate. Otherwise, since the order id of the sales and return transactions made on the same day are the same, it will not be understood whether the sale is canceled or the return is canceled and the sales transaction will be tried to be canceled.

### Request Example

```
1. <posnetRequest>
2. <mid>6700530119</mid>
3. <tid>67851794</tid>
4. <tranDateRequired>1</tranDateRequired>
5. <unmatchedreturn>
6. <ccno>4506349116608409</ccno>
7. <expDate>2101</expDate>
8. <currencyCode>YT</currencyCode>
9. <amount>245</amount>
10. <orderId>yy3aaaz89012345a7890123r</orderId>
11. </unmatchedreturn>
12. </posnetRequest>
```

### posnetRequest – return

It is used for the unconjugated return.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.

#### unmatchedreturn

<b>ccno</b>	Credit card number
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>Amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.

<b>orderID</b>	Order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. If a financialized transaction with 3D Secure payment management is being refunded, it is necessary to complete 24 digits by putting " <b>TDSC</b> " 3in front of the orderId with 20 digits. For example: TDSCYKB_0000190526121122. However, this method is worse in terms of performance than the use of hostlogkey. <b>If Hostlogkey is used, this field should not be included in xml.</b>
----------------	--

*Response Example*

```

16. <posnetResponse>
17.   <approved>1</approved>
18.   <hostlogkey>019799179990000191</hostlogkey>
19.   <authCode>991799</authCode>
20.   <instInfo>
21.     <inst1>00</inst1>
22.     <amnt1>000000000000</amnt1>
23.   </instInfo>
24.   <pointInfo>
25.     <point>00000000</point>
26.     <pointAmount>000000000000</pointAmount>
27.     <totalPoint>09856296</totalPoint>
28.     <totalPointAmount>000004928148</totalPointAmount>
29.   </pointInfo>
30. </posnetResponse>

```

### posnetResponse – instInfo – pointInfo

As a result of the Unconjugated Return transaction, installment amount and points information are accessed.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code. It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.

#### instInfo

<b>inst1</b>	İşlemin kaç taksite bölündüğünü gösterir. Ör: 00 veya 03
<b>amnt1</b>	Taksit tutarı - Kuruş cinsinden Ör : 12.34 TL için 1234 olarak yer alır.

#### pointInfo

<b>point</b>	
<b>pointAmount</b>	
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

## Transactions of Points

### Points Inquiry

This is used to display the world points of a World Card. It does not appear on the card statement and card page.

#### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <pointInquiry>
6.     <ccno>4048097006508842</ccno>
7.     <expDate>2002</expDate>
8.   </pointInquiry>

```

9. `</posnetRequest>`

<b>posnetRequest – pointInquiry</b>	
It is used to inquiry the World points.	
<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <code>&lt;%MERCHANT_ID%&gt;</code>
<b>tid</b>	YKB Merchant Terminal Number <code>&lt;%TERMINAL_ID%&gt;</code>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.
<b>pointInquiry</b>	
<b>ccno</b>	Credit Card Number
<b>expDate</b>	Credit card expiry date - In the following format: YY MM

#### Response Example

```
1. <posnetResponse>
2.   <approved>1</approved>
3.   <pointInfo>
4.     <point>019985843</point>
5.     <pointAmount>000009992921</pointAmount>
6.     <totalPoint>000000000</totalPoint>
7.     <totalPointAmount>00000000000</totalPointAmount>
8.   </pointInfo>
9. </posnetResponse>
```

<b>posnetResponse – pointInfo</b>	
Points information is accessed at the end of points inquiry transaction.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>pointInfo</b>	
<b>point</b>	Total world points available on the card
<b>pointAmount</b>	The total amount of world points available on the card as equivalent in TL Ex: 1234 for 12.34 TL
<b>totalPoint</b>	N/A
<b>totalPointAmount</b>	N/A

#### Points Usage

This is done to use the worldpoints of a WorldCard. The transaction appears in the "Your WorldPoint Information" section of the card statement. The use of points can also be realized by making mixed transactions.

By using Mixed Transaction instead of using points, 0 (zero) can be sent to amount field and desired points can be sent to point field. Please see: Mixed Transactions

#### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <pointUsage>
6.     <amount>175</amount>
7.     <ccno>4048097006508842</ccno>
8.     <currencyCode>TL</currencyCode>
9.     <expDate>2002</expDate>
10.    <orderID>YKB_TST_190609132900_024</orderID>
11.  </pointUsage>
12. </posnetRequest>

```

#### posnetRequest – pointUsage

It is used to use the World points.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>

#### pointUsage

<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>ccno</b>	Credit Card Number
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>orderID</b>	Order number In cases where the OrderID parameter of the merchant is activated, minimum 1 and maximum 24 digits must be sent.

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019959713990000191</hostlogkey>
4.   <pointInfo>
5.     <point>000000350</point>
6.     <pointAmount>00000000175</pointAmount>
7.     <totalPoint>019985493</totalPoint>
8.     <totalPointAmount>00000992746</totalPointAmount>
9.   </pointInfo>
10. </posnetResponse>

```

#### posnetResponse – pointInfo

At the end of the point usage transaction, the point information is accessed.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.



<b>hostlogkey</b>	The unique Id on the system. It should be kept for refund/cancellation transactions.
<b>pointInfo</b>	
<b>point</b>	World points used in the transaction
<b>pointAmount</b>	The total amount of world points used on the transaction as equivalent in TL Ex: 1234 for 12.34 TL
<b>totalPoint</b>	Remaining world points of the card
<b>totalPointAmount</b>	The total amount of remaining world points on the card as equivalent in TL Ex: 1234 for 12.34 TL

### Points Return

This transaction refunds all or only a certain amount of Points transaction made performed previously. It appears as a separate transaction on the card statement and does not delete the record of the returned transaction on the statement.

For Point Return, the amount to be refunded cannot exceed the original transaction amount and the sum of the amounts previously made on the same original transaction. For example, if you refunded 3 TL in the first return of a 10 TL Point Usage transaction, you can refund a maximum of 7 TL in the second refund. Refunds can be processed as long as the total amount of the returns for a transaction does not reach the amount of the original transaction.

It is also possible to return Points Usage transactions on the day they are performed. Thus, unlike cancellations, only part of the transaction can be refunded.

### Request Example

```

15. <?xml version="1.0" encoding="ISO-8859-9"?>
16. <posnetRequest>
17.   <mid>6706598320</mid>
18.   <tid>67005551</tid>
19.   <tranDateRequired>1</tranDateRequired>
20.   <pointReturn>
21.     <amount>100</amount>
22.     <currencyCode>TL</currencyCode>
23.     <hostLogKey>019676067890000191</hostLogKey>
24.     <!--
25.       <orderID></orderID>
26.     -->
27.   </pointReturn>
28. </posnetRequest>

```

<b>posnetRequest – return</b>	
Used to refund the transaction.	
<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.
<b>pointReturn</b>	
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"

<b>hostlogkey</b>	The unique Id on the system. For the relevant transaction, it is used as returned from the service.
<b>orderID</b>	Shopping order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. If a financialized transaction with 3D Secure payment management is being refunded, it is necessary to complete 24 digits by putting "TDSC" in front of the orderId with 20 digits. For example: TDSCYKB_0000190526121122 However, this method is worse in terms of performance than the use of hostlogkey. If Hostlogkey is used, this field should not be included in xml.

#### Response Sample

```

31. <posnetResponse>
32.   <approved>1</approved>
33.   <hostlogkey>019799179990000191</hostlogkey>
34.   <pointInfo>
35.     <point>00000000</point>
36.     <pointAmount>000000000000</pointAmount>
37.     <totalPoint>09856296</totalPoint>
38.     <totalPointAmount>000004928148</totalPointAmount>
39.   </pointInfo>
40. </posnetResponse>

```

<b>posnetResponse – instInfo – pointInfo</b>	
As a result of the refund transaction, the installment and points information are accessed.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code  It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>pointInfo</b>	
<b>point</b>	
<b>pointAmount</b>	
<b>totalPoint</b>	Total World points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

#### Cancellation of Points Usage

This operation is used to cancel the point usage transaction. The transaction will appear in the "Your World Point Information" section of the card statement as a separate transaction from the use of points.

## Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <reverse>
6.     <transaction>pointUsage</transaction>
7.     <hostLogKey>019959715690000191</hostLogKey>
8.   </reverse>
9. </posnetRequest>
```

### posnetRequest – reverse

It is used for cancellation (reverse) within the day of the transaction.

### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>

### reverse

<b>transaction</b>	The type of operation to be canceled is set in this field. Can be used as cancellation of sales, cancellation of provision, cancellation of financialization, cancellation of point usage, cancellation of VFT transaction, cancellation of refund. Sales: sale Provision: auth Financialization: capt Points Usage: pointUsage VFT Transaction: vftTransaction Return Transaction: return
<b>hostLogKey</b>	The unique Id on the system. For the relevant transaction, it is used as returned from the service.
<b>orderID</b>	Order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. However, this method is worse in terms of performance than the use of hostlogkey. If Hostlogkey is used, this field should not be included in xml.

## Response Example

```
1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019959715690000191</hostlogkey>
4.   <authCode>000000</authCode>
5.   <pointInfo>
6.     <totalPoint>19985493</totalPoint>
7.     <totalPointAmount>9992746</totalPointAmount>
8.   </pointInfo>
9. </posnetResponse>
```

### posnetResponse – pointInfo

At the end of point usage cancellation, the points information is accessed.

### posnetResponse

<b>approved</b>	Transaction result.
-----------------	---------------------

	0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.
<b>pointInfo</b>	
<b>totalPoint</b>	Remaining world points of the card
<b>totalPointAmount</b>	The total amount of remaining world points on the card as equivalent in TL Ex: 1234 for 12.34 TL

## Mixed Transactions

Mixed transactions provide both cash sales or sales in installments sales and allow the use of the points in the same transaction.

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <tranDateRequired>1</tranDateRequired>
6.   <saleWP>
7.     <amount>175</amount>
8.     <wpAmount>25</wpAmount>
9.     <ccno>4048097007190236</ccno>
10.    <currencyCode>TL</currencyCode>
11.    <cvc>000</cvc>
12.    <expDate>2002</expDate>
13.    <orderID>YKB_TST_190612020600_024</orderID>
14.    <installment>00</installment>
15.  </saleWP>
16. </posnetRequest>

```

### posnetRequest – saleWP

It is used for mixed transaction. In the request example above, it was requested to withdraw 1,75 TL from the card account and deduct TL 0,25 from world points to complete a transaction of 2 TL.

### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.

### saleWP

<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>wpAmount</b>	The total amount of world points used on the transaction as equivalent in TL Ex: 1234 should be set for 12.34 TL
<b>ccno</b>	Credit Card Number
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>cvc</b>	Credit card security number - CVV2

<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>orderID</b>	Order number. 24-digits alphanumeric
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>koiCode</b>	Joker Vadaa campaign type (customized transaction code). It should be used by querying the personal transaction list by questioning by card number. It is optional. If a value is not set, it should not be contained in xml. 1: Additional Installment 2: Postponing Installments 3: Extra Points 4: Gaining minutes 5: Postponing Statement 6: Special Delay Interest

### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019842038190000191</hostlogkey>
4.   <authCode>420381</authCode>
5.   <tranDate>190612020835</tranDate>
6.   <instInfo>
7.     <inst1>00
8.     <amnt1>000000000000</amnt1>
9.   </instInfo>
10.  <pointInfo>
11.    <point>00000000</point>
12.    <pointAmount>000000000000</pointAmount>
13.    <totalPoint>20028193</totalPoint>
14.    <totalPointAmount>000010014096</totalPointAmount>
15.  </pointInfo>
16. </posnetResponse>

```

### posnetResponse – instInfo – pointInfo

Hostlogkey must be recorded for being used in transactions such as cancellation/refund.

### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system. It should be kept for refund/cancellation transactions.
<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is conducted, it must be kept for refund and cancellation transactions.
<b>tranDate</b>	It is the actual date of operation according to the system setting and the date that will appear on the merchant administrator screens. If tranDateRequired = 1 is entered in the request, it is received. The format is as following: YYMMDDHHMMSS

instInfo	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
pointInfo	
<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

### Cancellation of Mixed Transaction

Cancellation of Mixed Transactions is performed with the same request package as the cancellation of sales transactions.

While canceling, Order ID or HostlogKey information is used in the Mixed transaction as in the Sales transaction. [Sales Cancellation](#)

When the mixed transaction is canceled, both the Use of Points transaction and the Sale transaction within the mixed transaction are canceled.

### Return of Mixed Transaction

Since the mixed transaction includes both Sale and Point Usage transactions, these transactions must be returned separately. The Point Usage transaction in the mixed transaction can be returned by [Point Return](#), and the Sales transaction within the Mixed transaction can be returned with the [Sales Returns](#) transaction.

## Transactions with Delay Interest (VFT)

It is the transaction of lending a credit to the credit card holders by applying interest at a predetermined rate over the cash amount of the product sold in the desired term (the number of delay can be up to 36 months).

In this type of transaction, the Merchant does not calculate any interest. The merchant enters the selected delay date and the product sales amount to the system. The Bank gives credit to the cardholder at the interest rate of consumer credit determined on the sales amount and the number of delays. The bank transfers the sales amount to the merchant account in accordance with agreement between the merchant and the bank.

This transaction is defined as cash sales for the merchant and credit sales for the bank.

This transaction should not be confused with the Sales in Installments at Cash Price. The amount and the number of installments specified in the Sales in Installment at Cash Price transaction is paid to the merchant in accordance with the agreement between the merchant and the bank. In the sales with Delay Interest, the transaction is considered to be cash for the merchant.

Although the interest rate applied to the VFT transactions is fixed, the interest is calculated daily. In this case, the interest calculation of the cardholder according to the statement date may vary. The card with less time remaining until the issuance of the statement will have less interest payment and the cardholder with more time remaining until the issuance of the statement will pay more interest.

For example, in VFT transactions made with a card, whose statement is issued on the 22nd day of the month and whose statement is issued on the 12th day of the month, total amount (transaction amount + interest) will be

different although the transaction has the same amount. There will be more interest on the card, whose statement is issued on the 22nd day of the month.

In order to inform the customers about the interest amounts to be applied to the VFT transactions, it may be appropriate for Posnet companies to show the interest amount to be applied by making a VFT query to their customers and to make the VFT sales transaction after the customer approval.

### Delay Interest Inquiry

It shows the Payment table according to the campaign selected for a transaction with the delay interest. It has no financial value. The Payment Table is valid for the day of the transaction. It does not appear on the card statement and card page.

#### Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <vftQuery>
6.     <ccno>4506344131339098</ccno>
7.     <amount>175</amount>
8.     <installment>03</installment>
9.     <vftCode>K001</vftCode>
10.  </vftQuery>
11. </posnetRequest>
```

#### posnetRequest – vftQuery

It is used for the delay interest inquiry transaction.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>

#### vftQuery

<b>ccno</b>	Credit Card Number
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>vftCode</b>	VFT Campaign code. It can be obtained by using the link of merchant information in the merchant administrator screens. The information in the Campaign Code column under VFT Campaign Details should be entered here. K001 is used for the test environment.

#### Response Example

```
1. <posnetResponse>
2.   <approved>1</approved>
3.   <instInfo>
4.     <inst1>03
5.     <amnt1>000000000059</amnt1>
6.   </instInfo>
7.   <vftInfo>
8.     <vftAmount>00000000002</vftAmount>
9.     <vftRate>000223</vftRate>
10.    <vftDayCount>0001</vftDayCount>
11.  </vftInfo>
```

12. </posnetResponse>

<b>posnetResponse – instInfo - vftInfo</b>	
Payment information and delay interest information for the installment option specified after the inquiry are accessed here.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>vftInfo</b>	
<b>vftAmount</b>	Delay interest amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>vftRate</b>	Delay interest is thousands in the last three digits Ex: 000223: 0.223%
<b>vftDayCount</b>	Number of delay, the number of days remaining to the first statement of the credit card after the transaction

## Sales in Installment with Delay Interest

### Request Example

It is the sales transaction made by applying the delay interest. vftCode is obtained from the administrators screens of the merchant. If you want to make transactions with a delay interest, you should include the koiCode field. For koiCode query and transactions, please see: Personal - Joker Vadaa Transactions.

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <vftTransaction>
6.     <ccno>4506344131339098</ccno>
7.     <cvc>000</cvc>
8.     <expDate>2002</expDate>
9.     <amount>175</amount>
10.    <currencyCode>TL</currencyCode>
11.    <installment>03</installment>
12.    <vftCode>K001</vftCode>
13.    <orderId>YKB_TST_190610234500_024</orderId>
14.    <!--
15.     <koiCode>1</koiCode>
16.    -->
17.  </vftTransaction>
18. </posnetRequest>

```

<b>posnetRequest – vftTransaction</b>	
It is used for sale with delay interest.	
<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>



<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>vftTransaction</b>	
<b>ccno</b>	Credit Card Number
<b>cvc</b>	Credit card security number - CVV2
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>vftCode</b>	VFT Campaign code. It can be obtained by using the link of merchant information in the merchant administrator screens. The information in the Campaign Code column under VFT Campaign Details should be entered here. K001 is used for the test environment.
<b>orderID</b>	Order number. 24-digits alphanumeric. In cases where the OrderID parameter of the merchant is activated, minimum 1 and maximum 24 digits must be sent.
<b>koiCode</b>	Joker Vadaa campaign type (customized transaction code). It should be used by querying the personal transaction list by questioning by card number. It is optional. If a value is not set, it should not be contained in xml. 1: Additional Installment 2: Postponing Installments 3: Extra Points 4: Gaining minutes 5: Postponing Statement 6: Special Delay Interest

### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>01996002229000191</hostlogkey>
4.   <authCode>600222</authCode>
5.   <instInfo>
6.     <inst1>03
7.     <amnt1>59</amnt1>
8.   </instInfo>
9.   <pointInfo>
10.    <point>0</point>
11.    <pointAmount>0</pointAmount>
12.    <totalPoint>9432912</totalPoint>
13.    <totalPointAmount>4716456</totalPointAmount>
14.  </pointInfo>
15.  <vftInfo>
16.    <vftAmount>2</vftAmount>
17.    <vftRate>223</vftRate>
18.    <vftDayCount>1</vftDayCount>
19.  </vftInfo>
20. </posnetResponse>

```

### posnetResponse – instInfo – pointInfo – vftInfo

Payment information, points information, delay interest information for the installment option specified after the transaction are accessed here.

### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful
-----------------	--

	1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system. It should be kept for refund/cancellation transactions.
<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is conducted, it must be kept for refund and cancellation transactions.
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	
<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL
<b>vftInfo</b>	
<b>vftAmount</b>	Delay interest amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>vftRate</b>	In delay interest rate, the last three digits are in thousands Ex: 000223: 0.223%
<b>vftDayCount</b>	Number of delay, the number of days remaining to the first statement of the credit card after the transaction

## Return of Sales with Delay Interest

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <vftReturn>
6.     <hostLogKey>019960027090000191</hostLogKey>
7.     <authCode>600270</authCode>
8.     <amount>175</amount>
9.     <currencyCode>TL</currencyCode>
10.    <!--
11.     <orderID>YKB_TST_190611004500_024</orderID>
12.    -->
13.   </vftReturn>
14. </posnetRequest>

```

### posnetRequest – vftReturn

Is used to return the Vft transaction. It can be used for closed as a group transactions that are not realized on the same day. Cancellation can be made within the same day. See: Credit Card Transactions, Cancellation transaction

### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>vftReturn</b>	

<b>hostlogkey</b>	The unique Id on the system. For the relevant transaction, it is used as returned from the service.
<b>authCode</b>	It is the authorization code generated based on the system authorization. Transaction with delay interest (VFT) is used as returned from the service. This field is required for VFT transaction cancellation.
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>orderID</b>	Order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. If a financialized transaction with 3D Secure payment management is being refunded, it is necessary to complete 24 digits by putting "TDSC" in front of the orderId with 20 digits. Example: TDSCYKB_0000190526121122 This method is worse in terms of performance than the use of hostlogkey. <b>If Hostlogkey is used, this field should not be included in xml.</b> Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>orderDate</b>	In cases where the OrderID parameter of the merchant is activated, the order date of the original transaction should be sent in this field for the singularity control of the transaction to be financialized with orderID. The format must be YYYYMMDD.

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019960027090000191</hostlogkey>
4.   <authCode>600270</authCode>
5.   <instInfo>
6.     <inst1>00
7.     <amnt1>000000000000</amnt1>
8.   </instInfo>
9.   <pointInfo>
10.    <point>00000000</point>
11.    <pointAmount>0000000000</pointAmount>
12.    <totalPoint>09856296</totalPoint>
13.    <totalPointAmount>000004928148</totalPointAmount>
14.  </pointInfo>
15. </posnetResponse>

```

#### posnetResponse – instInfo – pointInfo

At the end of return transaction, it is possible to access the number of installments and point information.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.
<b>instInfo</b>	

<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	
<b>point</b>	

<b>pointAmount</b>	
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

## Personalized - Joker Vadaa Transactions

Personalized Transaction (PT in short) is a form of transaction that allows World card owners to decide the number of points they will receive at the end of their purchases and the number of installments. It is also called Joker Vadaa.

In POSNET, Joker Vadaa is the award given to YKB credit card holders within the scope of the campaigns organized by YKB. These awards vary according to credit card holder, transaction type and merchant. Hence, the credit card holder can use different awards depending on the type of shopping and the campaign. These awards are as follows.

1. Additional Installment
2. Postponing Installment
3. Extra points
4. Gaining minutes
5. Postponing Statement
6. Special Delay Interest

### Using Joker Vadaa in Posnet

In order to use Joker Vadaa in POSNET, first of all, it is necessary to query the Joker Vadaa that the cardholder can use; secondly, to send the campaign code selected by the credit card owner and obtained by the inquiry to the relevant transaction (Provision, Sales or Sales with Delay Interest).

### Points need to be considered

In Posnet, Joker Vadaa transactions can only be performed in Provision, Sales, Installment and Sales with Delay Interest transactions.

Posnet can be used on Joker Vadaa, YKB and World License cards.

During the financialization of a Provision transaction using Joker Vadaa in Posnet, installment and amount information cannot be changed.

Postponement of an Installment and Additional Installment transactions among the awards are only available for transactions with installments.

Special Delay Interest from Awards can only be used in Sales with Delay Interest.

### Personalized Transaction Inquiry

Each credit card can have different rewards (maximum 8). Hence, it is necessary to show these awards to the user in order to select the awards that the user can use. To do this, you need to use Personalized Transaction - Joker Vadaa Inquiry Service. This service takes only credit card number as a parameter and shows the related award(s). Each obtained award contains a campaign code and a campaign message. The campaign code is a parameter that must be submitted when performing a Joker Vadaa operation (Provision, Sales, or Sales with Delay Interest). The campaign message is for informational purposes only.

If a Joker Vadaa transaction is desired, the campaign code of the award will be set (a value between 1 and 6) on koiCode field to the service and must be sent.

If you do not want to use Joker Vadaa in this transaction, this field should not be included in the service pattern.

If transactions are made through the 3DS or the bank Common Payment Page, the user will be given the chance to select joker vadaa campaigns by showing the joker vadaa campaigns on the pages provided by the bank.

#### Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <koiCampaignQuery>
6.     <ccno>4048097006508842</ccno>
7.   </koiCampaignQuery>
8. </posnetRequest>
```

#### posnetRequest – koiCampaignQuery

It is used to query personalized campaigns (Joker Vadaa)

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>

#### koiCampaignQuery

<b>ccno</b>	Credit Card Number
-------------	--------------------

#### Response Example

```
1. <posnetResponse>
2.   <approved>1</approved>
3.   <koiInfo>
4.     <code>1</code>
5.     <message>Sana1 Pos +2 ek taksit</message>
6.   </koiInfo>
7.   <koiInfo>
8.     <code>2</code>
9.     <message>Tum isyerleri 5 ay erteleme</message>
10.  </koiInfo>
11.  <koiInfo>
12.    <code>3</code>
13.    <message>Ekstra Puan Kazan</message>
14.  </koiInfo>
15. </posnetResponse>
```

#### posnetResponse – koiInfo

After the query, the specified joker vadaa campaign information is accessed.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.

#### koiInfo

<b>code</b>	Specifies the code for the Personalized Transaction (Joker Vadaa). Multiple types of personalized transaction can be realized with one card. 1: Additional Installment 2: Postponing Installment 3: Extra Points 4: Gaining minutes 5: Postponing Statement 6: Special Delay Interest
<b>message</b>	It contains the message to be shown to the cardholder together with the personalized transaction. This message is for informational purposes only and may vary by cardholder.

In the test environment, the messages to be returned on personalized transactions inquiries may be meaningless. This is normal, meaningful messages will be returned to your customers in the live environment.

At the end of the personalized transaction inquiry, more than one message (list of personalized transaction) with the same code can be returned;

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <koiInfo>
4.     <code>1</code>
5.     <message>Sana1 Pos +2 ek taksit</message>
6.   </koiInfo>
7.   <koiInfo>
8.     <code>1</code>
9.     <message>300TL uzeri +3 ek taksit</message>
10.  </koiInfo>
11.  <koiInfo>
12.    <code>2</code>
13.    <message>Tum isyerleri 5 ay erteleme</message>
14.  </koiInfo>
15. </posnetResponse>

```

The reason why the personalized transaction receives 2 different messages but the same personalized transaction code (1: Additional Installment) is that two different installment campaigns are defined for the customer. In this case, it is okay to show your customer both options. The purpose of this design is to enable the customer to see all defined campaigns. As the merchant, all you have to do is to show your customer all the options (all messages returned at the end of the inquiry). In fact, in the above example, no matter which option your customer chooses, if the transaction is over 300 TL, three additional installments will be offered. In short, the system automatically selects the most advantageous personalized transaction for the customer; regardless of which personalized transaction of the same code is selected. The object is to prevent the customer from accidentally selecting a less advantageous personalized transaction.

You only need to show your customer the messages returned to you at the end of the personalized transaction query. These messages include information about which code they belong to, such as additional installments or extra points.

## Trio Transactions

These are the transactions with a completely flexible delay structure to be used in purchases where buyers and sellers are commercial enterprises. Please see: <https://www.yapikredi.com.tr/kartlar/ticari-kart>

## Single Payment

### Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6797752273</mid>
4.   <tid>67003715</tid>
5.   <!--
6.     noWarranty parametresi trio isleminin garantisiz olup olmadigini belirtir. Girilmezse false
7.     kabul edilir.
8.   -->
9.   <trioSingle noWarranty="false">
10.    <ccno>6037978246464011</ccno>
11.    <expDate>2511</expDate>
12.    <cvc>000</cvc>
13.    <orderID>YKB_TST_190703093100_024</orderID>
14.    <amount>175</amount>
15.    <currencyCode>TL</currencyCode>
16.    <termDayCount>20</termDayCount>
17.  </trioSingle>
18.</posnetRequest>
```

#### posnetRequest – trioSingle

It is used for Trio single payment.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>

#### trioSingle

<b>ccno</b>	Credit Card Number
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>cvc</b>	Credit card security number - CVV2
<b>orderID</b>	Order number. 24-digits alphanumeric. Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>termDayCount</b>	Trio payment term count

### Response Example

```
1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>020527826090000191</hostlogkey>
4.   <authCode>011970</authCode>
5. </posnetResponse>
```

#### posnetResponse

Hostlogkey must be recorded for being used in transactions such as cancellation/refund.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.



<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.

## Fixed payment

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6797752273</mid>
4.   <tid>67003715</tid>
5.   <!--
6.     noWarranty parametresi trio isleminin garantisiz olup olmadigini belirtir. Girilmezse false
       kabul edilir.
7.   -->
8.   <trioFixed nowarranty="false">
9.     <ccno>60379782464011</ccno>
10.    <expDate>2511</expDate>
11.    <cvc>000</cvc>
12.    <orderID>YKB_TST_190706021900_025</orderID>
13.    <amount>175</amount>
14.    <currencyCode>TL</currencyCode>
15.  </trioFixed>
16. </posnetRequest>

```

### posnetRequest – trioFixed

It is used for Trio fixed payment.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>trioFixed</b>	
<b>ccno</b>	Credit Card Number
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>cvc</b>	Credit card security number - CVV2
<b>orderID</b>	Order number. 24-digits alphanumeric. Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"

### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>020527826090000192</hostlogkey>
4.   <authCode>011971</authCode>
5. </posnetResponse>

```

### posnetResponse

Hostlogkey must be recorded for being used in transactions such as cancellation/refund.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
-----------------	---

<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.

## Multi-Payment

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6797752273</mid>
4.   <tid>67003715</tid>
5.   <!--
6.     noWarranty parametresi trio isleminin garantisiz olup olmadigini belirtir. Girilmezse false
7.     kabul edilir.
8.   -->
9.   <trioMultiple noWarranty="false">
10.    <ccno>6037978246464011</ccno>
11.    <expDate>2511</expDate>
12.    <cvc>000</cvc>
13.    <orderID>YKB_TST_190707121900_024</orderID>
14.    <amount>1000</amount>
15.    <currencyCode>TL</currencyCode>
16.    <installment>03</installment>
17.    <termDayCount>20</termDayCount>
18.  </trioMultiple>
19. </posnetRequest>

```

### posnetRequest – trioMultiple

It is used for Trio multiple payments.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>trioMultiple</b>	
<b>ccno</b>	Credit Card Number
<b>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>cvc</b>	Credit card security number - CVV2
<b>orderID</b>	Order number. 24-digits alphanumeric. Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>installment</b>	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
<b>termDayCount</b>	Trio payment term count

### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>018971217390000181</hostlogkey>
4.   <authCode>010680</authCode>

```

```

5.     <instInfo>
6.         <inst1>03
7.         <amnt1>00000000333</amnt1>
8.     </instInfo>
9. </posnetResponse>

```

#### posnetResponse – instInfo

Hostlogkey must be recorded for being used in transactions such as cancellation/refund.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.

#### Return

#### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.     <mid>6797752273</mid>
4.     <tid>67003715</tid>
5.     <trioReturn>
6.         <hostLogKey>02052782609000191</hostLogKey>
7.         <!--
8.         <orderID>YKB_TST_190703093100_024<orderID>
9.         -->
10.    </trioReturn>
11. </posnetRequest>

```

#### posnetRequest – trioReturn

It is used to return the Trio transaction. It works with hostLogKey or orderId. The entire transaction amount can be returned.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>trioReturn</b>	
<b>hostlogkey</b>	The unique Id on the system. For the relevant transaction, it is used as returned from the service.
<b>orderId</b>	Order number. It is optional. If you do not keep the hostlogkey on your system, you can also cancel the transaction using the order number of the original transaction. <b>If Hostlogkey is used, this field should not be included in xml.</b> In cases where the OrderID parameter of the merchant is activated, minimum 1 and maximum 24 digits must be sent.

**orderDate**

In cases where the OrderID parameter of the merchant is activated, the order date of the original transaction should be sent in this field for the singularity control of the transaction to be cancelled with orderID. The format must be YYYYMMDD.

## Response Example

```
1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>020527826090000191</hostlogkey>
4.   <authCode>011970</authCode>
5.   <!-- coklu odeme islemleri icin yer alacaktır
6.   <instInfo><inst1>01</instInfo>
7.   -->
8.   <trioDate></trioDate>
9.   <trioAmount>00000000175</trioAmount>
10.  <ins1>
11.  <amt1></amt1>
12. </posnetResponse>
```

### posnetResponse – instInfo

At the end of return transaction, it is possible to access the number of installments and point information.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system.
<b>authCode</b>	It is the authorization code generated based on the system authorization.
<b>trioDate</b>	Trio payment date
<b>trioAmount</b>	Transaction amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>ins1</b>	
<b>amt1</b>	
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.

## Limit Inquiry

### Request Example

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6797752273</mid>
4.   <tid>67003715</tid>
5.   <trioAvailableLimitInq>
6.     <ccno>6037978246464011</ccno>
7.     <expDate>2511</expDate>
8.   </trioAvailableLimitInq>
9. </posnetRequest>
```

### posnetRequest – trioAvailableLimitInq

It is used to query the remaining limit of the card.

<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>trioAvailableLimitInq</b>	
<b>ccno</b>	Credit Card Number
<b>expDate</b>	Credit card expiry date - In the following format: YY MM

### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <limitInfo>
4.     <accountNumber>42049272</accountNumber>
5.     <availableAmount>000000000063 2347</availableAmount>
6.     <limit>0000000000800000</limit>
7.   </limitInfo>
8. </posnetResponse>

```

<b>posnetResponse – limitInfo</b>	
The remaining limit information of the card is accessed.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>limitInfo</b>	
<b>accountNumber</b>	Account number
<b>availableAmount</b>	Available amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>limit</b>	Limit

## Reconciliation Services

### Transaction Status Inquiry (Agreement)

If an error occurs during the data transfer between the bank servers and the merchant servers, the result of the transaction may not be visible to the merchant. This problem can be caused by systems failing to respond within the expected time or packet loss at the network level. In cases where the transaction is financialized or blocked the customer's card balance but the transaction does not get the end result, the result should be checked again with the inquiry service.

On the banks side, it is possible to connect with different banks or systems and wait up to 45 seconds for the transaction period. For this reason, it is recommended to use 1-minute timeout period in merchant systems.

If no response can be received from the bank system to a request for provision request sent by the merchant, it cannot be determined by the merchant whether the transaction has taken place. If the transaction is successfully completed by the bank, the card will be blocked the amount of the transaction and displayed by the customer in the card transactions. In this case, instead of performing a second provision transaction, the workplace should check the result of the transaction by inquiry. This should be taken into account in particular for

transactions that result in financial consequences (sales, use of points, etc.) or may be viewed by the user as a card transaction (provision). In order to prevent any problem, in the second transactions with the same order number, the bank system will perform singularity check to prevent duplicate transactions and return the appropriate message.

#### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <agreement>
6.     <orderID>YKB_TST_1905210122001234</orderID>
7.   </agreement>
8. </posnetRequest>

```

<b>posnetRequest – agreement</b>	
It is used for inquiry the transaction result with the order number.	
<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>agreement</b>	
<b>orderID</b>	Order number
<b>OrderDate</b>	When the OrderID parameters of the merchant become active The order date of the transaction to be queried must also be entered. The format must be YYYYMMDD.

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <transactions>
4.     <transaction>
5.       <orderID>YKB_TST_1905210122001234</orderID>
6.       <ccno>4048 09** **** *842</ccno>
7.       <amount>1,75</amount>
8.       <currencyCode>TL</currencyCode>
9.       <authCode>177500</authCode>
10.      <tranDate>2019-05-21 01:28:44.71</tranDate>
11.      <state>Authorization</state>
12.    </transaction>
13.  </transactions>
14. </posnetResponse>

```

<b>posnetResponse – transactions – transaction</b>	
The query results are reached after the query.	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.

transactions – transaction



<b>orderID</b>	Order number. 24-digits alphanumeric. Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>ccno</b>	Credit card number. The first 6 and last 4 digits are open while the other fields are masked.
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>authCode</b>	It is the authorization code generated based on the system authorization.
<b>tranDate</b>	It is the actual date of operation according to the system setting and the date that will appear on the merchant administrator screens. It is in the format of yyyy-MM-dd HH:mm:ss.nnn.
<b>state</b>	Indicates the type of transaction. Sale: Sales and Sales with Installment Authorization: Provision and Provision with Installment Bonus_Usage: Points Usage Sale_Reverse: Sales Cancellation and Sales Cancellation with Installment Return: Refund:
<b>Hostlogkey</b>	Single ID in the system.
<b>txnStatus</b>	Shows the success of the transaction. 1: Successful transaction (transaction amount will be reflected to merchant and cardholder) 0: Unsuccessful or canceled transaction (transaction amount depends on merchant and cardholder will not be reflected)

## Daily Transaction Report Inquiry

### Request Example

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6797752273</mid>
4.   <tid>67345854</tid>
5.   <merchantReconciliation>
6.     <startDate>2019-06-12</startDate>
7.     <endDate>2019-06-14</endDate>
8.     <currency>TL</currency>
9.   </merchantReconciliation>
10. </posnetRequest>

```

### posnetRequest – merchantReconciliation

It is used to inquiry the total number of transactions and the amount of transactions realized by the merchant on a given day.

#### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>

#### merchantReconciliation

<b>startDate</b>	It is received in YYYY-MM-DD format and specifies the start date of the requested data. The specified start date is included in the report. Ex: 2019-02-01
<b>endDate</b>	It is received in YYYY-MM-DD format and specifies the end date of the requested data. The specified end date is included in the report. The difference between the start and end dates can be 10 days at maximum. If the start and end dates are specified as the same day, a 1-day transaction

	report is issued. Ex: 2019-02-01
<b>currency</b>	Currency - "TL, US, EU"

### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <merchantReconciliation>
4.     <totalNumber>5</totalNumber>
5.     <totalAmount>50000</totalAmount>
6.     <transactionType>1</transactionType>
7.     <transactionName>Peşin Satış</transactionName>
8.   </merchantReconciliation>
9.   <merchantReconciliation>
10.    <totalNumber>2</totalNumber>
11.    <totalAmount>4000</totalAmount>
12.    <transactionType>2</transactionType>
13.    <transactionName>Taksitli Satış</transactionName>
14.  </merchantReconciliation>
15.  <merchantReconciliation>
16.    <totalNumber>3</totalNumber>
17.    <totalAmount>38107</totalAmount>
18.    <transactionType>6</transactionType>
19.    <transactionName>Peşin Satış İptal</transactionName>
20.  </merchantReconciliation>
21. </posnetResponse>

```

### posnetResponse – merchantReconciliation

For each type of transaction, the information on the number of transaction and total amount is provided.

#### posnetResponse

<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.

#### merchantReconciliation

<b>totalNumber</b>	Total number of transactions of the related transaction type
<b>totalAmount</b>	The total amount of the transaction in the relevant transaction type - In Kurus: Ex: 12.34 TL should be set as 1234.
<b>transactionType</b>	This is transaction type. Corresponds to the following values. 0: Transaction type undefined 1: Cash Sales 2: Sales with Installment 3: Cash Refund 4: Sales with Installment Refund 5: Financialization 6: Cash Sales Cancellation 7: Sales with Installments Cancellation 8: Financialization with Installment 9: Sales with Delay Interest 10: Sales with Points Points Usage

	11: Cash Sales Cancellation - Systemic 12: Sales with Installments with Points 13: Cash Sales with Points 14: Sales with Delay Interest Refund 15: Points Usage 16: Sales with Installments Cancellation - Systemic 17: Financialization Cancellation 18: Sales with Delay Interest Cancellation 19: Points Usage Refund 20: Financialization with Installment Cancellation 21: Sales with Installment Refund Cancellation 22: Cash Sales Refund Cancellation 23: Cash Sales - FTP 24: Sales with Installment - FTP 25: Cash Sales Cancellation - FTP 26: Sales with Installment Refund - FTP 27: Points Usage Cancellation 28: Sales with Points Points Usage Cancellation 29: Cash Sales with Points Cancellation 30: Cash Sales Cancellation - FTP 31: Sales with Installments Cancellation - FTP 32: Sales with Installments with Points Cancellation 33: Provision Cancellation 34: Provision 35: Joker Vadaa Inquiry 36: Points Inquiry
<b>transactionName</b>	This is the name of the transaction. Ex: Provision, Cash Sales, Cash Sales Cancellation, Sales with Installment, Sales with Delay Interest, Financialization, etc.

### Response Example (Incorrect)

```

1. <posnetResponse>
2.   <approved>0</approved>
3.   <respCode>E117</respCode>
4.   <respText>Tarih deęerleri boş olamaz</respText>
5. </posnetResponse>
6.
7. <posnetResponse>
8.   <approved>0</approved>
9.   <respCode>E154</respCode>
10.  <respText>Üye işyeri adı null ya da boş olamaz.</respText>
11. </posnetResponse>
12.
13. <posnetResponse>
14.   <approved>0</approved>
15.   <respCode>E215</respCode>
16.   <respText>Tarih alanı hatalı</respText>
17. </posnetResponse>
18.
19. <posnetResponse>
20.   <approved>0</approved>
21.   <respCode>0340</respCode>
22.   <respText>BAŞLANGIÇ VE BİTİŞ TARİHİ FARKI 10 GÜNÜ GEÇEMEZ.</respText>
23. </posnetResponse>
24.
25. <posnetResponse>

```

```

26. <approved>0</approved>
27. <respCode>0341</respCode>
28. <respText>BAŞLANGIÇ TARİHİ BİTİŞ TARİHİNDEN BÜYÜK OLAMAZ</respText>
29. </posnetResponse>

```

## New Generation Payment

With the change in technology, payment methods have also changed, and there have been cases where payments are made through the mobile-web channel or the transaction is made face-to-face or through the intermediary (systemic) on behalf of the cardholder. In addition, the merchants can store card information in digital wallets. BKM assigns a single value for each merchant and card storage solution (wallet) when applied through banks and merchants can share information stating how the card information is obtained from which channel of payment transactions they receive.

If this information is passed in posnetRequest in all financial services, Posnet stores this information by comparing it with the information defined in the system. New generation payment information, on the example below can be included in Posnet request: <sale>, Puan Kullanım <pointUsage>, Provizyon <auth>, Finansallştırma <capt>, İade <return>, Eşleniksiz İade <unmatchedreturn>, İptal <reverse>, VFT Satış <vftTransaction>, VFT İade <vftReturn>, Tekli Trio Ödeme <trioSingle>, Çoklu Trio Ödeme <trioMultiple>, Sabit Trio Ödeme <trioFixed>, 3D Secure İşlemin Finansallştırılması <oosTranData>. In<posnetRequest>, it is necessary to include <ngp> xml tag as childNode.

```

1. <ngp>
2. <txnType>1</txnType>
3. <txnSrcApp>3</txnSrcApp>
4. <assignedId>123456</assignedId>
5. <wProgData></wProgData>
6. <ffi>00</ffi>
7. <vType>1</vType>
8. </ngp>

```

### posnetRequest – ngp

Payment information is included in this field for new generation payment transactions.

#### ngp

<b>txnType</b>	Transaction Type - 2-digit numeric 1: Face to Face Mobile Transactions 2: Non-Face to Face Mobile Transactions 3: Transactions from the Web with Stored Card Data
<b>txnSrcApp</b>	Transaction Resource Application - 2 digit numeric 1: Wallets (Bank, Diagram, 3. Parties) 2: BKM Express 3: Merchant Mobile Applications
<b>assignedId</b>	Individual ID assigned by BKM for each merchant - 6-digits alphanumeric
<b>wProgData</b>	Wallet Program Information (unique ID assigned by BKM for card storage product) - 3-digits alphanumeric
<b>ffi</b>	Form Factor Indicator
<b>vType</b>	Verification Type

## Request Example

### Sales

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67016829</tid>
5.   <sale>
6.     <ccno>4506347022052795</ccno>
7.     <cvc>000</cvc>
8.     <expDate>2002</expDate>
9.     <amount>2000</amount>
10.    <currencyCode>TL</currencyCode>
11.    <installment>4</installment>
12.    <orderID>jokervadaa0sale0000000A1</orderID>
13.    <koiCode>1</koiCode>
14.  </sale>
15.  <ngp>
16.    <txnType>1</txnType>
17.    <txnSrcApp>3</txnSrcApp>
18.    <assignedId>123456</assignedId>
19.    <wProgData></wProgData>
20.    <ffi>00</ffi>
21.    <vType>1</vType>
22.  </ngp>
23. </posnetRequest>
```

### Points Usage

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67011071</tid>
5.   <pointUsage>
6.     <amount>245</amount>
7.     <ccno>4506347022052795</ccno>
8.     <currencyCode>TL</currencyCode>
9.     <expDate>2002</expDate>
10.    <orderID>1s3t56z8a012345673s01234</orderID>
11.  </pointUsage>
12.  <ngp>
13.    <txnType>1</txnType>
14.    <txnSrcApp>3</txnSrcApp>
15.    <assignedId>123456</assignedId>
16.    <wProgData></wProgData>
17.    <ffi>00</ffi>
18.    <vType>1</vType>
19.  </ngp>
20. </posnetRequest>
```

### Provision

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67011071</tid>
5.   <auth>
6.     <ccno>4506347022052795</ccno>
7.     <expDate>2002</expDate>
8.     <cvc>000</cvc>
9.     <currencyCode>TL</currencyCode>
10.    <amount>1000</amount>
11.    <orderID>onprovizyony00000000030</orderID>
12.  </auth>
```

```
13.     <ngp>
14.         <txnType>1</txnType>
15.         <txnSrcApp>3</txnSrcApp>
16.         <assignedId>123456</assignedId>
17.         <wProgData></wProgData>
18.         <ffi>00</ffi>
19.         <vType>1</vType>
20.     </ngp>
21. </posnetRequest>
```

## Financialization

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.     <mid>6706598320</mid>
4.     <tid>67011071</tid>
5.     <capt>
6.         <amount>1000</amount>
7.         <currencyCode>TL</currencyCode>
8.         <hostLogKey>019017272790000181</hostLogKey>
9.     </capt>
10.    <ngp>
11.        <txnType>1</txnType>
12.        <txnSrcApp>3</txnSrcApp>
13.        <assignedId>123456</assignedId>
14.        <wProgData></wProgData>
15.        <ffi>00</ffi>
16.        <vType>1</vType>
17.    </ngp>
18. </posnetRequest>
```

## Return

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.     <mid>6706598320</mid>
4.     <tid>67011071</tid>
5.     <return>
6.         <amount>245</amount>
7.         <currencyCode>TL</currencyCode>
8.         <hostLogKey>019139540590000191</hostLogKey>
9.     </return>
10.    <ngp>
11.        <txnType>1</txnType>
12.        <txnSrcApp>3</txnSrcApp>
13.        <assignedId>123456</assignedId>
14.        <wProgData></wProgData>
15.        <ffi>00</ffi>
16.        <vType>1</vType>
17.    </ngp>
18. </posnetRequest>
```

## Trio payment

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.     <mid>2634273367</mid>
4.     <tid>00293967</tid>
5.     <trioFixed noWarranty="false">
6.         <ccno>6037978199001901</ccno>
7.         <expDate>0907</expDate>
8.         <cvc>000</cvc>
9.         <orderID>1s3456z89012345678901234</orderID>
10.    <amount>2451</amount>
```

```

11.     <currencyCode>TL</currencyCode>
12. </trioFixed>
13. <ngp>
14.     <txnType>1</txnType>
15.     <txnSrcApp>3</txnSrcApp>
16.     <assignedId>123456</assignedId>
17.     <wProgData></wProgData>
18.     <ffi>00</ffi>
19.     <vType>1</vType>
20. </ngp>
21. </posnetRequest>

```

### 3D Secure Payment Financialization

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.     <mid>6706022701</mid>
4.     <tid>67002706</tid>
5.     <oosTranData>
6.         <bankData>87F491ACD24EAE64B519980F0B1BC7547BE4A7C5C614DC3A8CA3FC41B180EE7765851B081AAE61
221956C0C68B0AD69307B4386C7FCE451C272264251BD72BFCBA0A96A197C38C6CD39DD442BC179FF098824AFA15B1BB
320AD15DA2FB588ECC81B11A26D13764A57B57B49C4CA1BD5D46FA7E60EED480C944AE0817</bankData>
7.         <wpAmount>0</wpAmount>
8.         <mac>DF2323A3BMC782QOP42RT</mac>
9.     </oosTranData>
10.    <ngp>
11.        <txnType>1</txnType>
12.        <txnSrcApp>3</txnSrcApp>
13.        <assignedId>123456</assignedId>
14.        <wProgData></wProgData>
15.        <ffi>00</ffi>
16.        <vType>1</vType>
17.    </ngp>
18. </posnetRequest>

```

Transaction conducted with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card

Sales transaction with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card

It is the method used for conducting the transactions without sending the full version of the card information formulated for the use of insurance sector companies. It differentiates itself from the standard transaction by using the first 6 and last 4 digits of the credit card to be processed and the TR ID or Taxpayer Number of the cardholder. Since this type of transaction is regulated for the use of insurance companies, there is a control mechanism applied based on the sector definition of the contracted company. Transactions can be completed with all bank cards by typing the TR ID/Taxpayer Number and the first 6 and last 4 digits of the card. It can be used in e-commerce and mail order transactions.

In this context, sales, sales with installment, pre-authorization (<auth>) and pre-authorization financial (<capt>) transactions can be performed with the cardholder's TR ID No and the first 6 last 4 digits of the card holder. Sample request and response information are provided for the following types of transactions.

All transactions carried out using the TR ID/Taxpayer Number and the 6 and the last 4 digits of the card will be canceled or refunded as is the case for standard transactions, there is no change of policy in the cancellation and return request and response. A sample of cancellation and return request in transactions carried out using the TR ID/Tax Number and the card's first 6

digits and the last 4 digits is provided below.

*Sample Request of the sales transactions carried out with TR Identity No and first 6 and last 4 digits of the card:*

```
1 <posnetRequest>
2   <mid>6700972667</mid>
3   <tid>67510491</tid>
4   <sale>
5     <cvc>586</cvc>
6     <currencyCode>YT</currencyCode>
7     <amount>2323</amount>
8     <orderId>YKB TST 202010234500 020</orderId>
9     <cardInfo>
10      <inquiryValue>41839824368</inquiryValue>
11      <cardNoFirst>540061</cardNoFirst>
12      <cardNoLast>4581</cardNoLast>
13    </cardInfo>
14  </sale>
15 </posnetRequest>
```



Sample Request of the sales transactions carried out with TR Identity No and first 6 and last 4 digits of the card::

```
1. <posnetRequest>
2.   <mid>6700972667</mid>
3.   <tid>67510491</tid>
4.   <sale>
5.     <cvc>586</cvc>
6.     <currencyCode>YT</currencyCode>
7.     <amount>100</amount>
8.     <orderId>YKB TST 202010234500 021</orderId>
9.     <cardInfo>
10.      <inquiryValue>2620171779</inquiryValue>
11.      <cardNoFirst>540061</cardNoFirst>
12.      <cardNoLast>4581</cardNoLast>
13.    </cardInfo>
14.  </sale>
15. </posnetReq
```

Sample Request of Sales in Installments with TR Identity No and first 6 and last 4 digits of the Card:

```
1. <posnetRequest>
2.   <mid>6700972667</mid>
3.   <tid>67510491</tid>
4.   <sale>
5.     <cvc></cvc>
6.     <currencyCode>YT</currencyCode>
7.     <amount>111</amount>
8.     <orderId>TCKNVKN67100876500010010</orderId>
9.     <installment>3</installment>
10.    <cardInfo>
11.      <inquiryValue>41839824368</inquiryValue>
12.      <cardNoFirst>540061</cardNoFirst>
13.      <cardNoLast>4581</cardNoLast>
14.    </cardInfo>
15.  </sale>
16. </posnetRequest>
```

Sample Request of Sales with Mail Order with TR Identity No and first 6 and last 4 digits of the Card:

```
1. <posnetRequest>
2.   <mid>4200972718</mid>
3.   <tid>68536549</tid>
4.   <sale>
5.     <cvc></cvc>
6.     <currencyCode>YT</currencyCode>
7.     <amount>2323</amount>
8.     <orderId>TCKNVKN67103480000100004</orderId>
9.     <mailorderflag>Y</mailorderflag>
10.    <cardInfo>
11.      <inquiryValue>18463889018</inquiryValue>
12.      <cardNoFirst>444676</cardNoFirst>
13.      <cardNoLast>0226</cardNoLast>
14.    </cardInfo>
15.  </sale>
16. </posnetRequest>
```

Sample Request of Sales in Installments of Sub-distributor with TR Identity No and first 6 and last 4 digits of the Card:

```

1. <posnetRequest>
2.   <mid>6700972647</mid>
3.   <tid>67510393</tid>
4.   <sale>
5.     <cvc>586</cvc>
6.     <currencyCode>YT</currencyCode>
7.     <amount>245</amount>
8.     <orderID>TAKGUNz89019876678911237</orderID>
9.     <cardInfo>
10.      <inquiryValue>41839824368</inquiryValue>
11.      <cardNoFirst>540061</cardNoFirst>
12.      <cardNoLast>4581</cardNoLast>
13.    </cardInfo>
14.    <tckn></tckn>
15.    <vkn>9994567890</vkn>
16.    <subDealerCode>fatihto2</subDealerCode>
17.  </sale>
18. </posnetRequest>

```

<b>posnetRequest - sale</b>	
It is used for the sales transaction of insurance companies.	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.
<b>sale</b>	
<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>cvc</b>	Credit card security number - CVV2. It is not a required field, this information may be sent blank as<cvc> </cvc>, if this information is not found on your system. Companies that use the CVV2 value of 000 can continue to use it in the same way.
<b>orderID</b>	Order number. 24-digits alphanumeric. Minimum 1 and maximum 24 alphanumeric characters when OrderID parameter is active.
<b>installment</b>	Number of installments. "00" should be used for Cash Transaction or this field shouldn't be used. "00" should be used for Cash Transaction.
<b>cardInfo</b>	
<b>inquiryValue</b>	TR ID or Taxpayer information of the cardholder must be sent. TR ID or Taxpayer information values sent are controlled in terms of accuracy. Taxpayer Number is 10 digits and TR ID value comprises 11 digits, it is a numeric value.
<b>cardNoFirst</b>	First 6 digits of the card, 6-digit numeric value
<b>cardNoLast</b>	Last 4 digits of the card, 4-digit numeric value
<b>subMrcld</b>	If used by a Posnet payment facilitator, the payment facilitator must send to the Posnet system the information it identifies with its customers through
<b>mrcPfld</b>	

<b>Mcc</b>	these 3 fields. Standard merchants without payment facilitators should not include these fields in xml.
<b>tckn</b>	If sub-dealer transaction will not be made, TR Identity Number / Taxpayer Identity Number / SubdealerCode fields should not be sent.
<b>vkn</b>	
<b>subDealerCode</b>	
	<p>In order for a main dealer to send a sub-dealer transaction, the sub-dealer must send at least one of the registered TR ID Number and Taxpayer Identity information.</p> <p>When the transaction is sent with TR Identity Number / Taxpayer Identity Number:</p> <ul style="list-style-type: none"> <li>• If there is only one sub-dealer registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the sub-dealer.</li> <li>• If there are no sub-dealers registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the main dealer.</li> </ul>

*Sample Response of the sales transactions carried out with TR Identity No and first 6 and last 4 digits of the card:*

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>022719163090000191</hostlogkey>
4.   <authCode>191630</authCode>
5.   <instInfo>
6.     <inst1>00</inst1>
7.     <amnt1>000000000000</amnt1>
8.   </instInfo>

```

```

9.     <pointInfo>
10.    <point>00000138</point>
11.    <pointAmount>000000000069</pointAmount>
12.    <totalPoint>00558852</totalPoint>
13.    <totalPointAmount>000000279426</totalPointAmount>
14.  </pointInfo>
15. </posnetResponse>

```

Sample Response of the sales transactions carried out with Taxpayer Number and first 6 and last 4 digits of the card:

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>022390547990000191</hostlogkey>
4.   <authCode>905479</authCode>
5.   <instInfo>
6.     <inst1>00</inst1>
7.     <amnt1>000000000000</amnt1>
8.   </instInfo>
9.   <pointInfo>
10.    <point>00000006</point>
11.    <pointAmount>000000000003</pointAmount>
12.    <totalPoint>00562714</totalPoint>
13.    <totalPointAmount>000000281357</totalPointAmount>
14.  </pointInfo>
15. </posnetResponse>

```

<b>posnetResponse – sale</b>	
Hostlogkey must be recorded for being used in transactions such as cancellation/refund..	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code. It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>Hostlogkey</b>	The unique Id on the system. It should be stored for refund/cancellation and financialization transactions.
<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is performed, it must be retained for refund and cancellation transactions.
<b>tranDate</b>	It is the actual date of operation according to the system setting and the date that will appear on the merchant administrator screens. If tranDateRequired = 1 is entered in the request, it is received. The format is as following: YYMMDDHHMMSS
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	
<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

*Sample usage scenarios for sale, return and cancellation with the TR Identity Number and 6 digits and last 4 digits of the card:*

Cancellation or refund transactions can be performed using the "hostLogKey" information returned in the response information of the sales transaction or the "orderID" information sent in the sales transaction request. Sample scenarios are given below.

Cancellation Scenario - Sending the sales request:

```
1. <posnetRequest>
2.   <mid>6700972667</mid>
3.
4.   <sale>
5.     <cvc>586</cvc>
6.     <currencyCode>YT</currencyCode>
7.     <amount>2323</amount>
8.     <orderID>TCKNVKN6714560000000001</orderID>
9.     <inquiryValue>41839824368</inquiryValue>
10.    <cardNoFirst>540061</cardNoFirst>
11.    <cardNoLast>4581</cardNoLast>
12.  </cardInfo>
13. </sale>
14. </posnetRequest>
```

Cancellation Scenario - Getting the sales response:

```
1. <posnetResponse>
2.   <approved>1</approved>
3.
4.   <authCode>235703</authCode>
5.   <hostLogKey>029725570390000201</hostLogKey>
6.   <inst1>00</inst1>
7.   <amnt1>000000000000</amnt1>
8. </instInfo>
9. <pointInfo>
10.  <point>00000138</point>
11.  <pointAmount>000000000069</pointAmount>
12.  <totalPoint>10241809</totalPoint>
13.  <totalPointAmount>000005120904</totalPointAmount>
14. </pointInfo>
15. </posnetResponse>
```

Cancellation request (with orderID information):

```
1. <posnetRequest>
2.   <mid>6700972667</mid>
3.   <tid>67510491</tid>
4.   <reverse>
5.     <transaction>sale</transaction>
6.     <orderID>TCKNVKN6714560000000001</orderID>
7.   </reverse>
8. </posnetRequest>
```

Cancellation request (with hostLogKey information):

```
1. <posnetRequest>
2.   <mid>6700972667</mid>
3.   <tid>67510491</tid>
4.   <reverse>
5.     <transaction>sale</transaction>
```

```
6. <hostLogKey> 025723570390000201 </hostLogKey>
7. </reverse>
8. </posnetRequest>
```

#### Refund Scenario - Sending the sales request:

```
1. <posnetRequest>
2. <mid>6700972667</mid>
3.
4. <sale>
5. <cvc>586</cvc>
6. <currencyCode>YT</currencyCode>
7. <amount>2323</amount>
8. <orderID> TCKNVKN6716800000000003 </orderID>
9. <cardInfo>
10. <inquiryValue>41839824368</inquiryValue>
11. <cardNoFirst>540061</cardNoFirst>
12. <cardNoLast>4581</cardNoLast>
13. </cardInfo>
14. </sale>
15. </posnetRequest>
```

#### Refund Scenario - Getting a sales response:

```
1 <posnetResponse>
2 <approved>1</approved>
3.
4. <authCode>219946</authCode>
5. <hostLogKey>025521994690000201</hostLogKey>
6. <inst1>00</inst1>
7. <amnt1>00000000000</amnt1>
8. </instInfo>
9. <pointInfo>
10. <point>00000138</point>
11. <pointAmount>00000000069</pointAmount>
12. <totalPoint>10241809</totalPoint>
13. <totalPointAmount>000005120904</totalPointAmount>
14. </pointInfo>
15. </posnetResponse>
```

#### Return request (with orderID information):

```
1. <posnetRequest>
2. <mid>6700972667</mid>
3. <tid>67510491</tid>
4. <reverse>
5. <transaction>sale</transaction>
6. <orderID> TCKNVKN6716800000000003 </orderID>
7. </reverse>
8. </posnetRequest>
```

#### Refund request (with hostLogKey information):

```
1. <posnetRequest>
2. <mid>6700972667</mid>
3. <tid>67510491</tid>
4. <return>
5. <amount>2323</amount>
```

```

6.     <currencyCode>YT</currencyCode>
7.     <hostLogKey>025521994690000201</hostLogKey>
8.     </return>
9. </posnetRequest>

```

## Provision and provision financialization with Turkish Identity Number / Taxpayer Information and card first 6 last 4 information

Financialization transactions of the standard provision and provision transaction are carried out without the card data, with the Turkish ID Number and the first 6 last 4 digits of the card. There is no difference between the financialization of the provision sent with the TR Identity Number / Taxpayer Number and the first 6 last 4 digits of the card. For relevant details please see [Financialization & Financialization in Installment at Cash Price](#).

### Request Sample

```

1 <posnetRequest>
2   <mid>6700972667</mid>
3
4   <auth>
5     <cvc></cvc>
6     <currencyCode>YT</currencyCode>
7     <amount>300</amount>
8     <orderID>TCKNVKN67100876500110012</orderID>
9     <cardInfo>
10      <inquiryValue>41839824368</inquiryValue>
11      <cardNoFirst>540061</cardNoFirst>
12      <cardNoLast>4581</cardNoLast>
13    </cardInfo>
14  </auth>
15 </posnetRequest>

```

### posnetRequest – auth

Used for the provision transaction.

### posnetRequest

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>tranDateRequired</b>	It ensures that the time that the transaction takes place in the Posnet system is included in the response. If the support is needed, this information will speed up the process.

### auth

<b>amount</b>	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234..
<b>currencyCode</b>	Currency - "TL, US, EU"
<b>cvc</b>	Credit card security number - CVV2. It is not a required field, this information may be sent blank as<cvc> </cvc>, if this information is not found on your system. Companies that use the CVV2 value of 000 can continue to use it in the same way.
<b>orderID</b>	Shopping number
<b>installment</b>	Number of installments. "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.

### cardInfo



<b>inquiryValue</b>	TR ID or Taxpayer information of the cardholder must be sent. TR ID or Taxpayer information values sent are controlled in terms of accuracy. Taxpayer Number is 10 digits and TR ID number comprises 11 digits, it is a numeric value.
<b>cardNoFirst</b>	First 6 digits of the card, 6-digit numeric value
<b>cardNoLast</b>	Last 4 digits of the card, 4-digit numeric value
<b>subMrclid</b>	If used by a Posnet payment facilitator, the payment facilitator must send to the Posnet system the information it identifies with its customers through these 3 fields. Standard merchants without payment facilitators should not include these fields in xml.
<b>mrcPfld</b>	
<b>mcc</b>	
<b>tckn</b>	If sub-dealer transaction will not be made, TR Identity Number / Taxpayer Identity Number / SubdealerCode fields should not be sent.
<b>vkn</b>	
<b>subDealerCode</b>	<p>In order for a main dealer to send a sub-dealer transaction, the sub-dealer must send at least one of the registered TR ID Number and Taxpayer Identity information.</p> <p>When the transaction is sent with TR Identity Number / Taxpayer Identity Number:</p> <ul style="list-style-type: none"> <li>• If there is only one sub-dealer registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the sub-dealer.</li> <li>• If there are no sub-dealers registered with this TR Identity Number / Taxpayer Identity Number, the transaction will take place from the main dealer.</li> <li>• There can be more than one sub-dealer registered with this Identity Number/Taxpayer Identity Number. In this case, we expect the subdealerCode to be sent. In case the subDealerCode is not specified, it will not be known which sub-dealer will be used, so the transaction will take place from the main dealer.</li> </ul>

*Response Sample*

```

1. <posnetResponse>
2.   <approved>1</approved>
3.
4.   <authCode>215366</authCode>
5.   <posnetlogkey>02002100000000000201</posnetlogkey>
6.   <inst1>00</inst1>
7.   <amnt1>000000000000</amnt1>
8. </instInfo>
9. <pointInfo>
10.  <point>00000000</point>
11.  <pointAmount>000000000000</pointAmount>
12.  <totalPoint>12609525</totalPoint>
13.  <totalPointAmount>000006304762</totalPointAmount>
14. </pointInfo>
15.</posnetResponse>

```

#### posnetResponse – auth

Hostlogkey should be registered to be used more like in cancellation / refund transactions.

#### posnetResponse



**approved**

Transaction result.  
0: Unsuccessful.

	1: Successful.
<b>respCode</b>	Error code. It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	The unique Id on the system. It should be stored for refund/cancellation and financialization transactions.
<b>authCode</b>	It is the authorization code generated based on the system authorization. If a transaction with delay interest is performed, it must be retained for refund and cancellation transactions.
<b>tranDate</b>	It is the actual date of operation according to the system setting and the date that will appear on the merchant administrator screens. If tranDateRequired = 1 is entered in the request, it is received. The format is as following: YYMMDDHHMMSS
<b>instInfo</b>	
<b>inst1</b>	Indicates the number of installments used in the transaction. For example: 00 or 03
<b>amnt1</b>	Installment amount - in Kurus Ex: 12.34 TL should be set as 1234.
<b>pointInfo</b>	
<b>point</b>	World points earned from the transaction
<b>pointAmount</b>	World amount equivalent to TL earned from the transaction
<b>totalPoint</b>	Total world points of the card
<b>totalPointAmount</b>	Total world points equivalent of TL

Points transaction with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card

Point transactions can also be performed with the TR ID / Taxpayer Number and the first 6 and last 4 digits of the card. Below, points request and use of points are explained with sample requests. There is no change in the cancellation and return request and response structures of the points usage transactions performed with the TR ID / Taxpayer Number and the first 6 and the last 4 digits information of the card.

Points inquiry with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card

This is used to display the worldpoints of a WorldCard. It does not appear on the card statement and card page.

*Request Sample*

```

1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <pointInquiry>
6.     <cardInfo>
7.       <inquiryValue>41839824368</inquiryValue>
8.       <cardNoFirst>540061</cardNoFirst>
9.       <cardNoLast>4581</cardNoLast>
10.    </cardInfo>
11.  </pointInquiry>
12. </posnetRequest>

```

#### **posnetRequest – pointInquiry**

It is used to inquiry the World points.

#### **posnetRequest**

<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
------------	-------------------------------------

<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>pointInquiry</b>	
<b>cardInfo</b>	
<b>inquiryValue</b>	TR ID or Taxpayer information of the cardholder must be sent. TR ID or Taxpayer information values sent are controlled in terms of accuracy. Taxpayer Number is 10 digits and TR ID value comprises 11 digits, it is a numeric value.
<b>cardNoFirst</b>	First 6 digits of the card, 6-digit numeric value
<b>cardNoLast</b>	Last 4 digits of the card, 4-digit numeric value

#### Response Sample

```

1. <?xml version='1.0' encoding='iso-8859-9'?>
2. <posnetResponse>
3.   <approved>1</approved>
4.   <pointInfo>
5.     <point>59386187</point>
6.     <pointAmount>000029693093</pointAmount>
7.     <totalPoint>00000000</totalPoint>
8.     <totalPointAmount>000000000000</totalPointAmount>
9.   </pointInfo>
10. </posnetResponse>

```

<b>posnetResponse – pointInfo</b>	
Points information is accessed at the end of points inquiry transaction	
<b>posnetResponse</b>	
<b>approved</b>	Transaction result. 0: Unsuccessful 1: Successful
<b>respCode</b>	Error code. It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>pointInfo</b>	
<b>point</b>	Total world points available on the card
<b>pointAmount</b>	The total amount of world points available on the card, equivalent in TL Ex: 1234 for 12.34 TL
<b>totalPoint</b>	N/A
<b>totalPointAmount</b>	N/A

#### Points usage with TR ID No/ Taxpayer ID No and first 6 and last 4 digits of the card

This is performed to use the Worldpoints of a WorldCard. The transaction appears in the "Your WorldPoint Information" section of the card statement. The use of points can also be realized by making mixed transactions. By using Mixed Transaction instead of using points, 0 (zero) can be sent to amount field and desired points can be sent to point field. Please see, [Mixed Transactions](#)

## Request Sample

```
1. <?xml version="1.0" encoding="ISO-8859-9"?>
2. <posnetRequest>
3.   <mid>6706598320</mid>
4.   <tid>67005551</tid>
5.   <pointUsage>
6.     <cvc></cvc>
7.     <currencyCode>YT</currencyCode>
8.     <amount>100</amount>
9.     <orderID>YKB_TST_202102064500_020</orderID>
10.    <cardInfo>
11.      <inquiryValue>41839824368</inquiryValue>
12.      <cardNoFirst>540061</cardNoFirst>
13.      <cardNoLast>4581</cardNoLast>
14.    </cardInfo>
15.  </pointUsage>
16.</posnetRequest>
```

### posnetRequest – pointUsage

It is used to use the World points.

#### posnetRequest

mid	YKB Merchant Number <%MERCHANT_ID%>
tid	YKB Merchant Terminal Number <%TERMINAL_ID%>

#### pointUsage

amount	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
currencyCode	Currency - "TL, US, EU"
orderID	Shopping order number

#### cardInfo

inquiryValue	TR ID or Taxpayer information of the cardholder must be sent. TR ID or Taxpayer information values sent are controlled in terms of accuracy. Taxpayer Number is 10 digits and TR ID value comprises 11 digits, it is a numeric value.
cardNoFirst	First 6 digits of the card, 6 digits numeric value
cardNoLast	Last 4 digits of the card, 4 digit numeric value

## Response Sample

```
1. <?xml version='1.0' encoding='iso-8859-9'?>
2. <posnetResponse>
3.   <approved>1</approved>
4.
5.   <pointInfo>
6.
7.     <pointAmount>000000000000</pointAmount>
8.
9.     <totalPointAmount>000029692893</totalPointAmount>
10.
11. </posnetResponse>
```

### posnetResponse – pointInfo

At the end of the points usage transaction, the point information is accessed.

#### posnetResponse

approved	Transaction result.
----------	---------------------

	0: Unsuccessful 1: Successful
<b>respCode</b>	Error code. It must be considered when the transaction is unsuccessful. Error Codes section provides explanations.
<b>respText</b>	Error message.
<b>hostlogkey</b>	It is the unique ID of the transaction on the system side. Cancellation must be retained for refunds.
<b>pointInfo</b>	
<b>point</b>	Total world points available on the card
<b>pointAmount</b>	The total amount of world points available on the card, equivalent in TL Ex: 1234 for 12.34 TL
<b>totalPoint</b>	Remaining Worlds points of the card
<b>totalPointAmount</b>	The remaining amount of world points available on the card, equivalent in TL Ex: 1234 for 12.34 TL

## Error codes

The error codes that may be received in case of incorrect parameter entry or connection to Posnet are listed below.

<b>Error Code</b>	<b>What needs to be done</b>
<b>100 – OK</b>	Communication with the Posnet server was successfully established. However, this result code does not mean that the transaction is successful. The response from the server needs to be checked to see if the transaction is successful.
<b>101 - CONNECT_ERROR</b>	The connected server ip must be checked.
<b>103 - PACKET_ERROR</b>	This error is returned when Posnet server cannot resolve the packet it receives. Since source ip (ownIP) is used in the analysis process, it should be ensured that this parameter is the same as your IP. The information on the IP Based Errors page can also help you solve the problem.
<b>113 - CONNECT_CONNECT</b>	<p>It should be checked that the hostname parameter is set correctly and has internet connection. By establishing a telnet collection to Hostname parameter (address) in order to control the access to Firewall, etc., the existence of an access problem is controlled. (For example, from the command line: telnet 193.254.228.53 2222). When establishing a telnet connection, make sure that the value entered in the port parameter (2222 unless specified otherwise in the documentation) is also entered in the telnet command (you can only connect to the Posnet server from the correct port, this is also valid for telnet).</p> <p>If you cannot establish a telnet connection to the Posnet server, there is a problem with your internet connection. For example, in your firewall settings, you should ensure that the posnet server uses the correct port. Most firewalls allow only port to connect to http (80) and https (8080). In this case, 2222 (or the connection port specified in the documentation) must be added between the allowed ports.</p> <p>If there is no problem in your internet connection, you should contact</p>

	the test support group.
<b>115 - CONN_REFUSED</b>	Posnet server refused your connection request. You may have tried a transaction from an IP that is not in the list of IPs, where your company can send transactions to the Posnet system. The information on the IP Based Errors page can also help you solve the problem.
<b>120 - CGI_SERVLET_ERROR</b>	The connection was opened, but the packet could not be sent.
<b>121 - EXCHANGE_TIMEOUT</b>	No response from posnet server. There may be a problem with your Internet connection. If there is no problem with your Internet connection, try again, and if the problem persists, call test support team.
<b>131 - ERROR_CCNO</b>	Card No parameter is incorrect. See parameter descriptions.
<b>132 - ERROR_HOSTLOGKEY</b>	Hostlogkey parameter is incorrect. See parameter descriptions.
<b>133 - ERROR_AUTH</b>	The authorization code parameter is incorrect. See parameter descriptions.
<b>134 - ERROR_HOSTNAME</b>	Hostname parameter is incorrect. See parameter descriptions.
<b>135 - ERROR_PORT</b>	Port parameter is incorrect. See parameter descriptions.
<b>136 - ERROR_OWNIIP</b>	Ownip parameter is incorrect. See parameter descriptions.
<b>137 - ERROR_AMOUNT</b>	Amount parameter is incorrect. Before sending the amount, you must make sure that the last two digits are in kurus and that brackets such as cents or thousands are not used. For example, you must enter 512 to send 5.12 TL, or 500 to send 5 TL.
<b>138 - ERROR_EXPDATE</b>	The credit card expiration date parameter is incorrect. See parameter descriptions.
<b>139 - ERROR_CVC</b>	The credit card security number (CVC) parameter is incorrect. See parameter descriptions.
<b>140 - ERROR_TAKNUM</b>	The installment parameter is incorrect. The installment parameter must be 2 characters long and should be numeric. Ex: 02. If no installment will be used, 00 or 01 must be entered.  Entering the installment parameter 00 or 01 also causes this error in operations that must be in installments (for example; VFT).
<b>142 - ERROR_MIDNO</b>	The merchant number (MID) parameter is incorrect. See parameter descriptions.
<b>143 - ERROR_TIDNO</b>	The terminal number (TID) parameter is incorrect. See parameter descriptions.
<b>144 - ERROR_ORDERID</b>	The order number (ORDERID) parameter is incorrect. It must be 20 characters long and consists of only letters and numbers. Please see the parameter descriptions.
<b>146 - ENCRYPTION ERROR</b>	Encryption error. Send an e-mail to <a href="mailto:possupp@yapikredi.com.tr">possupp@yapikredi.com.tr</a>
<b>147 - CURRENCY CODE ERROR</b>	The currency parameter is incorrect. It is taken when a value other than "TL" or "YT" is entered in the CurrencyCode parameter. The most common cause of this error is to enter "YTL" as a parameter.
<b>156 - ERROR_VFT_CODE</b>	VFT Campaign Code is incorrect. It needs to be 4 characters long.
<b>180 - MULTI AND EXTRA POINTS</b>	It is not possible to specify both the multi points and extra points in the same transaction. You must enter either the multi point parameter 00 or the extra point parameter 000000.
<b>181 - ERROR_TXNSEQNO</b>	The TranSeqNo parameter is incorrect. See parameter descriptions.
<b>184 - ERROR_TRANSTYPE</b>	The transaction type parameter is incorrect. See parameter descriptions.
<b>185 - ERROR_BONUS</b>	Point transaction type is incorrect.
<b>186 - ERROR_EXTRAPPOINT</b>	Extra points parameter is incorrect. See parameter descriptions.
<b>187 - ERROR_MULTIPLE</b>	Multiple point parameter is incorrect. See parameter descriptions.

If there is no problem in the communication with posnet system and in the parameters (transaction error = 100), the errors that can be received and the actions that should be taken are given below.

<i>Error Code</i>	<i>Explanation</i>	<i>What needs to be done</i>
<b>0001</b>	BANKANIZI ARAYIN 0001 (CALL YOUR BANK)	The card does not allow this type of transaction or the credit of the card is insufficient. Call the bank that issued the card.
<b>0004</b>	RED-KARTA EL KOY 0004 (REJECT - CONFISCATE THE CARD)	The card is blocked.
<b>0005</b>	RED-ONAYLANMADI (REJECT-DIDN'T APPROVE)	<p>One or more of the card information (Credit card no, expiry date, CVV) may be entered incorrectly or the bank-defined daily limits for World cards may be exceeded.</p> <p>To make sure that the card information is entered correctly, a trial can be performed from the "Online Transactions" page on the Merchant Administrator Screens. Receiving this error also means that the card information is sent correctly.</p> <p>Another reason for this error is the <i>limit of daily transactions, defined by the cardholder bank, to be completed on internet</i> has been reached. This limit varies according to each bank and it is 3 for YKB credit cards; it means a YKB credit card can be used for up to 3 shopping on internet per day. If this limit is exceeded, the cardholder must call this bank's credit card customer service and reset it.</p> <p>The amount entered cannot be greater than the provision amount in the financialization process and the financialization amount in the refund process.</p>
<b>0007</b>	BANKANIZI ARAYIN 0007 (CALL YOUR BANK)	The card may be blocked/stolen/lost (special case).
<b>0012</b>	RED-GECERSIZ ISLEM (REJECT-INVALID TRANSACTION)	<p>The most common cause of this error is that you try to install with the wrong number of installments. To find out how many installments you can use, you should call 444 0 448. If you are making this transaction with test cards, you may find out the information on <a href="mailto:posnet.support@yapikredi.com.tr">posnet.support@yapikredi.com.tr</a> address. Generally, up to 9 installments can be used for normal transactions.</p> <p>Another reason you get error 0012 is that you do something that the card does not allow. For example, you will receive this error if you try to sell with installment on a credit card belonging to another bank.</p> <p>If these steps didn't help you resolve the issue, there</p>

		may be problems with your bank merchant definitions. By calling our merchant service, you need to give your merchant number and the detail of the transaction that causes this error.
0014	RED-HATALI KART 0014 (REJECT - INCORRECT CARD)	The number does not belong to a credit card/Card number is incorrect.
0015	PROVIZYON BULUNAMADI (NO PROVISION FOUND)	No provision has been placed. Provision may have been canceled. You must place the provision again.
0015	TERMINAL İŞLEM YETKİSİ YOK (NO TERMINAL TRANSACTION AUTHORITY)	Terminal authorization is not suitable for the transaction.
0015	İŞYERİ STATÜSÜ HATALI (MERCHANT STATUS INCORRECT)	Merchant status is not appropriate.
0015	TAKSİT İÇİN YETERSİZ TUTAR (INSUFFICIENT AMOUNT FOR INSTALLMENT)	This error is given if the amount entered for the installment is below the minimum amount.
0030	BANKANIZI ARAYIN 0030 (CALL YOUR BANK)	The reason for this error is the corrupt data sent by the issuer bank to the YKB provision system. The bank that issued the card should be called and indicated that this error was received in a virtual pos transaction. In order to find a solution to the problem until the error is resolved, the transaction can be sent to YKB via mail order. To realize a mail order transaction, call our <a href="#">merchant service</a> .
0041	RED-KARTA EL KOY 0041 (REJECT - CONFISCATE THE CARD)	Lost Card - Call (444 0 448).
0043	RED-KARTA EL KOY 0043 (REJECT - CONFISCATE THE CARD)	The cause of the problem is that the credit card used in the transaction is in the <b>stolen credit card list</b> , held in YKB provision system. The transaction is rejected before forwarding to the cardholder bank.  Credit cards used in virtual POS transactions may be put into a blacklist list by YKB for various reasons. If you believe that the card is incorrectly in the stolen list (the card is a trusted card), you should call the YKB Merchant Operations Service (444 0 448).
0051	RED-YETERSİZ BAKIYE 0051 (REJECT - INSUFFICIENT BALANCE)	The card has insufficient balance. Call the bank that issued the card.
0053	BANKANIZI ARAYIN 0053 (CALL YOUR BANK)	This account isn't found.
0054	RED-ONAYLANMADI 0054 (REJECT- WASN'T APPROVED)	The credit card is expired.
0057	RED-ONAYLANMADI 0057 (REJECT- WASN'T APPROVED)	The transaction cannot be realized with the type of card used (Debit/credit). Example: POSNET cannot process debit cards (debit cards are used to withdraw money from ATMs). In the error message, where "X" is specified, the type of the card is specified (D:Debit /K: Credit card).
0057	RED-ONAYLANMADI 0057 (REJECT- WASN'T APPROVED)	This error is received when there is a problem with the authorization of the credit card used in the transaction



		to make transactions from the internet. The cardholder should contact the credit card service of the bank where s/he receives the credit card and indicate that s/he cannot use the credit card in e-commerce.
<b>0058</b>	RED-ONAYLANMADI 0058 (REJECT- WASN'T APPROVED)	The terminal is not authorized for the transaction type.
<b>0062</b>	RED-ONAYLANMADI 0062 (REJECT- WASN'T APPROVED)	Restricted card.
<b>0065</b>	RED-ONAYLANMADI 0065 (REJECT- WASN'T APPROVED)	This error, given when the credit card withdrawal limit is exceeded, should not be returned in virtual pos transactions under normal circumstances. If this error is received, the issuer bank should be called and it should be stated that the error has been received in a virtual pos transaction. In order to find a solution to the problem until the error is resolved, the transaction can be sent to YKB via mail order. To realize a mail order transaction, call our <a href="#">merchant service</a> .
<b>0091</b>	BANKANIZI ARAYIN 0091 (CALL YOUR BANK)	There was a timeout in communication with the issuer bank (no timely response from the bank). Try again; if the problem persists, call the issuing bank and indicate that this error was received in a virtual pos transaction.
<b>0100</b>	HOST RECEIVE PROBLEM	This error can sometimes be received when there are instant problems in our bank systems. Try again, if the problem persists, contact <a href="mailto:posnet.destek@ykb.com">posnet.destek@ykb.com</a> .  If this error occurs in the test environment, deleting the definitions of the test card used may cause the problem. To eliminate this possibility, you may need to try with several different test cards.
<b>0122</b>	DATABASE DE ISTENILEN KAYIT YOK (REQUIRED RECORD DOESN'T EXIST ON DATABASE)	Error in cancellation. Cancellation can be done up to 1 week after the provision. This error may also be received if the financialization is canceled without completing the financialization.  One reason for this error is that you want to financialize or cancel a transaction you have already done with your merchant mid using another mid of your company. The most common way to do this is to programmatically financialize or cancel a transaction made using one mid programmatically using another mid.
<b>0123</b>	ORJINAL ISLEM BULUNAMADI (ORIGINAL TRANSACTION CANNOT BE FOUND)	The transaction to be financialized, refunded or cancelled cannot be found. You are probably trying to financialize/cancel with the wrong YKB ref.no or order no. The transaction you are trying to financialize/cancel may not have been sent to the Posnet system at all.  In the cancellation of <a href="#">VFT</a> transactions, if the cancellation is made with YKB ref. no, the authentication code is also checked. In case of cancellation, the confirmation code should be checked

		<p>together with YKB ref.no.</p> <p>When no response is received from the Posnet system for an transaction, it is normal to receive this error upon automatic cancellation; this means that the transaction never reaches the Posnet system.</p>
<b>0124</b>	HOST SESSION OPEN PROBLEM	This error is due to the environment of our bank. Occasionally, there are instant interruptions due to the work in our Bank's environments. If this error is received, the transaction should be retried after a while, if the problem persists, <a href="mailto:posnet.destek@ykb.com">posnet.destek@ykb.com</a> should be contacted.
<b>0125</b>	ORDERID VAR HOSTLOGKEY YOK DB ERR (ORDERID EXISTS, HOSTLOGKEY DOESN'T EXIST, DB ERR)	Call YKB.
<b>0126</b>	ORDERID VAR KK SIFRELEME HATASI (ORDERID EXISTS, CREDIT CARD ENCRYPTION ERROR)	Call YKB.
<b>0127</b>	ORDERID DAHA ONCE KULLANILMIS (ORDERID HAS BEEN USED BEFORE)	The order no (orderId) you are using has been previously used. Try again with a new order no.
<b>0129</b>	KREDİ KARTI MERCHANT BLACKLIST TE (CREDIT CARD IS IN THE MERCHANT BLACKLIST)	This credit card is included into the merchant <a href="#">blacklist</a> . The card must be removed from the blacklist before the merchant can use it.
<b>0146*</b>	HATALI SIFRELEME : KULLANICI ISMI & SIFRE veya NO GENERATED RECORD (ERROR IN ENCRYPTION: USER NAME & PASSWORD OR NO GENERATED RECORD)	The user name, password or encryption key is entered incorrectly. Please check StubF1Class.setUserName, StubF1Class.setPassword, StubF1Class.setEncKey methods for more information. It is necessary to use " <a href="#">Create Key</a> " on the main menu of Merchant Administrator Screens and new user name, password and key must be generated and retry this transaction with new information.
<b>0147*</b>	HATALI KULLANICI ISMI & SIFRE (ERROR IN USER NAME & PASSWORD)	See explanations of error 146.
<b>0148*</b>	CRYPTO HATASI : MID (ERROR IN CRYPTO: MID)	<p>Your web server's date, time, or TimeZone information may be incorrect. If there is no problem with this information, please contact our Technical Support.</p> <p>Posnet Service, which responds to the information you send, uses date and time to open some encrypted information. If your server's date or time is incorrect, this information cannot be resolved by the service.</p>
<b>0148*</b>	HATALI MID (ERROR IN MID)	The merchant number cannot be found. The merchant no (MID) parameter is incorrect.
<b>0148*</b>	MID,TID,IP HATALI: X.X.XX (ERROR IN MID, TID, IP)	You are trying to make a connection from a wrong or unauthorized IP when making a connection. Sending a process to the wrong environment (for example, live environment mid and test environment) also causes this problem.

		<p>For the test environment, you need to send the transaction to <a href="https://setmpos.ykb.com/posnetwebservice/xml">https://setmpos.ykb.com/posnetwebservice/xml</a> while you need to send the transaction to <a href="https://www.posnet.ykb.com/posnetwebservice/xml">https://www.posnet.ykb.com/posnetwebservice/xml</a> for live environment.</p> <p>If you are sure that you are sending your transaction to the correct environment, you can change your IP definition by sending your request ip to <a href="mailto:posnet.destek@ykb.com">posnet.destek@ykb.com</a> with mid/tid as indicated in the error message XXXX.</p>
0150	PAKET HATALI (ERROR IN PACKET)	In cases where the OrderID parameter of the merchant is activated, not sending the orderDate information along with the orderID in transactions that require the original transaction to be present, such as cancellation / return / financialization, causes this error.
0150	INVALID MID TID IP	You are trying to make a transaction from a wrong IP or a wrong mid/tid. The information on the <a href="#">IP Based Errors</a> page can also help you solve the problem.
0200	GECERSIZ ISLEM (INVALID TRANSACTION)	Received when you submit an invalid transaction. For example, attempting to refinance a transaction that has already been financialized, or to refund a transaction that is in provision status. This type of invalid transaction is not already allowed on the Merchant Administrator Screens, but this control is performed for the transactions sent in the program (using technology such as ASP).
0205	GECERSIZ TUTAR (INVALID AMOUNT)	<p>This error is received under the following conditions:</p> <ul style="list-style-type: none"> <li>When the amount of the transaction exceeds the maximum transaction amount (99.999.99 TL). Up to 99.999.99 TL, a transaction can be made in Posnet system at one time.</li> <li><a href="#">While financializing</a> the amount of the transaction exceeds the <a href="#">provision overrun percentage</a>.</li> <li>In <a href="#">return</a> transactions, when the transaction amount exceeds the refundable amount.</li> </ul>
0211	GROUP CLOSING COMPLETED	This error is received when making financialization or sales cancellation. The transaction you want to cancel is financialized and can no longer be canceled. To return your financialization or refund the sales, you need to make a <a href="#">return</a> transaction.
0217	GEÇERSİZ İŞLEM STATÜSÜ (INVALID TRANSACTION STATUS)	Stolen card. It is necessary to notify YKB about the user name and card number.
0218	BU SIPARIS DAHA ONCE IADE EDİLDİĞİ İÇİN İPTAL İŞLEMİ GEÇERSİZDİR	Received when attempting to cancel a refunded transaction.

	SINCE THIS ORDER HAS BEEN REFUNDED, CANCELLATION IS INVALID	
<b>0219</b>	CANNOT FINANCIALIZE AGAIN SINCE FINANCIALIZATION HAS BEEN DONE TODAY WITH THIS ORDER ID	For provisioning transactions with the same order id performed on different days, financialization can only be sent to one of them on the same day, this error is received when it is sent to a second transaction. Financialization should be tried on a different day.
<b>0220</b>	IPTALISLEMI YAPILMIS,	This error is received when you try to cancel again a

	(CANCELLATION COMPLETED)	transaction already cancelled.
<b>0223</b>	ONAYLANMADI (WASN'T APPROVED)	Without financialization, financialization is requested to be canceled.
<b>0232</b>	KREDİKARTI İŞLEM SINIRI AŞILDI (CREDIT CARD LIMIT EXCEEDED)	When the maximum number that can be processed with a credit card is exceeded in a certain period defined by the Merchant in the Posnet system, the related error is received. See. <a href="#">Transaction Restriction</a>
<b>0370</b>	İSLEM İPTALI YAPILMIS (TRANSACTION CANCELLED)	The cancellation has already been done.
<b>0400</b>	DB ERROR	Posnet server is having a technical problem. Try again, if the problem repeats, contact the <a href="#">Technical support</a> team.
<b>0411</b>	İSLEM HENUZ FINANSALLASMAMIS (TRANSACTION NOT YET FINANCIALIZED)	This error received when making a refund indicates that the amount specified in the financialization transaction has not yet been collected from the card and reflected in your account. Therefore, you do not need to make a refund transaction; you must cancel the financialization.
<b>0444</b>	BANKANIZI ARAYIN (CALL YOUR BANK)	Call YKB.
<b>0450</b>	İADE İSLEMİ YAPILAMIYOR (RETURN TRANSACTION CANNOT BE COMPLETED)	It may be <a href="#">refunded</a> from a screen other than the merchant administrator screen. You may have been refunded the transaction by calling our <a href="#">merchant service</a> . If you did not request such a refund, you should call our merchant service.
<b>0788</b>	FINANSAL İSLEM YAPILMIS (FINANCIAL TRANSACTION COMPLETED)	Financialization is completed. If you want to cancel a provision, financialization must be canceled first.
<b>0354</b>	POINTAMOUNT AND CARDINFO INFORMATION CANNOT BE SENT TOGETHER	This error is received when the fields used for the use of points are sent in transactions with the first 6 last 4 digits of the card and the TR Identity Number / Taxpayer Number, and points cannot be used in these transactions.

## Steps for Going Live

After completing your tests in the test environment, you must send your request to go live to [possupp@yapikredi.com.tr](mailto:possupp@yapikredi.com.tr). In the mail attachment you will send, you need to include distinctive information (MERCHANT\_ID, TERMINAL\_ID, POSNET\_ID, SOURCE\_IP, ORDER\_NO, TRANSACTION\_DATE, etc.) and the date of the transaction.

For each service request integration, following information shall be added to Request Header: X-MERCHANT-ID, X-TERMINAL-ID, X-POSNET-ID, X-CORRELATION-ID.

1. The MERCHANT\_ID, TERMINAL\_ID, POSNET\_ID information can also be found on the Merchant information page on the Merchant Admin Screens.
2. If the environment variables and XML\_SERVICE\_URL are used, OOS\_TDS\_SERVICE\_URL is added to the merchant live environment application.

- Live environment IP information is defined to the system through merchant management screens.

Merchant application configurations are updated so that variables defined as environment variables are used in a live environment.

<i>Key</i>	<i>Type</i>	<i>Description</i>	<i>Sample Data</i>
MERCHANT_ID	String	10 digit YKB (Yapı Kredi Bank) merchant number	6706598320
TERMINAL_ID	String	8 digits YKB merchant terminal number	67005551
POSNET_ID	String	Up to 16 digits, YKB merchant POSNET number. It is used in 3D Secure encryption transactions.	9644
XML_SERVICE_URL	String	Bank integration service address	<a href="https://www.posnet.ykb.com/PosnetWebService/XML">https://www.posnet.ykb.com/PosnetWebService/XML</a>
OOS_TDS_SERVICE_URL	String	Bank common payment and 3D Secure page address	<a href="https://www.posnet.ykb.com/3DSWebService/YKBPaymentService">https://www.posnet.ykb.com/3DSWebService/YKBPaymentService</a>

If the merchant is making 3D secure payments or using the common payment page provided by Posnet, it means that 3D Secure is active and the customer of the merchant, so the end user, will be directed from the merchant screens to the bank screens and sent back to the merchant screen after passing through the security and verification steps on the bank screens. MAC validation is performed on 3DS payment flows in order to prevent the customer's movement between networks. To create MAC data, it is necessary to follow the Key Generation step from the merchant management screens and set a **ENCYKEY** value for the live environment. It should be noted that this value does not contain Turkish characters and spaces.

## History

Date	Version	Explanation	Prepared by
19.05.2019	2.0.1.0 beta	A platform independent integration document was created by using the reference of the documentation prepared on the development environment (.net, java, php, etc.). <ul style="list-style-type: none"> <li>• Credit Card Transactions</li> <li>• Transactions of Points</li> <li>• Mixed Transaction</li> <li>• Transactions with Delay Interest (VFT)</li> <li>• Joker Vadaa Transactions</li> <li>• Error codes</li> </ul>	Nazım Sezer - Virtual Pos and Campaign Application Development
12.06.2019	2.0.1.1 beta	Reconciliation services have been added. <ul style="list-style-type: none"> <li>• Transaction Status Inquiry</li> <li>• Daily Transaction Report Inquiry</li> </ul>	Nazım Sezer - Virtual Pos and Campaign Application Development
03.07.2019	2.0.1.2 beta	Trio services have been added.	Nazım Sezer - Virtual Pos and Campaign Application Development
12.07.2019	2.0.1.3 beta	New generation payment services added	Nazım Sezer - Virtual Pos and Campaign Application Development
11.10.2019	2.0.1.5	Agreement metodunun response unayeni eklene alanlarla ilgili olarak güncelleme yapılmıştır.	Tolga Akgün – Virtual POS Application Development
10.01.2020	2.0.1.6	TCKN/VKN vekaletinin ilk 6 veson 4 hanesi ile satış işlemine ait bilgiler eklenmiştir.	Tolga Akgün - Virtual POS Application Development
08.04.2020	2.0.1.7	<ul style="list-style-type: none"> <li>• Hata kodları bölümü güncellendi.</li> <li>• TCKN/VKN vekaletinin ilk 6 veson 4 hanesi ile satış işlemine ait bilgiler güncellendi, aşağıda belirtilenler eklendi; <ul style="list-style-type: none"> <li>○ Provizyon ve provizyon finansallandırma işlemleri örnek request ve response bilgileri</li> <li>○ Taksitli satış işlemi örnek request ve response bilgileri</li> </ul> </li> </ul>	Tolga Akgün - Virtual POS Application Development

		<ul style="list-style-type: none"> <li>Mailorder işlem örnek request ve response bilgileri</li> <li>İade ve İptal örnek senaryolar</li> </ul>	
14.12.2020	2.0.1.8	<ul style="list-style-type: none"> <li>Altbayi entegrasyonuna ait tckn, vkn, subdealercode alanı hakkındaki bilgi düzenlendi</li> <li>Posnet.ykb.com urleri Posnet.yapikredi.com.tr olarak güncellendi.</li> </ul>	Enes Köksalınış - Virtual POS Application Development
05.01.2021	2.0.1.9	<ul style="list-style-type: none"> <li>Karma İşlem tutar bilgileri düzeltildi</li> <li>Puan İade akış eklendi</li> <li>Karma işlem İadesi ve İptali akış eklendi</li> <li>İadesi olan bir işlemin iptal edilemeyeceği konusundaki iş kuralları eklendi.</li> <li>Eşleniksiz iade işlem bilgileri eklendi.</li> <li>Satış ve Taksitli Satış işlemlerinin Mail Order olarak gerçekleştirilmesi akış eklendi.</li> </ul>	Fatih Topçu - Virtual POS Application Development
06.02.2021	2.0.2.0	<ul style="list-style-type: none"> <li>TCKN/VKN vekartınıilk6 veson4 hanesi ile puansorguve kullanım işlemlerine ait bilgilere eklenmiştir.</li> </ul>	Tolga Akgün - Virtual POS Application Development
25.03.2021	2.0.2.1	<ul style="list-style-type: none"> <li>Kredi Kartı ve işyeri bilgileri güncellendi</li> </ul>	Tolga Akgün - Sanal Pos Uygulama Geliştirme
15.04.2021	2.1	<ul style="list-style-type: none"> <li>Puan Sorgulama işleminde Marka Puan sorgulama yönteminin eklenmesi</li> <li>Puan Kullanma işleminde Marka Puan kullanım yönteminin eklenmesi</li> <li>Puan İade işleminin İptali fonksiyonunun eklenmesi.</li> <li>Karma işlemlerde Marka Puan kullanım yönteminin tariflenmesi.</li> <li>Karma İşlemin Oransal İadesi fonksiyonunun eklenmesi.</li> </ul>	Fatih Topçu - Sanal Pos Uygulama Geliştirme
27.08.2021	2.1.1.0	<ul style="list-style-type: none"> <li>TDSC ifadesi TDS_ olarak değiştirildi.</li> <li>PointReturn isteğindeki &lt;amount&gt; tagi &lt;wpAmount&gt; olarak güncellendi.</li> </ul>	Burcu Kaya - Sanal Pos Uygulama Geliştirme
15.04.2022	2.1.1.1	<ul style="list-style-type: none"> <li>İşlem Durumu Sorgulama (Agreement) Servisinden dönen değerler güncellendi.</li> </ul>	Bengi Başar - E-Ticaret ve Ödeme Çözümleri Uygulama Geliştirme



10.01.2023	2.1.1.2	<ul style="list-style-type: none"><li>Order id parametresi kapsamındaki geliřtirmeler eklendi</li></ul>	Tuba inal – E-Ticaret ve Ödeme özümleri Uygulama Geliřtirme
24.05.2023	2.1.1.3	<ul style="list-style-type: none"><li>Order id parametresinin aktif olması durumu ile ilgili povizyon iřlemi, iade iřlemi, iadenin iptali iřlemi ve mutabakat iřlemine kurallar eklenmiřtir.</li></ul>	Tuba inal – E-Ticaret ve Ödeme özümleri Uygulama Geliřtirme