

# POSNET XML Service

## Content

Content.....	2
Introduction.....	3
General Structure of the Service .....	3
Credit Card Transactions .....	4
Sales (Direct Sales) & Sales in Installments at Cash Price .....	4
Provision & Provision in Installment at Cash Price.....	7
Financialization & Financialization in Installment at Cash Price .....	10
Cancellation transaction (sale, provision, financialization, etc.).....	12
Return transaction.....	14
Transactions of Points .....	16
Points Inquiry.....	16
Points Usage .....	17
Cancellation of Points Usage .....	18
Mixed Transactions .....	19
Transactions with Delay Interest (VFT) .....	22
Delay Interest Inquiry .....	22
Sales in Installment with Delay Interest.....	24
Return of Sales with Delay Interest.....	26
Personalized - Joker Vadaa Transactions .....	27
Personalized Transaction Inquiry .....	28
Trio Transactions .....	30
Single Payment.....	30
Fixed payment .....	31
Multi-Payment.....	33
Return.....	34
Limit Inquiry.....	35
Reconciliation Services .....	36
Transaction Status Inquiry (Agreement) .....	36
Daily Transaction Report Inquiry.....	38
New Generation Payment .....	41
Error codes .....	44
Steps for Going Live .....	52
History .....	53

## 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 'going to live request' mail to [posnet.support@yapikredi.com.tr](mailto:posnet.support@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 merchants whom integrated to the system the best experience to the customers, Joker Vadaa Inquiry (that provides personal discounts) should be checked after the card number is entered on the screens.

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 "POSNET 3D Secure Integration" document.

## General Structure of the Service

Posnet XML service that enables Posnet merchants to make posnet transactions by sending XML documents. The merchants should POST the xml document after encoding the xml with UTF-8 URL Encode to <%XML\_SERVICE\_URL%> address (testing environment: <https://setmpos.ykb.com/PosnetWebService/XML>) at "xmldata" parameter with Content-Type=application/x-www-form-urlencoded; charset=utf-8. The result is returned to the merchant as an XML document. <%XML\_SERVICE\_URL%> shall be an environment variable

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, 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)
- 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>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.      <!--
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>
24. </posnetRequest>

```

#### posnetRequest – sale

It is used for direct sales 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.

#### sale

<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
<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 these 3 fields. Standard merchants without payment facilitators should not include these fields in xml.
<b>mrcPfid</b>	
<b>mcc</b>	
<b>tckn</b>	When the merchants that make transactions with the Main-Dealer and sub-dealer relationship fill out any or all of these 3 fields, the transaction is performed by identifying the sub-dealer with the tckn/vkn/subdealercode of the main dealer located in the mid and tid fields. Standard merchants without Main dealer-Sub dealer relationship should not include these fields in xml.
<b>vkn</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>000000000000</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.

#### posnetResponse

<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. When requesting the service with a previously used orderId, the response will be received as follows. If the result of the first request could not be obtained due to network reasons, the second request will be completed after viewing approved:2 and respCode:127.

```
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>
```

### 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.
<b>auth</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
<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 these 3 fields. Standard merchants without payment facilitators should not include these fields in xml.
<b>mrcPfld</b>	
<b>mcc</b>	
<b>tckn</b>	When the merchants that make transactions with the Main-Dealer and sub-dealer relationship fill out any or all of these 3 fields, the transaction is performed by identifying the sub-dealer with the tckn/vkn/subdealercode of the main dealer located in the mid and tid fields. Standard merchants without Main dealer-Sub dealer relationship should not include these fields in xml.
<b>vkn</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>000000000000</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.

#### 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

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

### Financialization & Financialization in Installment at Cash Price

Financialisation 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 [posnet.destek@ykb.com](mailto:posnet.destek@ykb.com). 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.

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.

#### 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>

```

posnetRequest – capt	
It is used to financialize the provision.	
posnetRequest	
mid	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.

#### 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>000000000000</pointAmount>
12.    <totalPoint>04860169</totalPoint>
13.    <totalPointAmount>000002430084</totalPointAmount>
14.  </pointInfo>
15. </posnetResponse>

```

#### posnetResponse – instInfo – pointInfo

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.

### 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%>

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. <b>If Hostlogkey is used, this field should not be included in xml.</b>
<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](#)

### 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

	<p>original transaction. If a financialized transaction with 3D Secure payment management is being refunded, it is necessary to complete 24 digits by padding "TDSC" in front of the orderId with 20 digits.</p> <p>Example: TDSCYKB_0000190526121122</p> <p>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></p>
--	---

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <hostlogkey>019799179990000191</hostlogkey>
4.   <authCode>991799</authCode>
5.   <instInfo>
6.     <inst1>00
7.     <amnt1>000000000000</amnt1>
8.   </instInfo>
9.   <pointInfo>
10.    <point>00000000</point>
11.    <pointAmount>000000000000</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 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.
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>
```

#### 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>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.

##### pointInquiry

<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>000000000000</totalPointAmount>
8.   </pointInfo>
9. </posnetResponse>
```

#### posnetResponse – pointInfo

Points information is accessed at the end of points inquiry transaction.

##### 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>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.

<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>pointUsage</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>expDate</b>	Credit card expiry date - In the following format: YY MM
<b>orderId</b>	Order number

#### 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>000009992746</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.

#### **pointInfo**

<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

### 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

## 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.

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

```

#### posnetResponse – instInfo - vftInfo

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>
```

### posnetRequest – vftTransaction

It is used for sale with delay interest.

#### posnetRequest

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

#### vftTransaction

ccno	Credit Card Number
cvc	Credit card security number - CVV2
expDate	Credit card expiry date - In the following format: YY MM
amount	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
currencyCode	Currency - "TL, US, EU"
installment	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
vftCode	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.
orderID	Order number. 24-digits alphanumeric
koiCode	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>019960022290000191</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.

#### 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

#### vftInfo

<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%>

#### vftReturn

<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. If Hostlogkey is used, this field should not be included in xml.

### 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>000000000000</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)

<b>posnetRequest</b>	
<b>mid</b>	YKB Merchant Number <%MERCHANT_ID%>
<b>tid</b>	YKB Merchant Terminal Number <%TERMINAL_ID%>
<b>koiCampaignQuery</b>	
<b>ccno</b>	Credit Card Number

#### Response Example

```

1. <posnetResponse>
2.   <approved>1</approved>
3.   <koiInfo>
4.     <code>1</code>
5.     <message>Sanal 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.

<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>koiInfo</b>	
<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>Sanal 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-kartlar>

### 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. Giril
   mezse false kabul edilir.
7.   -->
8.   <trioSingle noWarranty="false">
9.     <ccno>6037978246464011</ccno>
10.    <expDate>2511</expDate>
11.    <cvc>000</cvc>
12.    <orderID>YKB_TST_190703093100_024</orderID>
13.    <amount>175</amount>
```

```

14.      <currencyCode>TL</currencyCode>
15.      <termDayCount>20</termDayCount>
16.    </trioSingle>
17.  </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
<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. Giril
mezse false kabul edilir.
7.     -->
8.     <trioFixed noWarranty="false">
9.         <ccno>6037978246464011</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
<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. Giril
7.     mezse false 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

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

##### trioMultiple

ccno	Credit Card Number
expDate	Credit card expiry date - In the following format: YY MM
cvc	Credit card security number - CVV2
orderId	Order number. 24-digits alphanumeric
amount	Shopping amount - in Kurus Ex: 12.34 TL should be set as 1234.
currencyCode	Currency - "TL, US, EU"
installment	Number of installments "00" should be used for Cash Transaction. "02" should be used for a transaction in installments.
termDayCount	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>000000000333</amnt1>
8.   </instInfo>
9. </posnetResponse>
```

#### posnetResponse – instInfo

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.
<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>020527826090000191</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.

<b>posnetRequest</b>	
<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>

### 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</inst1></instInfo>
7.      -->
8.      <trioDate></trioDate>
9.      <trioAmount>000000000175</trioAmount>
10.     <ins1></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>	

#### 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.

## 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.

#### posnetRequest

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

#### trioAvailableLimitInq

<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>0000000000632347</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>6700972667</mid>
4.   <tid>67510491</tid>
5.   <agreement>
6.     <orderID>TDS_YKB_0000191010111730</orderID>
7.   </agreement>
8. </posnetRequest>
```

#### posnetRequest – agreement

It is used for inquiry the transaction result with the order number.

#### posnetRequest

<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

#### Response Example

```
1. <?xml version="1.0" encoding="utf-16"?>
2. <posnetResponse>
3.   <approved>1</approved>
4.   <transactions>
5.     <transaction>
6.       <orderID>TDS_YKB_0000191010111730</orderID>
7.       <ccno>4506 34** **** 4637</ccno>
8.       <amount>1,16</amount>
9.       <currencyCode>TL</currencyCode>
10.      <authCode>504289</authCode>
11.      <tranDate>2019-10-10 11:21:14.281</tranDate>
12.      <state>Sale</state>
13.      <txnStatus>1</txnStatus>
14.      <hostLogKey>021450428990000191</hostLogKey>
15.    </transaction>
16.  </transactions>
17. </posnetResponse>
```

#### posnetResponse – transactions – transaction

The query results are reached after the query.

#### 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.

transactions – transaction	
<b>orderId</b>	Order number. 24-digits alphanumeric
<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 Currency Ex: 12.34 TL should be set as 12,34
<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>	The unique Id on the system.
<b>txnStatus</b>	Shows the success status of transaction. 1: Successful transaction (transaction amount will be reflected to the merchant and cardholder) 0: Failed or canceled transaction (transaction amount will not be reflected to the merchant and cardholder)

## 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 değerleri 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>, <pointUsage>, <auth>, <capt>, <return>, <unmatchedreturn>, <reverse>, Sales with Delay Interest <vftTransaction>, return of Sales with Delay Interest <vftReturn>, Single Trio Payment <trioSingle>, Multi Trio Payment <trioMultiple>, Fixed Trio Payment <trioFixed>, Financialisation 3D Secure Transaction <oosTranData>. It is necessary to include <ngp> xml tag as childNode in <posnetRequest> node.

```

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>onprovizyonu00000000030</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>87F491ACD24EAE64B519980F0B1BC7547BE4A7C5C614DC3A8CA3FC41B180EE7765
851B081AAE61221956C0C68B0AD69307B4386C7FCE451C272264251BD72BFCBA0A96A197C38C6CD39DD4
42BC179FF098824AFA15B1BB320AD15DA2FB588ECC81B11A26D13764A57B57B49C4CA1BD5D46FA7E60EE
D480C944AE0817</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>

```

## Error codes

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

Error Code	What needs to be done
<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>	It should be checked that the hostname parameter is set correctly and has internet connection. By establishing a telnet collection to Hostname

	<p>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 the test support group.</p>
<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_OWNIPI</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 kuruş 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>	<p>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.</p> <p>Entering the installment parameter 00 or 01 also causes this error in operations that must be in installments (for example; VFT).</p>
<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:posnet.support@yapikredi.com.tr">posnet.support@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>

0007	BANKANIZI ARAYIN 0007 (CALL YOUR BANK)	The card may be blocked/stolen/lost (special case).
0012	RED-GEÇERSİZ İŞLEM (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 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.</p>
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)	<p>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.</p> <p>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</p>

		card is a trusted card), you should call the YKB Merchant Operations Service (444 0 448).
<b>0051</b>	RED-YETERSİZ BAKIYE 0051 (REJECT - INSUFFICIENT BALANCE)	The card has insufficient balance. Call the bank that issued the card.
<b>0053</b>	BANKANIZI ARAYIN 0053 (CALL YOUR BANK)	This account isn't found.
<b>0054</b>	RED-ONAYLANMADI 0054 (REJECT- WASN'T APPROVED)	The credit card is expired.
<b>0057</b>	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).
<b>0057</b>	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	<p>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> .</p> <p>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.</p>
<b>0122</b>	DATABASE DE İSTENİLEN 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.



		<p>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.</p>
<b>0123</b>	ORJINAL ISLEM BULUNAMADI (ORIGINAL TRANSACTION CANNOT BE FOUND)	<p>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.</p> <p>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 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	<p>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.</p>
<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.X.X (ERROR IN MID, TID, IP)	<p>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> <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 posnet.destek@ykb.com with mid/tid as indicated in the error message XXXX.</p>
<b>0150</b>	PAKET HATALI (ERROR IN PACKET)	Wrong CVC number is used. This error is received if XXX used in the live environment is used in the test environment. In the live environment, your customer must enter the CVC code. In addition, entering a meaningless CVC (such as xxx) other than XXX in the test environment will also cause this error.
<b>0150</b>	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.
<b>0200</b>	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).
<b>0205</b>	GECERSIZ TUTAR (INVALID AMOUNT)	This error is received under the following conditions:

		<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>
<b>0211</b>	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.
<b>0217</b>	GEÇERSİZ İŞLEM STATÜSÜ (INVALID TRANSACTION STATUS)	Stolen card. It is necessary to notify YKB about the user name and card number.
<b>0220</b>	İPTAL İŞLEMİ YAPILMIŞ (CANCELLATION COMPLETED)	This error is received when you try to cancel again a 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>	İŞLEM İPTALI YAPILMIŞ (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>	İŞLEM HENÜZ FİNANSALLAŞMAMIS (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 İŞLEMİ 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>	FİNANSAL İŞLEM YAPILMIŞ (FINANCIAL TRANSACTION COMPLETED)	Financialization is completed. If you want to cancel a provision, financialization must be canceled first.

## Steps for Going Live

After completing your tests in the test environment, you must send your request to go live to [posnet.support@yapikredi.com.tr](mailto:posnet.support@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.
3. 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.

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://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	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>	Nazim Sezer - Virtual Pos and Campaign Application Development
12.06.2019	2.0.1.1	Reconciliation services have been added. <ul style="list-style-type: none"> <li>• Transaction Status Inquiry</li> <li>• Daily Transaction Report Inquiry</li> </ul>	Nazim Sezer - Virtual Pos and Campaign Application Development
03.07.2019	2.0.1.2	Trio services have been added.	Nazim Sezer - Virtual Pos and Campaign Application Development
12.07.2019	2.0.1.3	New generation payment services added	Nazim Sezer - Virtual Pos and Campaign Application Development
11.10.2019	2.0.1.5	Newly added fields to the response of the agreement method.	Tolga Akgün - Virtual Pos and Campaign Application Development