The Account Information

Service Manual

AIS (v2)

Change log

Date of publishing	Version	Date of effectiveness	Description
3.12.2018	1		First document version
22.3.2019	2		Point 1 Change of link and deleting information of previous COBS version
22.3.2019	2		Tables 1 – 24: update according COBS v1.2 and changes within KB solution
22.3.2019	2		Deleting of all structure details
22.3.2019	2		Update of all table responses
29.4.2019	3	10.8.2019	Table 24: new column for card transactions changes added
29.4.2019	3	10.8.2019	Update of all request headers: TPP-Name, TPP-Identification
29.4.2019	3	10.8.2019	Change in chapter 4., table 13, parameter "order" – order of transactions allowed from 25 May 2019
7.8.2019	3	10.8.2019	Minor change in chapter 4., table 20 Domestic payment – occurrence of elements: • status – can now contain value PDNG for transactions in pending status. Added description card holds identification. • bookingDate.date – now not mandatory (in case of transactions in PDNG status)
18.10.2019	4	18.11.2019	Chapter 5 - Adding a new resource List of client's standing orders Chapter 4: • Extending the maximum transaction duration from 90 days up to 2 years
8.12.2019	5	8.2.2020	Chapter 2 – Adding name of account holder into resource List of Client's Payment Accounts
17.2.2020	6	21.2.2020	List of Client's Payment Accounts – Resource Characteristics – ownersNames as Array
2.7.2020	7	2.7.2020	Error reporting
17.9.2021	8	20.9.2021	New chapter added 6. Standing order details
02.05.2022	9	12.05.2022	Addition of error codes: • Error codes defined for the GET service – List of Client's Payment Accounts • Error codes defined for the GET service – Payment account balance • Error codes defined for the GET service – Payment account transaction history
30.06.2022	10	11.07.2022	Komerční banka begins the migration of the our PSD2 API to the new version, namely the PSD2 API account information service AIS v2. This version mainly changes the technological improvement without major functional impacts. PSD2 API account information service AIS v1 will be switched off on 12.10.2022.
20.07.2023	11	20.10.2023	Updated validity of SCA/refresh token according to EU Regulation 2022/2360 for 180 days - chapter 1. Added information on Multicurrency accounts - in chapters 1 to 3.

Differences from Czech Open Banking Standard are highlighted in yellow colour. Shaded green are the latest changes in ver. 11.

Contents

1. The Account Information Service (AIS) 5

Error reporting 5

2. List of Client's Payment Accounts – Resource Characteristics 7

Query parameters of the request 7

Request header parameters 7

Response header parameters 8

Error codes defined for the GET service – List of Client's Payment Accounts 8

Elements of the response message – List of Client's Payment Accounts 9

3. Client's Payment Account Balance – Resource Characteristics 12

Path parameter of the request 12

Query parameters of the request 12

Request header parameters 12

Account Balance response elements 13

Response header parameters 15

Error codes defined for the GET service – Payment account balance 15

4. Client's payment account transaction history – Resource Characteristics 17

Path parameter request 17

Query parameters of the request 17

Request header parameters 18

Response header parameters 18

Error codes defined for the GET service – Payment account transaction history 18

Client's payment account transaction history – response elements 19

Interest – occurrence of elements 34

Charge/fee – occurrence of elements 37

Domestic payment – occurrence of elements 40

SEPA payment – occurrence of elements 43

Foreign payment – occurrence of elements 49

Cash – occurrence of elements 53

Payment card transaction – occurrence of elements 56

5. List of Client's Standing Orders 61

Query parameters of the request 61

Request header parameters 61

Response header parameters 62

Error codes defined for the GET service – Standing orders 62

List of Client's Standing Orders response elements: 63

6. Standing order details 67

Query parameters of the request 67

Request header parameters 67
Response header parameters 67
Error codes defined for the service GET - Standing order details: 68
Standing order details report elements 68

The Account Information Service (AIS) Manual

1. The Account Information Service (AIS)

The following information is provided as part of the AIS service:

- List of client's accounts, in the case of a Multicurrency account, we display the currency subfolders as separate accounts, but with the same IBAN, and different currency and account id.
- Account balance:
    - Available balance – CLAV (ClosingAvailable)
    - Current balance – PRCD (PreviouslyClosedBooked)

  We will return the balance of the Multicurrency account only for the currency subfolder, which is identified by the account id.

  We do not refund one total Multicurrency Account balance (for all subfolders).
- Transaction history for up to 24 months - as required in PSD2 RTS 2018/389 Art. 10 (updated EU Regulation 2022/2360), requests for transaction history older than 90 days can only be performed under strong client authentication. According to KB, this regulation complies with a request that was made within 5 minutes of issuing a refresh token (which is issued under strong client authentication).

  We return the history of transactions for a Multicurrency account only for the currency subfolder that is identified by the account id, if the relevant currency was specified in the request. Otherwise, we return the transaction history for all currency subfolders.
- List of client's standing orders - compared to the ČOBS standard, the list is extended with information about the beneficiary's account (creditorAccount) and information about the variable symbol of the payment.

The client (PSU) authorizes the calling of the services that remain valid for 180 days. Consent to providing information to a third party can be revoked by the client in its internet banking at any time within 180 days, otherwise it remains valid for the full 180 days.

Komerční banka has based its approach on the unified structure and format of information defined by the Czech Banking Association in the Czech Open Banking Standard. The differences between the KB implementation mode and the published standard are described in this document and they are marked in yellow colour.

The information provided through API Open Banking is in both Czech and English.

The allowed character set is based exclusively on the SWIFT character set (i.e., exclusively without diacritics). Only one query can be sent and processed during a single call.

A paginated list can be requested for all AIS services. The page and size query parameters are used for this query. Each source that allows for requesting a paginated list has this feature explicitly listed. The return responses to these API calls will then contain the parameters of a paginated response.

Error reporting

Reporting quarantined errors or calling them always takes place via the mailbox api@kb.cz. The e-mail sent must contain the following information, in case the required information is missing, it will not be possible to process the query or error.

PSD2 API: CZ, SK

Environment: Sandbox, Production

Whether it was called from FE Sandbox incl. the type and version of the browser used or, in the case of a BE call, the name and version of the program for the BE call

Request type

Date and time of the call

IP address

The error and its most accurate description, which can be supplemented with the appropriate screenshot

Without the above values, it is not possible to solve the reported error.

2. List of Client's Payment Accounts – Resource Characteristics

URI:	/my/accounts{?size,page,sort,order}
HTTP Metoda:	GET
Request URL:	https://api.kb.cz/serverapi/aisp/v2/my/accounts
Authorization:	the request requires an authorisation by the user/client as part of the API call.
Certification:	the request requires the use of the third party qualified certificate as part of establishing two-way TSL communication with the server. The third party is identified by verifying the validity and content of this certificate.
Pagination:	yes
Sorting:	no
Filtering:	no

Supported encoding: `charset=UTF-8`

Query parameters of the request

Parameter	Type	Manda-tory	Purpose
size	Int	No	Pagination. A number of records on the page.
page	Int	No	Pagination. The required page. + Default: 0.
sort	Text	No	A list of fields for sorting separated with a comma and sorted in order of importance. Not supported.
order	Text	No	A list of types of sorting (ASC, DESC) separated with a comma. The order matches the order of the fields in the 'sort' parameter. Not supported.

Example of the API call curl:

```
curl -X GET --header 'Accept: application/json' --header 'x-request-id: 12345' --header 'Authorization: Bearer INPUT_ACCESS_TOKEN_HERE' 'https://api.kb.cz/serverapi/aisp/v1/my/accounts?size=2&page=1&sort=iban'
```

Request header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
API-key	Text	Ne	An optional string issued to a communicating third party as the call identifier of that party primarily serving as the configuration element of communication. Not supported.
Authorization	Text	Yes	A parameter used for forwarding the authenticated user's access token along with its type.
TPP-Name	Text	Yes	The name of the original TPP that created the request. Eg. 'Star corporation, a.s.'
TPP-Identification	Text	No	The identification (licence number) of the original TPP that created the request. Eg. 'CZ013574-15'
x-request-id	Text	No	A unique identification of the caller's each particular query. The value of this parameter should therefore be generated randomly, and the individual x-request-ids of the same caller within a short time interval should not be identical. This parameter service returns responses to the calling system within response headers.

Example of request headers:

```
{
```

```
  "Accept": "application/json",
  "x-request-id": "12345",
}
```

Response header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
x- request -id	Text	No	Returns the original request id of the API call.

Example of response headers:

```
{
Date: Mon, 18 Mar 2019 09:57:26 GMT
x-request-id: 85638
x-response-id: urn:uuid:521c7fb7-b5eb-412f-8b7d-54b61e35f07c
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET
Access-Control-Allow-Headers: authorization,Access-Control-Allow-Origin,Content-Type,SOAPAction
Content-Language: cs
Content-Type: application/json; charset=utf-8
Keep-Alive: timeout=60, max=10000
Connection: Keep-Alive
Transfer-Encoding: chunked
Strict-Transport-Security: max-age=16070400; includeSubDomains }
```

Error codes defined for the GET service – List of Client's Payment Accounts

HTTP status code	Error code	Purpose
401	UNAUTHORISED	Invalid/missing access token = the user has not been authenticated
401	UNAUTHORISED	Invalid/ missing certificate = the provider has not been authenticated
404	PAGE_NOT_FOUND	The query concerns a non-existent page
400	PARAMETER_INVALID	The parameter value is not valid
400	NARR	[AccessDenied - Unauthorized access to account]
400	NARR	[AccessDenied - User is not in active state]
400	NARR	[AccessDenied - Debtor payment account is not available in internet banking]
400	AC12	[InvalidAccountType]
400	NARR	[AccessDenied - Client is not in active state]

Example of response headers:

```
{"errors": [{
  "error": "ERR_CODE_401",
  "message": "UNAUTHORISED"
}]}
```

Komerční banka provides the information about the payment account and transaction history in the following structure:

Elements of the response message – List of Client's Payment Accounts

Level	Message e lement	Occurence	Payment type	Format type	Presentation
+	accounts	[1..1]	AISP	±	Collection of client's accounts
++	id	[1..1]	AISP	Text	API Identifier of the payment account
++	identification	[1..1]	AISP	±	Debtor's account identification
+++	iban	[1..1]	AISP	IBAN2007Identifi er	IBAN
+++	other	[0..1]	AISP	Max35Text	Another identifier, e.g. debtor's account number
++	currency	[0..1]	AISP	CurrencyCode, ISO 4217	Debtor's account currency
++	servicer	[1..1]	AISP	±	The identification information of the bank the account is served by - bankCode - countryCode - bic
+++	bankCode	[0..1]	AISP	Text	Bank code in national format
+++	countryCode	[0..1]	AISP	CountryCode, ISO 3166	Bank's country code
+++	bic	[0..1]	AISP	Max35Text	Bank's BIC
++	nameI18N	[0..1]	AISP	Text	Account name
++	productI18N	[0..1]	AISP	Text	Product name
++	ownersNames	[0..1]	AISP	Array of Max100Text	Array of Surname/s and first name/s of the account holder for consumers or trade name for commercial clients.

JSON – example of a response:

```
{
  "pageNumber": 1,
  "pageCount": 11,
```

```
  "pageSize": 1,
  "nextPage": 2,
  "accounts": [  {
    "id":        "Rn133fW7M2MJq5owXc5YbwwjR7BC5UCBzBnovdhUU9mLyMXp-
NY8WXDVXVjePM9NJAVersuceCtoUpqxJzJQlw",
    "identification":     {
      "iban": "CZ7301000901148109130227",
      "other": 901148109130227
    },
    "currency": "USD",
    "servicer":     {
      "bankCode": "0100",
      "countryCode": "CZ",
      "bic": "KOMBCZPPXXX"
    },
    "nameI18N": "Muj hlavni osobni ucet",
    "productI18N": "Běžný účet/Current account",
    "ownersNames": [
        "Novak Jan"
    ]
  }
 ]
}
```


Multi-currency accounts with currency subfolders will break down into individual accounts (per currency).

Each currency subfolder will have its own unique "**id**" in the response, which will consist of 1.) IBAN account and 2.) currency.

JSON - example of Response for a list of accounts:


```
 "accounts":
{
  "pageNumber": 0,
  "pageCount": 1,
  "pageSize": 20,
  "nextPage": 0,
  "accounts":   [
        {
      "id":
"cjJDa1k2OTlrZEtocGFpdmdhcVdOWThYL1VtMnlySjFMQS96dE16MTI1UUdDWW9ldjVYa1NzR2NkTGNI
Rk9QODdrNzYrc1RaT01kU0VBdHFrakVVUlE9PQ",
      "identification":       {
        "iban": "CZ7701000000000102163257",
        "other": "102163257"
      },
      "currency": "EUR",
      "servicer":       {
        "bankCode": "0100",
        "countryCode": "CZ",
        "bic": "KOMBCZPPXXX"
      },
      "productI18N": "Běžný účet/Current account",
      "ownersNames": ["NOVAK JAN"]
    },
        {
```

```
      "id":
"b1Z1UWFoazd0dUs2WFdYVjVZNERnVW1QR1Voa0h6RzNsZS9HQXlZNzVma2RGSFdCZUN3MEFFZUloa
XBQWFRkL3FLSWlhQk8xRzZKS1B2Yzk5aTFuaGc9PQ",
      "identification":      {
        "iban": "CZ7701000000000102163257",
        "other": "102163257"
      },
      "currency": "CZK",
      "servicer":      {
        "bankCode": "0100",
        "countryCode": "CZ",
        "bic": "KOMBCZPPXXX"
      },
      "productI18N": "Běžný účet/Current account",
      "ownersNames": ["NOVAK JAN"]
    },
      {
      "id":
"STRSYkc1eWswOUl6VGNOeU5EYUpCWWUvV3pSVUM1cTl3NkxoZndNVnUrRzFmKzhsbTV3M0hpanBF
NElxMDlFWHdCZFZNb2xlYmV2VllYNWk2dGxMcGc9PQ",
      "identification":      {
        "iban": "CZ7701000000000102163257",
        "other": "102163257"
      },
      "currency": "USD",
      "servicer":      {
        "bankCode": "0100",
        "countryCode": "CZ",
        "bic": "KOMBCZPPXXX"
      },
      "productI18N": "Běžný účet/Current account",
      "ownersNames": ["NOVAK JAN"]
  },
```

3. Client's Payment Account Balance – Resource Characteristics

A balance of the client's particular account according to the account's reference id.

For a **Multicurrency account**, the "currency" parameter in the header parameters will not affect the content of the Response, but if TPP fills it, we will validate that the currency corresponds to the currency hashed in the account id. If it is different, we will return an error: 400 - AC09 – InvalidAccounCurrency. We will return the balance only for the currency subfolder, which is hashed in the account id.

We do not return the total balance of the Multicurrency Account (for all subfolders).

URI:	/my/accounts/{id}/balance{?currency}
HTTP Method:	GET
Request URL:	https://api.kb.cz/serverapi/aisp/v2/my/accounts/{id}/balance{?currency}
Authorization:	the request requires an authorisation by the user/client as part of the API call.
Certification:	the request requires the use of the third party qualified certificate as part of establishing two-way TSL communication with the server. The third party is identified by verifying the validity and content of this certificate.
Pagination:	no
Sorting:	no
Filtering:	no

Supported encoding: `charset=UTF-8`

Path parameter of the request

Parameter	Type	Manda-tory	Purpose
id	Text	Yes	Client's account system identifier in the hashed format.

Query parameters of the request

Parameter	Type	Manda-tory	Purpose
currency	Text	Ne	The required currency of the account. It must match the currency in which the account is held with KB.

Example of the API call curl:

```
curl -X GET --header 'Accept: application/json' --header 'x-request-id: 12345' --header 'Authorization: Bearer INPUT_ACCESS_TOKEN_HERE ' 'https://api.kb.cz/serverapi/aisp/v1/my/accounts/C2D2DDBCA5415621A34BB1BB234DC1322EA641A3/balance'
```

Request header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
API-key	Text	Ne	An optional string issued to a communicating third party as the call identifier of that party primarily serving as the configuration element of communication. Not supported.
Authorization	Text	Yes	A parameter used for forwarding the authenticated user's access token along with its type.
TPP-Name	Text	Yes	The name of the original TPP that created the request. Eg. 'Star

			corporation, a.s.'
TPP-Identification	Text	No	The identification (licence number) of the original TPP that created the request. Eg. 'CZ013574-15'
x-request-id	Text	No	A unique identification of the caller's each particular query. The value of this parameter should therefore be generated randomly, and the individual x-request-ids of the same caller within a short time interval should not be identical. This parameter service returns responses to the calling system within response headers.

Example of request headers:

```
{
  "Accept": "application/json",
  "x-request-id": "12345"",
}
```

Account Balance response elements

Level	Message e lement	Occurence	Payment type	Format type	Presentation
+	balances	[1..1]	AISP	±	A collection of the client's payment account balance
++	type	[1..1]	AISP	±	A designation of the balance type to which the information about the balance pertains
+++	codeOrProprietary	[1..1]	AISP	±	
++++	code	[1..1]	AISP	Balance type	Balance type codes
++	creditLine	[0..1]	AISP	±	An amount of the agreed authorised overdraft/overdraft facility
+++	included	[0..1]	AISP	Boolean	Indicates whether the allowed debit is included in balance
+++	amount	[0..1]	AISP	±	An amount of the agreed authorised overdraft
++++	value	[0..1]	AISP	Number	An amount of the agreed authorised overdraft
++++	currency	[0..1]	AISP	Text	The currency corresponds to the currency of the account to which the generated statement pertains
++	amount	[1..1]	AISP	±	The value/amount of the account balance, depending on the balance type. The currency corresponds to the currency of the account to which the generated statement pertains.

+++	value	[1..1]	AISP	Number	Account balance amount
+++	currency	[1..1]	AISP	Text	The currency corresponds to the currency of the account to which the generated statement pertains
++	creditDebitIndicator	[1..1]	AISP	Text	An indicator showing whether the balance of the account to which the generated statement pertains is greater or less than zero
++	date	[1..1]	AISP	±	The date (and time) of the balance of the account to which the generated statement pertains. The format corresponds to the ISO 8601.
+++	dateTime	[1..1]	AISP	Text	The date (or date and time) of the balance as per the ISO 8601

JSON – example of a response:


```
{"balances": [
    {
    "type": {"codeOrProprietary": {"code": "CLAV"}},
    "creditLine":      {
      "included": false,
      "amount":        {
        "value": 0,
        "currency": "USD"
      }
    },
    "amount":      {
      "value": 6157.08,
      "currency": "USD"
    },
    "creditDebitIndicator": "CRDT",
    "date": {"dateTime": "2019-03-18T11:23:42.210+01:00"}
  },
    {
    "type": {"codeOrProprietary": {"code": "PRCD"}},
    "creditLine":      {
      "included": false,
      "amount":        {
        "value": 0,
        "currency": "USD"
      }
    },
    "amount":      {
      "value": 6157.08,
      "currency": "USD"
    },
    "creditDebitIndicator": "CRDT",
    "date": {"dateTime": "2019-03-18T11:23:42.210+01:00"}
```

```
  }
]}
```

Return types of balances in the "codeOrProprietary" parameter

Code	Description
PRCD	PreviouslyClosedBooked – initial balance
CLAV	Closing available – available balance
CLBD	ClosingBooked – closing balance. Not supported.

Return types of balances in the "creditDebitIndicator" parameter

Code	Description
DBIT	Balance < 0
CRDT	Balance > 0 or Balance = 0

Response header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
x- request -id	Text	No	Returns the original request id of the API call.

Example of response headers:

```
{
Date: Mon, 18 Mar 2019 10:23:42 GMT
x-request-id: 51518
x-response-id: urn:uuid:4d9a8ec7-d8c1-4495-8c6e-7d864f39ce45
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET
Access-Control-Allow-Headers: authorization,Access-Control-Allow-Origin,Content-Type,SOAPAction
Content-Language: cs
Content-Type: application/json; charset=utf-8
Keep-Alive: timeout=60, max=10000
Connection: Keep-Alive
Transfer-Encoding: chunked
Strict-Transport-Security: max-age=16070400; includeSubDomains
}
```

Error codes defined for the GET service – Payment account balance

HTTP status kód	Error kód	Účel
401	UNAUTHORISED	Invalid/missing access token = the user has not been authenticated
401	UNAUTHORISED	Invalid/missing certificate = the provider has not been authenticated
404	ID_NOT_FOUND	Invalid/unknown account ID

400	AC09	[InvalidAccountCurrency] – in case of multicurrency accounts, or the exchange list shows am unsupported currency
400	NARR	[AccessDenied - Unauthorized access to account]
400	NARR	[AccessDenied - User is not in active state]
400	NARR	[AccessDenied - Debtor payment account is not available in internet banking]
400	AC12	[InvalidAccountType]
400	NARR	[AccessDenied - Client is not in active state]

JSON – example of an error message body:

```
{"errors": [{
  "error": "AC09",
  "message": "InvalidAccountCurrency"
}]}
```

4. Client's payment account transaction history – Resource Characteristics

Paginated list of transactions of the selected client account.

For **Multicurrency accounts** with currency subfolders, we return only those transactions whose currency corresponds to the currency specified in the request for the relevant account id (if there are no transactions in the given subfolder, an empty list is returned). If the currency field is not filled in the request, we will return the transactions of all currency subfolders, i.e. the history of transactions for the entire account.

URI:	/my/accounts/{id}/transactions{?fromDate,toDate,currency,size,page,sort,order}
HTTP Method:	GET
Request URL:	https://api.kb.cz/serverapi/aisp/v2/my/accounts/{id}/transactions{?fromDate,toDate,currency,size,page,sort,order}
Authorization:	the request requires an authorisation by the user/client as part of the API call.
Certification:	the request requires the use of the third party qualified certificate as part of establishing two-way TSL communication with the server. The third party is identified by verifying the validity and content of this certificate.
Pagination:	yes
Sorting:	no
Filtering:	no

Supported encoding: `charset=UTF-8`

Path parameter request

Parameter	Type	Manda-tory	Purpose
id	Text	Ano	Client's account system identifier.

Query parameters of the request

Parameter	Type	Manda-tory	Purpose
currency	Text	Ne	The required currency of the account. It must match the currency in which the account is held with KB.
fromDate	Text	Ne	The initial date and time of the required transaction history. ~~According to ISO 8601~~. The date must be in the YYYY-MM-DD format without stating the specific time.
toDate	Text	Ne	The final date and time of the transaction history [including]. The date must be in the YYYY-MM-DD format without stating the specific time.
size	Int	Ne	Pagination. A number of records on the page.
page	Int	Ne	Pagination. The required page. + Default: 0
sort	Text	Ne	A list of fields for sorting separated with a comma and sorted in order of importance. Not supported.
order	Text	Ne	A list of types of sorting (ASC, DESC) separated with a comma. The order matches the order of the fields in the 'sort' parameter.

Transaction order is supported only by date, default order is DESC (from newest to oldest).

Example of the API call curl:

```
curl -X GET --header 'Accept: application/json' --header 'x-request-id: 12345' --header 'Authorization: Bearer INPUT_ACCESS_TOKEN_HERE ' 'https://api.kb.cz/serverapi/aisp/v1/my/accounts/CZ9501000000001234567899/transactions'
```

Request header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
API-key	Text	Ne	An optional string issued to a communicating third party as the call identifier of that party primarily serving as the configuration element of communication. Not supported.
Authorization	Text	Yes	A parameter used for forwarding the authenticated user's access token along with its type.
TPP-Name	Text	Yes	The name of the original TPP that created the request. Eg. 'Star corporation, a.s.'
TPP-Identification	Text	No	The identification (licence number) of the original TPP that created the request. Eg. 'CZ013574-15'
x-request-id	Text	No	A unique identification of the caller's each particular query. The value of this parameter should therefore be generated randomly, and the individual x-request-ids of the same caller within a short time interval should not be identical. This parameter service returns responses to the calling system within response headers.

Example of request headers:

```
{
  "Accept": "application/json",
  "x-request-id": "12345",
}
```

Response header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
x- request -id	Text	No	Returns the original request id of the API call.

Example of response headers:

```
{
Date: Mon, 18 Mar 2019 10:28:41 GMT
x-request-id: 17549
x-response-id: urn:uuid:df1cb535-6332-4184-8346-e0742c43706e
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET
Access-Control-Allow-Headers: authorization,Access-Control-Allow-Origin,Content-Type,SOAPAction
Content-Language: cs
Content-Type: application/json; charset=UTF-8
Keep-Alive: timeout=60, max=10000
Connection: Keep-Alive
Transfer-Encoding: chunked
Strict-Transport-Security: max-age=16070400; includeSubDomains }
```

Error codes defined for the GET service – Payment account transaction history

HTTP status kód	Error kód	Účel

401	UNAUTHORISED	Invalid/missing access token = the user has not been authenticated
401	UNAUTHORISED	Invalid/missing certificate = the provider has not been authenticated
404	ID_NOT_FOUND	Invalid/unknown account ID
404	PAGE_NOT_FOUND	The query concerns a non-existent page
400	PARAMETER_INVALID	The parameter value is not valid
400	AC09	[InvalidAccountCurrency] – in case of multicurrency accounts ***)**, or the exchange list shows an unsupported currency
400	DT01	[InvalidDate] Invalid date
400	NARR	[AccessDenied - Unauthorized access to account]
400	NARR	[AccessDenied - User is not in active state]
400	NARR	[AccessDenied - Debtor payment account is not available in internet banking]
400	AC12	[InvalidAccountType]
400	NARR	[AccessDenied - Client is not in active state]
400	NARR	Request for transaction history without SCA exceeded 90 days "message": "ACCESS_TOKEN_EXPIRED – note.- an error is recorded here, it is not an access token, but an expired refresh token.

***)** For **Multicurrency accounts**, if the currency field contains a different currency than the account id field, error 400 - AC09 – InvalidAccounCurrency will be returned.

Client's payment account transaction history – response elements

Level	Message element	Format type	Presentation
+	entryReference	Max35Text	Payment ID number assigned by the bank.
+	amount	Amount	The payment currency expressed in the currency of the account to which the generated statement relates.
++	value	Amount	
++	currency	CurrencyCode	
+	creditDebitIndicator	CreditDebitCode	Indicator showing whether the payment is debited or credited to the account. A debited / credited payment is indicated by one of the

			below codes. DBIT: debited payment; CRDT: all other cases.
+	reversalIndicator	TrueFalseIndicator	Indicator showing whether or not it is a reversal (cancellation). It is expressed using one of the below codes: true = reversal, false = no reversal.
+	status	Code	Status of the item in the account (credited or debited payment) from the bank's point of view. The statement only shows booked items (BOOK constant) or blocked items (PDNG constant).
+	bookingDate	±	Date of the payment processing/entering in the accounts (booking) by the bank in the ISODate or ISODateTime format, i.e. YYYY-MM-DD, or YYYY-MM-DDThh:mm:ss.sTZD.
++	date	ISODate	Date of the payment processing/entering in the accounts (booking) by the bank in the ISODate format, i.e. YYYY-MM-DD
+	valueDate	±	Maturity/value date of the payment in the ISODate or ISODateTime format, i.e. YYYY-MM-DD, or YYYY-MM-DDThh:mm:ss.sTZD.
++	date	ISODate	Maturity/value date of the payment in the ISODate format, i.e. YYYY-MM-DD
+	bankTransactionCode	±	Ancestor element to the bank transaction code as per the code list – see bankTransactionCode.
++	proprietary	±	Ancestor element to the bank transaction code as per the code list – see bankTransactionCode.
+++	code	Max35Text	Bank transaction code as per the code list – see bankTransactionCode.
+++	issuer	Max35Text	Identification of the bank transaction code list issuer, which takes its value from the Czech Banking Association list.
+	entryDetails	±	Turnover details. This level

			occurs only once in each given item
++	transactionDetails	±	Payment details. This level occurs only once in each given item.
+++	references	±	Set of references that unequivocally identify the payment.
++++	messageIdentification	Max35Text	Received payment identification entered by the client when initiating the payment, or a sequence of the given payment in the payment history statement.
++++	accountServicerReference	Max35Text	Bank reference assigned to the given payment, e.g. when initiated via direct banking services.
++++	paymentInformationIdentification	Max35Text	Other bank reference assigned to the payment allotted by the bank; in case of card payments, a payment card sequence number can be filled in, or alternatively the specific symbol.
++++	mandateIdentification	Max35Text	Payment identification entered by a third party, or alternatively the specific symbol.
++++	endToEndIdentification	Max35Text	Unique identification entered by the client who initiates the payment; it allows for an unequivocal identification of the payment and is forwarded, unaltered, throughout the payment chain, as the case may be, the variable symbol may be entered here.
++++	mandateIdentification	Max35Text	In case of SEPA Direct Debits (SDD), a Unique Mandate Reference is specified for a given SDD as a mandatory field [1..1].
++++	chequeNumber	Max35Text	In case of cheque transactions or card transactions, a cheque number or a card number with asterisks can be inputted here; respectively. The card number is always in the xxxxxxxxxxxx1234 format, where the standard

			only allows for digits from 0 to 9 (8-28 characters) i.e. without the xxxx substitution.
++++	clearingSystemReference	Max35Text	Code list value defined by the bank to identify the payment type, or a used payment type name. A card association can be identified here in case of card payments.
+++	amountDetails	+	Details concerning the payment amount, especially in case of a converse payment or Cashback.
++++	instructedAmount	+	Amount and currency of the payment expressed in the currency required by the client for the transfer. E.g., in the case of intrabank payments, it is the currency in which the debtor's account is denominated and, at the same time, the payment amount, if the client instructed that the payment should be made in the debtor's account currency.
+++++	amount	Amount	Amount and currency of the payment expressed in the currency in which the client's account is denominated, after the conversion of the amount required by the client for the transfer.
++++++	value	Amount	
++++++	currency	CurrencyCode	
++++	transactionAmount	+	Amount and currency of the payment for cumulated payments and Cashback.
+++++	amount	Amount	Amount and currency of the payment for cumulated payments and Cashback, where the total payment amount including Cashback is filled in.
++++++	value	Amount	
++++++	currency	CurrencyCode	
++++	counterValueAmount	+	Amount and currency of the payment expressed in the currency in which the

			client's account is denominated, after the conversion of the amount required by the client for the transfer.
+++++	amount	Amount	Ultimate amount and currency of the payment required by the client for the transfer.
++++++	value	Amount	
++++++	currency	CurrencyCode	
+++++	currencyExchange	+	Information about the used currencies and exchange rates.
++++++	sourceCurrency	CurrencyCode	Client's account currency (source/original currency / debtor's account currency for intrabank conversion payments).
++++++	targetCurrency	CurrencyCode	Currency of the payment (target currency / creditor's account currency for intrabank conversion payments).
++++++	exchangeRate	BaseOneRate	Exchange rate used for entering the payment into the accounts. One rate is filled-in only, even in the case of a cross conversion.
++++	proprietaryAmount	+	Amount of cash withdrawn via Cashback.
+++++	type	Max35Text	'CASHBACK' constant is to be filled in.
+++++	amount	Amount	Cashback amount and currency – only the portion of the payment withdraen in cash via Cashback.
++++++	value	Amount	
++++++	currency	CurrencyCode	
+++	charges	+	Information about charges.
++++++	bearer	Code	Identification of the bearer of the charges (OUR, SHA, BEN).
+++	relatedParties	+	Information about the debtor, debtor's account, ultimate debtor, and the creditor, creditor's account, ultimate creditor, as specified in the payment instruction.

++++	debtor	+	Information about the debtor. Depending on the direction of the payment, filled in for the counterparty.
+++++	name	Max140Text	Debtor's name.
+++++	postalAddress	+	Debtor's postal address.
++++++	streetName	Max70Text	Street name used in the debtor's postal address.
++++++	buildingNumber	Max16Text	Building number used in the debtor's postal address.
++++++	postCode	Max16Text	Postal code used in the debtor's postal address.
++++++	townName	Max35Text	Town name used in the debtor's postal address.
++++++	country	CountryCode	Country name used in the debtor's postal address.
++++++	addressLine	Max70Text	Debtor's postal address in an unstructured format.
+++++	identification	+	Identification of the debtor.
++++++	organisationIdentification	+	Unique identification of the debtor as an organization/legal person.
+++++++	bicOrBei	BICIdentifier	Identification of the debtor as an organization/legal person in the form of the BIC or BEI code.
+++++++	other	+	Other identification of the debtor as an organization/legal person.
++++++++	identification	Max35Text	Other identification of the debtor as an organization/legal person in an unstructured format.
++++++++	schemeName	+	Type of the code identifying the debtor as an organization/legal person.
+++++++++	code	Code	Type of the code identifying the debtor as an organization/legal person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the debtor as an organization/legal person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the debtor as an organization/legal person.

++++++	privateIdentification	+	Unique identification of the debtor as a natural person.
+++++++	other	+	Other identification of the debtor as a natural person.
++++++++	identification	Max35Text	Other identification of the debtor as a natural person in an unstructured format.
++++++++	schemaName	+	Type of the code identifying the debtor as a natural person.
+++++++++	code	Code	Type of the code identifying the debtor as a natural person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the debtor as a natural person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the debtor as a natural person.
++++	debtorAccount	+	Information about the debtor's account. Depending on the direction of the payment, filled in for the counterparty.
+++++	identification	+	Debtor's account type identification.
++++++	iban	IBAN2007Identifier	Debtor's account number in the IBAN international format.
++++++	other	+	Debtor's account number expressed in other/local format.
+++++++	identification	Max34Text	Value of debtor's account number expressed in other/local format.
+++++	currency	CurrencyCode	Debtor's account currency.
+++++	name	Max70Text	Debtor's account name.
++++	ultimateDebtor	+	Information about the ultimate debtor. Depending on the direction of the payment, filled in for the counterparty.
+++++	name	Max140Text	Ultimate debtor's name.
+++++	postalAddress	+	Ultimate debtor's postal address.
++++++	streetName	Max70Text	Street name used in the ultimate debtor's postal address.

++++++	buildingNumber	Max16Text	Building number used in the ultimate debtor's postal address.
++++++	postCode	Max16Text	Postal code used in the ultimate debtor's postal address.
++++++	townName	Max35Text	Town name used in the ultimate debtor's postal address.
++++++	country	CountryCode	Country name used in the ultimate debtor's postal address.
++++++	addressLine	Max70Text	Ultimate debtor's postal address in an unstructured format.
+++++	identification	+	Identification of the ultimate debtor.
++++++	organisationIdentification	+	Unique identification of the ultimate debtor as an organization/legal person.
+++++++	bicOrBei	BICIdentifier	Identification of the ultimate debtor as an organization/legal person in the form of the BIC or BEI code.
+++++++	other	+	Other identification of the ultimate debtor as an organization/legal person.
++++++++	identification	Max35Text	Other identification of the ultimate debtor as an organization/legal person in an unstructured format.
++++++++	schemeName	+	Type of the code identifying the ultimate debtor as an organization/legal person.
+++++++++	code	Code	Type of the code identifying the ultimate debtor as an organization/legal person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the ultimate debtor as an organization/legal person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the ultimate debtor as an organization/legal person.
++++++	privateIdentification	+	Unique identification of the ultimate debtor as a natural person.

+++++++	other	+	Other identification of the ultimate debtor as a natural person.
++++++++	identification	Max35Text	Other identification of the ultimate debtor as a natural person in an unstructured format.
++++++++	schemaName	+	Type of the code identifying the ultimate debtor as a natural person.
+++++++++	code	Code	Type of the code identifying the ultimate debtor as a natural person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the ultimate debtor as a natural person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the ultimate debtor as a natural person.
++++	creditor	+	Information about the creditor. Depending on the direction of the payment, filled in for the counterparty.
+++++	name	Max140Text	Creditor's name.
+++++	postalAddress	+	Creditor's postal address.
++++++	streetName	Max70Text	Street name used in the creditor's postal address.
++++++	buildingNumber	Max16Text	Building number used in the creditor's postal address.
++++++	postCode	Max16Text	Postal code used in the creditor's postal address.
++++++	townName	Max35Text	Town name used in the creditor's postal address.
++++++	country	CountryCode	Country name used in the c.
++++++	addressLine	Max70Text	Creditor's postal address in an unstructured format.
+++++	identification	+	Identification of the creditor.
++++++	organisationIdentification	+	Unique identification of the creditor as an organization/legal person.
+++++++	bicOrBei	BICIdentifier	Identification of the creditor as an organization/legal person in the form of the BIC or BEI code.

+++++++	other	+	Other identification of the creditor as an organization/legal person.
++++++++	identification	Max35Text	Other identification of the creditor as an organization/legal person in an unstructured format.
++++++++	schemeName	+	Type of the code identifying the creditor as an organization/legal person.
+++++++++	code	Code	Type of the code identifying the creditor as an organization/legal person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the creditor as an organization/legal person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the creditor as an organization/legal person.
++++++	privateIdentification	+	Unique identification of the creditor as a natural person.
+++++++	other	+	Other identification of the creditor as a natural person.
++++++++	identification	Max35Text	Other identification of the creditor as a natural person in an unstructured format.
++++++++	schemaName	+	Type of the code identifying the creditor as a natural person.
+++++++++	code	Code	Type of the code identifying the creditor as a natural person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the creditor as a natural person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the creditor as a natural person.
++++	creditorAccount	+	Information about the creditor's account. Depending on the direction of the payment, filled in for the counterparty.
+++++	identification	+	Creditor's account type identification.
++++++	iban	IBAN2007Identifier	Creditor's account number in the IBAN international

			format.
++++++	other	+	Creditor’s account number expressed in other/local format.
+++++++	identification	Max34Text	Value of creditor’s account number expressed in other/local format.
+++++	currency	CurrencyCode	Creditor’s account currency.
+++++	name	Max70Text	Creditor’s account name.
++++	ultimateCreditor	+	Information about the ultimate creditor. Depending on the direction of the payment, filled in for the counterparty.
+++++	name	Max140Text	Ultimate creditor’s name.
+++++	postalAddress	+	Ultimate creditor’s postal address.
++++++	streetName	Max70Text	Street name used in the ultimate creditor’s postal address.
++++++	buildingNumber	Max16Text	Building number used in the ultimate creditor’s postal address.
++++++	postCode	Max16Text	Postal code used in the ultimate creditor’s postal address.
++++++	townName	Max35Text	Town name used in the ultimate creditor’s postal address.
++++++	country	CountryCode	Country name used in the ultimate creditor’s postal address.
++++++	addressLine	Max70Text	Ultimate creditor’s postal address in an unstructured format.
+++++	identification	+	Identification of the ultimate creditor.
++++++	organisationIdentification	+	Unique identification of the ultimate creditor as an organization/legal person.
+++++++	bicOrBei	BICIdentifier	Identification of the ultimate creditor as an organization/legal person in the form of the BIC or BEI code.
+++++++	other	+	Other identification of the ultimate creditor as an organization/legal person.
++++++++	identification	Max35Text	Other identification of the

			ultimate creditor as an organization/legal person in an unstructured format.
++++++++	schemeName	+	Type of the code identifying the ultimate creditor as an organization/legal person.
+++++++++	code	Code	Type of the code identifying the ultimate creditor as an organization/legal person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the ultimate creditor as an organization/legal person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the ultimate creditor as an organization/legal person.
++++++	privateIdentification	+	Unique identification of the ultimate creditor as a natural person.
+++++++	other	+	Other identification of the ultimate creditor as a natural person.
++++++++	identification	Max35Text	Other identification of the ultimate creditor as a natural person in an unstructured format.
++++++++	schemaName	+	Type of the code identifying the ultimate creditor as a natural person.
+++++++++	code	Code	Type of the code identifying the ultimate creditor as a natural person in the form of a code as per the ISO code list.
+++++++++	proprietary	Max35Text	Type of the code identifying the ultimate creditor as a natural person in a free text format.
++++++++	issuer	Max35Text	Issuer of the code identifying the ultimate creditor as a natural person.
++++	proprietary	+	More detailed specification on the ATM at which the card transaction took place.
+++++	type	Max35Text	Specification of a bank's own / other bank's ATM.
+++++	party	+	Specification of the ATM's name / owner.

++++++	name	Max140Text	Specification of the ATM's name / location.
+++	relatedAgents	+	Information about the payment-related debtor's bank and creditor's bank.
++++	debtorAgent	+	Information about the debtor's bank. Depending on the direction of the payment, filled in for the counterparty.
+++++	financialInstitutionIdentification	+	Debtor's bank code expressed in the BIC / SWIFT international code format. Filled-in value (constant): KOMBCZPPXXX.
++++++	bic	BICIdentifier	Debtor's bank BIC / SWIFT code. Filled-in value (constant): KOMBCZPPXXX.
++++++	clearingSystemMemberIdentification	+	Debtor's bank code in a local format – either as a code or a text description.
+++++++	clearingSystemIdentification	+	Identification of the debtor's bank in the local payment system in which the debtor's bank operates.
++++++++	code	Code	Identification of the debtor's bank in the local payment system in which the debtor's bank operates, in the form of a payment system code.
++++++++	proprietary	Max35Text	Identification of the debtor's bank in the local payment system in which the debtor's bank operates, in an unstructured format of a descriptive text.
+++++++	memberIdentification	Max35Text	Debtor's bank code in a local format.
++++++	name	Max140Text	Debtor's bank name as given in the SWIFT Directory. Filled-in value (constant): KOMERCNI BANKA A.S.
++++++	postalAddress	+	Debtor's bank postal address.
+++++++	streetName	Max70Text	Street name used in the debtor's postal address.
+++++++	buildingNumber	Max16Text	Building number used in the debtor's postal address.
+++++++	postCode	Max16Text	Postal code used in the debtor's postal address.

+++++++	townName	Max35Text	Town name used in the debtor's postal address.
+++++++	country	CountryCode	Country name used in the debtor's postal address.
+++++++	addressLine	Max70Text	Debtor's postal address in an unstructured format.
++++++	other	+	Other identification of the debtor's bank.
+++++++	identification	Max35Text	Other identification of the debtor's bank, in particular in the form of the so-called local bank code.
++++	creditorAgent	+	Information about the creditor's bank. Depending on the direction of the payment, filled in for the counterparty.
+++++	financialinstitutionidentification	+	Creditor's bank code expressed in the BIC / SWIFT international code format. Filled-in value (constant): KOMBCZPPXXX.
++++++	bic	BICIdentifier	Creditor's bank BIC / SWIFT code. Filled-in value (constant): KOMBCZPPXXX.
++++++	clearingSystemMemberIdentification	+	Creditor's bank code in a local format – either as a code or a text description.
+++++++	clearingSystemIdentification	+	Identification of the creditor's bank in the local payment system in which the creditor's bank operates.
++++++++	code	Code	Identification of the creditor's bank in the local payment system in which the creditor's bank operates, in the form of a payment system code.
++++++++	proprietary	Max35Text	Identification of the creditor's bank in the local payment system in which the creditor's bank operates in an unstructured format of a descriptive text.
+++++++	memberIdentification	Max35Text	Creditor's bank code in a local format.
++++++	name	Max140Text	Creditor's bank.
++++++	postalAddress	+	Creditor's bank postal address.

+++++++	streetName	Max70Text	Street name used in the creditor's postal address.
+++++++	buildingNumber	Max16Text	Building number used in the creditor's postal address.
+++++++	postCode	Max16Text	Postal code used in the creditor's postal address.
+++++++	townName	Max35Text	Town name used in the creditor's postal address.
+++++++	country	CountryCode	Country name used in the creditor's postal address.
+++++++	addressLine	Max70Text	Creditor's postal address in an unstructured format.
++++++	other	+	Other identification of the creditor's bank, in particular in the form of the so-called local bank code.
+++++++	identification	Max35Text	Information about the creditor's bank. Depending on the direction of the payment, filled in for the counterparty.
+++	purpose	+	The purpose of the payment.
++++	code	Code	The purpose of the payment expressed in the form of the code, as stated in the payment instruction.
++++	proprietary	Max35Text	The purpose of the payment expressed as unstructured information, as stated in the payment instruction.
+++	remittanceInformation	+	Additional information about the payment.
++++	unstructured	Max140Text	Additional details concerning the payment filled in as unstructured information. If the unstructured record of the additional details is contained more than once in the payment instruction, only the first occurrence is taken into account.
++++	structured	+	Payment symbols are expected here (a variable, specific, and constant symbol). If a variable, specific or constant symbol is not filled-in in the payment instruction, the

			entire Structured Remittance Information structure remains blank.
+++++	creditorReferenceInformation	+	Displaying information about the variable, specific, and constant symbol.
++++++	reference	Max35Text	The field should start with 'VS:' (for the variable symbol), or 'SS:' (for the specific symbol), or 'KS:' (for the constant symbol). One repetition of the 'reference' field may contain all 3 symbols, but each of them can occur there only once. The symbols are filled in as follows - VS:|KS:|SS:)[0-9]{1,10}.
+++	additionalTransactionInformation	Max500Text	Additional information provided by the bank; filled in only if they are contained in the record. Additional information concerning SEPA DD is given here (e.g. Creditor Identifier, Payment scheme, SEPA DD sequence etc.)

Interest – occurrence of elements

Level	Message element	Occurence
+	entryReference	[0..1] [1..1]
+	amount	[1..1]
++	value	[1..1]
++	currency	[1..1]
+	creditDebitIndicator	[1..1]
+	reversalIndicator	[0..1] [0..0]
+	status	[1..1] always "BOOK"
+	bookingDate	[1..1]
++	date	[1..1]
+	valueDate	[1..1]
++	date	[1..1]
+	bankTransactionCode	[1..1]
++	proprietary	[1..1]
+++	code	[1..1]
+++	issuer	[1..1] always "CBA"
+	entryDetails	[0..1] [1..1]

++	transactionDetails	[0..1] [1..1]
+++	references	[0..1] [0..0]
++++	messageIdentification	[0..1] [0..0]
++++	accountServicerReference	[0..1] [0..0]
++++	paymentInformationIdentification	[0..1] [0..0]
++++	instructionIdentification	[0..1] [0..0]
++++	endToEndIdentification	[0..1] [0..0]
++++	chequeNumber	[0..1] [0..0]
++++	clearingSystemReference	[0..1] [0..0]
+++	amountDetails	[0..1] [1..1]
++++	instructedAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
++++	counterValueAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
+++++	currencyExchange	[0..1] [1..1]
++++++	sourceCurrency	[1..1]
++++++	targetCurrency	[0..1]
++++++	exchangeRate	[1..1] [0..1]
++++	proprietaryAmount	[0..0]
+++++	type	[0..0]
+++++	amount	[0..0]
++++++	value	[1..1] [0..0]
++++++	currency	[1..1] [0..0]
+++	relatedParties	[0..1] [0..0]
++++	debtor	[0..1] [0..0]
+++++	name	[0..1] [0..0]
++++	debtorAccount	[0..1] [0..0]
+++++	identification	[1..1] [0..0]
++++++	iban	[1..1] [0..0]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditor	[0..1] [0..0]
+++++	name	[0..1] [0..0]
++++	creditorAccount	[0..1] [0..0]

+++++	identification	[1..1] [0..0]
++++++	iban	[1..1] [0..0]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	relatedAgents	[0..1] [0..0]
++++	debtorAgent	[0..1] [0..0]
+++++	financialInstitutionIdentification	[1..1] [0..0]
++++++	bic	[0..1] [0..0]
++++++	name	[0..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditorAgent	[0..1] [0..0]
+++++	financialinstitutionidentification	[1..1] [0..0]
++++++	bic	[0..1] [0..0]
++++++	name	[0..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	remittanceInformation	[0..1]
++++	unstructured	[0..1]
++++	structured	[0..1]
+++++	creditorReferenceInformation	[0..1]
++++++	reference	[0..1]
+++	additionalTransactionInformation	[0..1]

JSON – example of a response:


```
===============
{
 "pageNumber": 5,
 "pageCount": 7,
 "pageSize": 100,
 "nextPage": 6,
 "transactions": [
      {
      "entryReference": "060-060-004-370459",
      "amount":       {
        "value": 0.59,
        "currency": "EUR"
      },
      "creditDebitIndicator": "DBIT",
      "status": "BOOK",
      "bookingDate": {"date": "2019-01-31"},
      "valueDate": {"date": "2019-01-31"},
      "bankTransactionCode": {"proprietary":       {
        "code": 90000201003,
        "issuer": "CBA"
```

```
    }},
    "entryDetails": {"transactionDetails":      {
      "amountDetails":          {
        "instructedAmount": {"amount":            {
          "value": 0.59,
          "currency": "EUR"
        }},
        "counterValueAmount":           {
          "amount":              {
            "value": 0.59,
            "currency": "EUR"
          },
          "currencyExchange":              {
            "sourceCurrency": "EUR",
            "targetCurrency": "EUR",
            "exchangeRate": "1"
      }}},
      "additionalTransactionInformation": "ODEPSANÝ ÚROK"
  }}}
```


Charge/fee – occurrence of elements

Level	Message element	Occurence
+	entryReference	[0..1] [1..1]
+	amount	[1..1]
++	value	[1..1]
++	currency	[1..1]
+	creditDebitIndicator	[1..1]
+	reversalIndicator	[0..1] [0..0]
+	status	[1..1] always “BOOK”
+	bookingDate	[1..1]
++	date	[1..1]
+	valueDate	[1..1]
++	date	[1..1]
+	bankTransactionCode	[1..1]
++	proprietary	[1..1]
+++	code	[1..1]
+++	issuer	[1..1] always “CBA“
+	entryDetails	[0..1] [1..1]
++	transactionDetails	[0..1] [1..1]
+++	references	[0..1] [0..0]
++++	messageIdentification	[0..1] [0..0]
++++	accountServicerReference	[0..1] [0..0]
++++	paymentInformationIdentification	[0..1] [0..0]
++++	instructionIdentification	[0..1] [0..0]

++++	endToEndIdentification	[0..1] [0..0]
++++	chequeNumber	[0..1] [0..0]
++++	clearingSystemReference	[0..1] [0..0]
+++	amountDetails	[0..1] [1..1]
++++	instructedAmount	[0..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
++++	counterValueAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
+++++	currencyExchange	[0..1] [1..1]
++++++	sourceCurrency	[1..1]
++++++	targetCurrency	[0..1]
++++++	exchangeRate	[1..1] [0..1]
++++	proprietaryAmount	[0..0]
+++++	type	[0..0]
+++++	amount	[0..0]
++++++	value	[1..1] [0..0]
++++++	currency	[1..1] [0..0]
+++	relatedParties	[1..1] [0..0]
++++	debtor	[1..1] [0..0]
+++++	name	[0..1] [0..0]
++++	debtorAccount	[0..1] [0..0]
+++++	identification	[0..1] [0..0]
++++++	iban	[0..1] [0..0]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditor	[1..1] [0..0]
+++++	name	[1..1] [0..0]
++++	creditorAccount	[0..1] [0..0]
+++++	identification	[0..1] [0..0]
++++++	iban	[0..1] [0..0]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	relatedAgents	[1..1] [0..0]
++++	debtorAgent	[1..1] [0..0]

+++++	financialInstitutionIdentification	[0..1] [0..0]
++++++	bic	[0..1] [0..0]
++++++	name	[1..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[0..1] [0..0]
++++	creditorAgent	[0..1] [0..0]
+++++	financialinstitutionidentification	[1..1] [0..0]
++++++	bic	[0..1] [0..0]
++++++	name	[1..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[0..1] [0..0]
+++	remittanceInformation	[0..1]
++++	unstructured	[0..1]
++++	structured	[0..1]
+++++	creditorReferenceInformation	[0..1]
++++++	reference	[0..1]
+++	additionalTransactionInformation	[0..1]

JSON – example of a response:


```
===============
{
  "pageNumber": 0,
  "pageCount": 1,
  "pageSize": 20,
  "transactions":  [
        {
      "entryReference": "001-04032019 1602 602023 745261",
      "amount":       {
        "value": 250,
        "currency": "CZK"
      },
      "creditDebitIndicator": "DBIT",
      "status": "BOOK",
      "bookingDate": {"date": "2019-03-04"},
      "valueDate": {"date": "2019-03-04"},
      "bankTransactionCode": {"proprietary":       {
        "code": 40000201000,
        "issuer": "CBA"
      }},
      "entryDetails": {"transactionDetails":       {
        "amountDetails":         {
          "instructedAmount": {"amount":           {
            "value": 250,
            "currency": "CZK"
          }},
          "counterValueAmount":           {
            "amount":             {
              "value": 250,
```

```
            "currency": "CZK"
          },
          "currencyExchange":              {
            "sourceCurrency": "CZK",
            "targetCurrency": "CZK",
            "exchangeRate": "1"
      }}},
      "remittanceInformation": {
        "structured": {
          "creditorReferenceInformation": {
            "reference":          [
              "VS:0000000009",
              "SS:7831291011",
              "KS:0000000898"
      ]}}},
      "additionalTransactionInformation": "Platba na vrub vašeho účtu,
        POPLATEK ZA ZAHRANIČNÍ ODCHOZÍ ÚHRADU, IU01RFF9MWS 12"
  }}}
```


Domestic payment – occurrence of elements

Level	Message element	Occurence
+	entryReference	[0..1] [1..1]
+	amount	[1..1]
++	value	[1..1]
++	currency	[1..1]
+	creditDebitIndicator	[1..1]
+	reversalIndicator	[0..1] [0..0]
+	status	[1..1] “BOOK” or “PDNG”. Payment card holds can be identified by combination of status PDNG and bankTransaction code 30000301000.
+	bookingDate	[1..1]
++	date	[1..1] [0..1] not mandatory for transaction in PDNG status (e.g. payment card holds)
+	valueDate	[1..1]
++	date	[1..1]
+	bankTransactionCode	[1..1]
++	proprietary	[1..1]
+++	code	[1..1]
+++	issuer	[1..1] always “CBA“
+	entryDetails	[1..1]
++	transactionDetails	[1..1]
+++	references	[0..1] [0..1]
++++	messageIdentification	[0..1] [0..0]
++++	accountServicerReference	[0..1] [0..1]
++++	paymentInformationIdentification	[0..1] [0..0]

++++	instructionIdentification	[0..1] [0..0]
++++	endToEndIdentification	[0..1] [0..0]
++++	chequeNumber	[0..1] [0..0]
++++	clearingSystemReference	[0..1] [0..0]
+++	amountDetails	[0..1] [1..1]
++++	instructedAmount	[0..1] [0..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
++++	counterValueAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
+++++	currencyExchange	[0..1] [1..1]
++++++	sourceCurrency	[1..1]
++++++	targetCurrency	[0..1]
++++++	exchangeRate	[1..1] [0..1]
++++	proprietaryAmount	[0..0]
+++++	type	[0..0]
+++++	amount	[0..0]
++++++	value	[1..1] [0..0]
++++++	currency	[1..1] [0..0]
+++	relatedParties	[1..1]
++++	debtor	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	name	[1..1]
++++	debtorAccount	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	identification	[1..1]
++++++	iban	[1..1]
++++++	other	[1..1]-[0..0]
+++++++	identification	[1..1] [0..0]
++++	creditor	[0..1] depending on the direction of the payment, to be filled-in in case of credit transfers within KB
+++++	name	[0..1]
++++	creditorAccount	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	identification	[1..1]
++++++	iban	[1..1]
++++++	other	[1..1]-[0..0]

+++++++	identification	[1..1] [0..0]
+++	relatedAgents	[1..1]
++++	debtorAgent	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	financialInstitutionIdentification	[1..1]
++++++	bic	[0..1]-[1..1]
++++++	name	[0..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditorAgent	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	financialinstitutionidentification	[1..1]
++++++	bic	[0..1] [1..1]
++++++	name	[0..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	remittanceInformation	[0..1]
++++	unstructured	[0..1]
++++	structured	[0..1]
+++++	creditorReferenceInformation	[0..1]
++++++	reference	[0..1]
+++	additionalTransactionInformation	[0..1]

JSON – example of a response:


```
===============
{
  "pageNumber": 0,
  "pageCount": 3,
  "pageSize": 20,
  "nextPage": 1,
  "transactions":    [
        {
      "entryReference": "357-12032019 1602 602033 935171",
      "amount":        {
        "value": 1.23,
        "currency": "CZK"
      },
      "creditDebitIndicator": "DBIT",
      "status": "BOOK",
      "bookingDate": {"date": "2019-03-12"},
      "valueDate": {"date": "2019-03-12"},
      "bankTransactionCode": {"proprietary":        {
        "code": 10000101008,
        "issuer": "CBA"
      }},
      "entryDetails": {"transactionDetails":        {
```

```
      "references": {"accountServicerReference": "ZU000005UO8"},
      "amountDetails":          {
        "instructedAmount": {"amount":            {
          "value": 1.23,
          "currency": "CZK"
        }},
        "counterValueAmount":            {
          "amount":              {
            "value": 1.23,
            "currency": "CZK"
          },
          "currencyExchange":              {
            "sourceCurrency": "CZK",
            "targetCurrency": "CZK",
            "exchangeRate": "1"
      }}},
      "relatedParties": {
      "creditorAccount": {
        "identification": {
          "iban": "CZ3203000000000001111132"}}},
      "relatedAgents": {
        "creditorAgent": {
          "financialInstitutionIdentification": {
            "bic": "CEKOCZPPXXX"}}},
      "remittanceInformation": {
        "unstructured": "Poznámka pro příjemce"},
      "additionalTransactionInformation": "Platba na vrub vašeho účtu"
  }}}
```


SEPA payment – occurrence of elements

Level	Message element	Occurence
+	entryReference	[0..1] [1..1]
+	amount	[1..1]
++	value	[1..1]
++	currency	[1..1]
+	creditDebitIndicator	[1..1]
+	reversalIndicator	[0..1] [0..0]
+	status	[1..1] always “BOOK”
+	bookingDate	[1..1]
++	date	[1..1]
+	valueDate	[1..1]
++	date	[1..1]
+	bankTransactionCode	[1..1]
++	proprietary	[1..1]
+++	code	[1..1]
+++	issuer	[1..1] always “CBA“
+	entryDetails	[1..1]

++	transactionDetails	[1..1]
+++	references	[0..1]
++++	messageIdentification	[0..1] [0..0]
++++	accountServicerReference	[0..1]
++++	paymentInformationIdentification	[0..1] [0..0]
++++	instructionIdentification	[0..1] [0..0]
++++	endToEndIdentification	[0..1]
++++	mandateIdentification	[0..0]/[0..1]
++++	chequeNumber	[0..1] [0..0]
++++	clearingSystemReference	[0..1] [0..0]
+++	amountDetails	[0..1] [1..1]
++++	instructedAmount	[0..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
++++	counterValueAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
+++++	currencyExchange	[0..1] [1..1]
++++++	sourceCurrency	[1..1]
++++++	targetCurrency	[0..1]
++++++	exchangeRate	[1..1] [0..1]
++++	proprietaryAmount	[0..0]
+++++	type	[0..0]
+++++	amount	[0..0]
++++++	value	[1..1] [0..0]
++++++	currency	[1..1] [0..0]
+++	charges	[0..1]
++++++	bearer	[0..1]
+++	relatedParties	[1..1]
++++	debtor	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	name	[1..1]
+++++	postalAddress	[0..1] [0..0]
++++++	streetName	[0..1] [0..0]
++++++	buildingNumber	[0..1] [0..0]
++++++	postCode	[0..1] [0..0]
++++++	townName	[0..1] [0..0]

++++++	country	[0..1] [0..0]
++++++	addressLine	[0..7] [0..0]
+++++	identification	[0..1] [0..0]
++++++	organisationIdentification	[1..1] [0..0]
+++++++	bicOrBei	[0..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemeName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++++	privateIdentification	[1..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemaName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++	debtorAccount	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	identification	[1..1]
++++++	iban	[1..1]
+++++	name	[0..1] [0..0]
++++	ultimateDebtor	[0..1] [0..0]
+++++	name	[0..1] [0..0]
+++++	postalAddress	[0..1] [0..0]
++++++	streetName	[0..1] [0..0]
++++++	buildingNumber	[0..1] [0..0]
++++++	postCode	[0..1] [0..0]
++++++	townName	[0..1] [0..0]
++++++	country	[0..1] [0..0]
++++++	addressLine	[0..7] [0..0]
+++++	identification	[0..1] [0..0]
++++++	organisationIdentification	[1..1] [0..0]
+++++++	bicOrBei	[0..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemeName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]

+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++++	privateIdentification	[1..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemaName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++	creditor	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	name	[1..1]
+++++	postalAddress	[0..1] [0..0]
++++++	streetName	[0..1] [0..0]
++++++	buildingNumber	[0..1] [0..0]
++++++	postCode	[0..1] [0..0]
++++++	townName	[0..1] [0..0]
++++++	country	[0..1] [0..0]
++++++	addressLine	[0..7] [0..0]
+++++	identification	[0..1] [0..0]
++++++	organisationIdentification	[1..1] [0..0]
+++++++	bicOrBei	[0..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemeName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++++	privateIdentification	[1..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemaName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++	creditorAccount	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	identification	[1..1]
++++++	iban	[1..1]

+++++	name	[0..1] [0..0]
++++	ultimateCreditor	[0..1] [0..0]
+++++	name	[0..1] [0..0]
+++++	postalAddress	[0..1] [0..0]
++++++	streetName	[0..1] [0..0]
++++++	buildingNumber	[0..1] [0..0]
++++++	postCode	[0..1] [0..0]
++++++	townName	[0..1] [0..0]
++++++	country	[0..1] [0..0]
++++++	addressLine	[0..7] [0..0]
+++++	identification	[0..1] [0..0]
++++++	organisationIdentification	[1..1] [0..0]
+++++++	bicOrBei	[0..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemeName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
++++++	privateIdentification	[1..1] [0..0]
+++++++	other	[0..n] [0..0]
++++++++	identification	[1..1] [0..0]
++++++++	schemaName	[0..1] [0..0]
+++++++++	code	[1..1] [0..0]
+++++++++	proprietary	[1..1] [0..0]
++++++++	issuer	[0..1] [0..0]
+++	relatedAgents	[1..1]
++++	debtorAgent	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	financialInstitutionIdentification	[1..1]
++++++	bic	[1..1]
++++++	name	[0..1] [0..0]
++++	creditorAgent	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	financialinstitutionidentification	[1..1]
++++++	bic	[1..1]
++++++	name	[0..1] [0..0]
+++	purpose	[0..1] [0..0]
++++	code	[1..1] [0..0]

++++	proprietary	[1..1] [0..0]
+++	remittanceInformation	[0..1]
++++	unstructured	[0..1]
++++	structured	[0..1]
+++++	creditorReferenceInformation	[0..1]
++++++	reference	[0..1]
+++	additionalTransactionInformation	[0..1]

JSON – example of a response:


```
===============
{
 "pageNumber": 10,
 "pageCount": 8,
 "pageSize": 20,
 "nextPage": 2,
 "transactions": [
        {
      "entryReference": "001-19022019 1602 602000 210641",
      "amount":        {
        "value": 88.01,
        "currency": "CZK"
      },
      "creditDebitIndicator": "DBIT",
      "status": "BOOK",
      "bookingDate": {"date": "2019-02-19"},
      "valueDate": {"date": "2019-02-19"},
      "bankTransactionCode": {"proprietary":        {
        "code": 10000401001,
        "issuer": "CBA"
      }},
      "entryDetails": {"transactionDetails":        {
        "references":          {
          "accountServicerReference": "IU01RFEL7A9",
          "endToEndIdentification": "SEPA E2E8888"
        },
        "amountDetails":          {
          "instructedAmount": {"amount":            {
            "value": 3.33,
            "currency": "EUR"
          }},
          "counterValueAmount":            {
            "amount":              {
              "value": 88.01,
              "currency": "CZK"
            },
            "currencyExchange":              {
              "sourceCurrency": "CZK",
              "targetCurrency": "EUR",
              "exchangeRate": 26.4292
            }
          }
        },
        "relatedParties":          {
```

```
        "creditor": {"name": "Lenina z Tatrabank SK"},
        "creditorAccount": {
          "identification": {
            "iban": "SK9711000000002621370505"}}
      },
      "relatedAgents": {
        "creditorAgent": {
          "financialInstitutionIdentification": {
            "bic": "TATRSKBXXXX"}}},
      "remittanceInformation":          {
        "unstructured": "ZPL SEPA XXXX  /VS/999999999/KS/3333/SS/111111",
        "structured": {
          "creditorReferenceInformation": {
            "reference":            [
              "VS:0999999999",
              "SS:7831148411",
              "KS:6020000000"
        ]}}},
      "additionalTransactionInformation": "Platba na vrub vašeho účtu, ucetSK9711000000002621370505 rfKB
7831148411602, bankTATRSKBXXXX, IU01RFEL7A9 11"
    }}}
```


Foreign payment – occurrence of elements

Level	Message element	Occurence
+	entryReference	[0..1] [1..1]
+	amount	[1..1]
++	value	[1..1]
++	currency	[1..1]
+	creditDebitIndicator	[1..1]
+	reversalIndicator	[0..1] [0..0]
+	status	[1..1] always “BOOK”
+	bookingDate	[1..1]
++	date	[1..1]
+	valueDate	[1..1]
++	date	[1..1]
+	bankTransactionCode	[1..1]
++	proprietary	[1..1]
+++	code	[1..1]
+++	issuer	[1..1] always “CBA“
+	entryDetails	[1..1]
++	transactionDetails	[1..1]
+++	references	[0..1]
++++	messageIdentification	[0..1] [0..0]
++++	accountServicerReference	[0..1]
++++	paymentInformationIdentification	[0..1] [0..0]

++++	instructionIdentification	[0..1] [0..0]
++++	endToEndIdentification	[0..1] [0..0]
++++	chequeNumber	[0..1] [0..0]
++++	clearingSystemReference	[0..1] [0..0]
+++	amountDetails	[0..1] [1..1]
++++	instructedAmount	[0..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
++++	counterValueAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
+++++	currencyExchange	[0..1] [1..1]
++++++	sourceCurrency	[1..1]
++++++	targetCurrency	[0..1]
++++++	exchangeRate	[1..1] [0..1]
++++	proprietaryAmount	[0..0]
+++++	type	[0..0]
+++++	amount	[0..0]
++++++	value	[1..1] [0..0]
++++++	currency	[1..1] [0..0]
+++	charges	[0..1]
++++++	bearer	[0..1]
+++	relatedParties	[1..1]
++++	debtor	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	name	[1..1]
+++++	postalAddress	[0..1] [0..0]
++++++	streetName	[0..1] [0..0]
++++++	buildingNumber	[0..1] [0..0]
++++++	postCode	[0..1] [0..0]
++++++	townName	[0..1] [0..0]
++++++	country	[0..1] [0..0]
++++++	addressLine	[0..7] [0..0]
++++++++	issuer	[0..1] [0..0]
++++	debtorAccount	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	identification	[1..1]

++++++	iban	[1..1]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++++	name	[0..1] [0..0]
++++	creditor	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	name	[1..1]
+++++	postalAddress	[0..1] [0..0]
++++++	streetName	[0..1] [0..0]
++++++	buildingNumber	[0..1] [0..0]
++++++	postCode	[0..1] [0..0]
++++++	townName	[0..1] [0..0]
++++++	country	[0..1] [0..0]
++++++	addressLine	[0..7] [0..0]
++++	creditorAccount	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	identification	[1..1]
++++++	iban	[1..1]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++++	name	[0..1] [0..0]
+++	relatedAgents	[1..1]
++++	debtorAgent	[1..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	financialInstitutionIdentification	[1..1]
++++++	bic	[0..1] [1..1]
++++++	clearingSystemMemberIdentification	[0..1] [0..0]
+++++++	clearingSystemIdentification	[0..1] [0..0]
++++++++	code	[1..1] [0..0]
++++++++	proprietary	[1..1] [0..0]
+++++++	memberIdentification	[1..1] [0..0]
++++++	name	[0..1] [0..0]
++++++	postalAddress	[0..1] [0..0]
+++++++	streetName	[0..1] [0..0]
+++++++	buildingNumber	[0..1] [0..0]
+++++++	postCode	[0..1] [0..0]
+++++++	townName	[0..1] [0..0]
+++++++	country	[0..1] [0..0]
+++++++	addressLine	[0..7] [0..0]

++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditorAgent	[1..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	financialinstitutionidentification	[1..1]
++++++	bic	[0..1] [1..1]
++++++	clearingSystemMemberIdentification	[0..1] [0..0]
+++++++	clearingSystemIdentification	[0..1] [0..0]
++++++++	code	[1..1] [0..0]
++++++++	proprietary	[1..1] [0..0]
+++++++	memberIdentification	[1..1] [0..0]
++++++	name	[0..1] [0..0]
++++++	postalAddress	[0..1] [0..0]
+++++++	streetName	[0..1] [0..0]
+++++++	buildingNumber	[0..1] [0..0]
+++++++	postCode	[0..1] [0..0]
+++++++	townName	[0..1] [0..0]
+++++++	country	[0..1] [0..0]
+++++++	addressLine	[0..7] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	remittanceInformation	[0..1]
++++	unstructured	[0..1]
++++	structured	[0..1]
+++++	creditorReferenceInformation	[0..1]
++++++	reference	[0..1]
+++	additionalTransactionInformation	[0..1]

JSON – example of a response:


```
===============
{
  "pageNumber": 5,
  "pageCount": 7,
  "pageSize": 100,
  "nextPage": 6,
  "transactions": [
    {
      "entryReference": "357-28022019 1586 586004 320041",
      "amount":        {
        "value": 9.81,
        "currency": "USD"
      },
      "creditDebitIndicator": "DBIT",
```

```
    "status": "BOOK",
    "bookingDate": {"date": "2019-02-28"},
    "valueDate": {"date": "2019-02-28"},
    "bankTransactionCode": {"proprietary":        {
      "code": 10000201006,
      "issuer": "CBA"
    }},
    "entryDetails": {"transactionDetails":        {
      "references": {"accountServicerReference": "ZU000005IPQ"},
      "amountDetails":          {
        "instructedAmount": {"amount":             {
          "value": 13.13,
          "currency": "AUD"
        }},
        "counterValueAmount":             {
          "amount":               {
            "value": 9.81,
            "currency": "USD"
          },
          "currencyExchange":               {
            "sourceCurrency": "AUD",
            "targetCurrency": "USD",
            "exchangeRate": 0.7471
      }}},
      "charges": {"bearer": "OUR"},
      "relatedParties":          {
        "creditor": {"name": "Sultan Sulejman"},
        "creditorAccount": {
           "identification": {
           "iban": "TR560006701000000081658540"}
    }},
      "relatedAgents": {
      "creditorAgent": {
        "financialInstitutionIdentification": {
          "bic": "YAPITRISFEX"}
    }},
      "remittanceInformation":          {
        "unstructured": "ZPL mimo EHP,/KS/0308,/VS/123456789,AUD platba z USD",
        "structured": {
          "creditorReferenceInformation": {
             "reference":            [
              "VS:0123456789",
              "SS:7831259721",
              "KS:5860000308"]
      }}},
      "additionalTransactionInformation": "Odchozí platba, ZU000005IPQ"
  }}}
```


Cash – occurrence of elements

Level	Message element	Occurence
+	entryReference	[0..1] [1..1]
+	amount	[1..1]
++	value	[1..1]
++	currency	[1..1]

+	creditDebitIndicator	[1..1]
+	reversalIndicator	[0..1] [0..0]
+	status	[1..1] always “BOOK”
+	bookingDate	[1..1]
++	date	[1..1]
+	valueDate	[1..1]
++	date	[1..1]
+	bankTransactionCode	[1..1]
++	proprietary	[1..1]
+++	code	[1..1]
+++	issuer	[1..1] always “CBA“
+	entryDetails	[0..1] [1..1]
++	transactionDetails	[0..1] [1..1]
+++	references	[0..1] [0..0]
++++	messageIdentification	[0..1] [0..0]
++++	accountServicerReference	[0..1] [0..0]
++++	paymentInformationIdentification	[0..1] [0..0]
++++	instructionIdentification	[0..1] [0..0]
++++	endToEndIdentification	[0..1] [0..0]
++++	chequeNumber	[0..1] [0..0]
++++	clearingSystemReference	[0..1] [0..0]
+++	amountDetails	[0..1] [1..1]
++++	instructedAmount	[0..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
++++	counterValueAmount	[0..1] [1..1]
+++++	amount	[1..1]
++++++	value	[1..1]
++++++	currency	[1..1]
+++++	currencyExchange	[0..1] [1..1]
++++++	sourceCurrency	[1..1] [1..1]
++++++	targetCurrency	[0..1]
++++++	exchangeRate	[1..1] [0..1]
++++	proprietaryAmount	[0..0]
+++++	type	[0..0]
+++++	amount	[0..0]
++++++	value	[1..1] [0..0]

++++++	currency	[1..1] [0..0]
+++	relatedParties	[0..1] [0..0]
++++	debtor	[0..1] [0..0]
+++++	name	[0..1] [0..0]
++++	debtorAccount	[0..1] [0..0]
+++++	identification	[1..1] [0..0]
++++++	iban	[1..1] [0..0]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditor	[0..1] [0..0]
+++++	name	[0..1] [0..0]
++++	creditorAccount	[0..1] [0..0]
+++++	identification	[1..1] [0..0]
++++++	iban	[1..1] [0..0]
++++++	other	[1..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	relatedAgents	[0..1]
++++	debtorAgent	[0..1] depending on the direction of the payment, to be filled-in in case of debit transfers
+++++	financialInstitutionIdentification	[1..1]
++++++	bic	[0..1] [1..1]
++++++	name	[0..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
++++	creditorAgent	[0..1] depending on the direction of the payment, to be filled-in in case of credit transfers
+++++	financialinstitutionidentification	[1..1]
++++++	bic	[0..1] [1..1]
++++++	name	[0..1] [0..0]
++++++	other	[0..1] [0..0]
+++++++	identification	[1..1] [0..0]
+++	remittanceInformation	[0..1]
++++	unstructured	[0..1]
++++	structured	[0..1]
+++++	creditorReferenceInformation	[0..1]
++++++	reference	[0..1]
+++	additionalTransactionInformation	[0..1]

JSON – example of a response:

```
===============
{
  "pageNumber": 0,
  "pageCount": 1,
  "pageSize": 20,
  "transactions": [  {
    "entryReference": "301-12022019 1031 700001 138752",
    "amount":     {
      "value": 37.65,
      "currency": "EUR"
    },
    "creditDebitIndicator": "CRDT",
    "status": "BOOK",
    "bookingDate": {"date": "2019-02-12"},
    "valueDate": {"date": "2019-02-12"},
    "bankTransactionCode": {"proprietary":     {
      "code": 20000100000,
      "issuer": "CBA"
    }},
    "entryDetails": {"transactionDetails":     {
      "amountDetails":       {
        "instructedAmount": {"amount":         {
          "value": 1000,
          "currency": "CZK"
        }},
        "counterValueAmount":         {
          "amount":           {
            "value": 37.65,
            "currency": "EUR"
          },
          "currencyExchange":           {
            "sourceCurrency": "EUR",
            "targetCurrency": "CZK",
            "exchangeRate": 26.5577
      }}},
      "relatedAgents": {
        "debtorAgent": {
          "financialInstitutionIdentification": {
            "bic": "KOMBCZPPXXX"}}},
      "remittanceInformation":       {
        "unstructured": "abc def",
        "structured": {
          "creditorReferenceInformation": {
            "reference":         [
              "VS:0000000009",
              "SS:0123456789",
              "KS:0000000379"
        ]}}},
      "additionalTransactionInformation": "VKLAD HOTOVOSTI"
}}}]}
```


Payment card transaction – occurrence of elements

Level	Message element	Occurence	Latest changes
+	entryReference	[0..1] [1..1]	

+	amount	[1..1]	
++	value	[1..1]	
++	currency	[1..1]	
+	creditDebitIndicator	[1..1]	
+	reversalIndicator	[0..1] [0..0]	
+	status	[1..1] always “BOOK”	
+	bookingDate	[1..1]	
++	date	[1..1]	
+	valueDate	[1..1]	
++	date	[1..1]	
+	bankTransactionCode	[1..1]	
++	proprietary	[1..1]	
+++	code	[1..1]	
+++	issuer	[1..1] always “CBA“	
+	entryDetails	[0..1] [1..1]	
++	transactionDetails	[0..1] [1..1]	
+++	references	[0..1]	[0..0]
++++	messageIdentification	[0..1] [0..0]	
++++	accountServicerReference	[0..1]	[0..0]
++++	paymentInformationIdentification	[0..1] [0..0]	
++++	instructionIdentification	[0..1] [0..0]	
++++	endToEndIdentification	[0..1] [0..0]	
++++	chequeNumber	[0..1] [0..0]	
++++	clearingSystemReference	[0..1] [0..0]	
+++	amountDetails	[0..1] [1..1]	
++++	instructedAmount	[0..1]	
+++++	amount	[1..1]	
++++++	value	[1..1]	
++++++	currency	[1..1]	
++++	counterValueAmount	[0..1] [1..1]	
+++++	amount	[1..1]	
++++++	value	[1..1]	
++++++	currency	[1..1]	
+++++	currencyExchange	[0..1] [1..1]	[0..0]
++++++	sourceCurrency	[1..1]	[0..0]
++++++	targetCurrency	[0..1]	[0..0]
++++++	exchangeRate	[1..1] [0..1]	[0..0]
++++	proprietaryAmount	[0..0]	

+++++	type	[0..0]	
+++++	amount	[0..0]	
++++++	value	[1..1] [0..0]	
++++++	currency	[1..1] [0..0]	
+++	relatedParties	[0..1]	
++++	debtor	[0..1] depending on the direction of the payment, to be filled-in in case of debit transfers	
+++++	name	[0..1] [1..1]	
++++	debtorAccount	[0..1]	[0..0]
+++++	identification	[1..1]	[0..0]
++++++	iban	[1..1]	[0..0]
++++++	other	[1..1] [0..0]	
+++++++	identification	[1..1] [0..0]	
++++	creditor	[0..1] depending on the direction of the payment, to be filled-in in case of credit transfers	
+++++	name	[0..1] [1..1]	
++++	creditorAccount	[0..1]	[0..0]
+++++	identification	[1..1]	[0..0]
++++++	iban	[1..1]	[0..0]
++++++	other	[1..1] [0..0]	
+++++++	identification	[1..1] [0..0]	
+++	relatedAgents	[0..1]	[0..0]
++++	debtorAgent	[0..1]	[0..0]
+++++	financialInstitutionIdentification	[1..1]	[0..0]
++++++	bic	[0..1]	[0..0]
++++++	name	[0..1] [0..0]	
++++++	other	[0..1] [0..0]	
+++++++	identification	[1..1] [0..0]	
++++	creditorAgent	[0..1]	[0..0]
+++++	financialinstitutionidentification	[1..1]	[0..0]
++++++	bic	[0..1]	[0..0]
++++++	name	[0..1] [0..0]	
++++++	other	[0..1] [0..0]	
+++++++	identification	[1..1] [0..0]	
+++	remittanceInformation	[0..1]	
++++	unstructured	[0..1]	

++++	structured	[0..1]	
+++++	creditorReferenceInformation	[0..1]	
++++++	reference	[0..1]	
+++	additionalTransactionInformation	[0..1]	

JSON – example of a response:


```
==============
{
  "pageNumber": 0,
  "pageCount": 1,
  "pageSize": 20,
  "transactions": [  {
    "entryReference": "355-25012019 1086 602013 225091",
    "amount":     {
      "value": 8.57,
      "currency": "USD"
    },
    "creditDebitIndicator": "CRDT",
    "status": "BOOK",
    "bookingDate": {"date": "2019-01-25"},
    "valueDate": {"date": "2019-01-25"},
    "bankTransactionCode": {"proprietary":     {
      "code": 30000103000,
      "issuer": "CBA"
    }},
    "entryDetails": {"transactionDetails":     {
      "references": {"accountServicerReference": "WU010003JIF"},
      "amountDetails":       {
        "instructedAmount": {"amount":         {
          "value": 200,
          "currency": "CZK"
        }},
        "counterValueAmount":         {
          "amount":           {
            "value": 8.57,
            "currency": "USD"
          },
          "currencyExchange":           {
            "sourceCurrency": "CZK",
            "targetCurrency": "USD",
            "exchangeRate": "23.3372"
      }}},
      "relatedParties":       {
        "debtor": {"name": "ATM KB Na poříčí 712/3                "},
        "debtorAccount": {
          "identification": {
            "iban": "CZ3201000000436965780207"}}
      },
      "relatedAgents": {
        "debtorAgent": {
          "financialInstitutionIdentification": {
            "bic": "KOMBCZPPXXX"}}},
      "remittanceInformation":       {
        "unstructured": "ATM TEST PRAHA 1 83 4459 66** **** 4661 VISA
          24.01.2019  200,00 CZK",
```

```
        "structured": {
        "creditorReferenceInformation": {
              "reference":          [
                "VS:0501713027",
                "SS:0202050000"
      ]}}},
      "additionalTransactionInformation":"Příchozí platba,
        201901259643705750D06164104008, VKLAD NA ATM, WU010003JIF 62"
}}}}}
```

5. List of Client's Standing Orders

Paged list of client's standing orders All active standing orders are provided within this resource. These are orders entered by the client as well as orders submitted by the bank (for example, repayments of loans, mortgages, etc.).

URI: /serverapi/aisp/v1/my/standingorders{?size,page,sort,order})
HTTP Method: GET
Request URL: https://api.kb.cz/serverapi/aisp/v2/my/standingorders
Authorization: the request requires the authorization of user/client as part of API call
Certification: the request requires the use of the third party qualified certificate as part of establishing two-way TSL communication with the server. The third party is identified by verifying the validity and content of this certificate.
Pagination: yes
Sorting: no
Filtering: no
Supported encoding: `charset=UTF-8`

Query parameters of the request

Parameter	Type	Manda-tory	Purpose
size	Number	No	Paging. Number of entries per page
page	Number	No	Paging. Required page. + Default: 0
sort	Text	No	A list of fields separated by comma for sorting, arranged according to the meaning
order	Text	No	A list of arrangement methods (ASC, DESC) separated by comma. The order corresponds to the order of fields in the sort parameter.

Example of the API call curl:

```
curl -X GET --header 'Accept: application/json' --header 'x-request-id: 12345' --header 'Authorization: Bearer INPUT_ACCESS_TOKEN_HERE'
'https://api.kb.cz/serverapi/aisp/v1/my/standingorders?size=2&page=1&sort=iban&~~order=ASC'~~
```

Request header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
API-key	Text	Ne	An optional string issued to a communicating third party as the call identifier of that party primarily serving as the configuration element of communication. Not supported.
Authorization	Text	Yes	A parameter used for forwarding the authenticated user's access token along with its type.
TPP-Name	Text	Yes	The name of the original TPP that created the request. Eg. 'Star corporation, a.s.'
TPP-Identification	Text	No	The identification (licence number) of the original TPP that created the request. Eg. 'CZ013574-15'
x-request-id	Text	No	A unique identification of the caller's each particular query. The value of this parameter should therefore be generated randomly, and the individual x-request-ids of the same caller within a short time interval should not be identical. This parameter service returns responses to the

			calling system within response headers.

Example of request headers:

```
{
  "Accept": "application/json",
  "x-request-id": "12345",
}
```

Response header parameters

PARAMETER	*TYPE*	*MANDA-TORY*	*PURPOSE*
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the **application/json** format is primarily supported.
x- request -id	Text	No	Returns the original request id of the API call.

Example of response headers:

```
{
Date: Mon, 18 Mar 2019 10:28:41 GMT
x-request-id: 17549
x-response-id: urn:uuid:df1cb535-6332-4184-8346-e0742c43706e
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET
Access-Control-Allow-Headers: authorization,Access-Control-Allow-Origin,Content-Type,SOAPAction
Content-Language: cs
Content-Type: application/json; charset=UTF-8
Keep-Alive: timeout=60, max=10000
Connection: Keep-Alive
Transfer-Encoding: chunked
Strict-Transport-Security: max-age=16070400; includeSubDomains }
```

Error codes defined for the GET service – Standing orders

HTTP status code	Error code	Purpose
401	UNAUTHORISED	Invalid/missing access token = the user has not been authenticated
401	UNAUTHORISED	Invalid/missing certificate = the provider has not been authenticated
403	FORBIDDEN	Authentication with an invalid certificate or expired access token, a call that does not match a third-party license.
404	PAGE_NOT_FOUND	The query concerns a non-existent page
400	PARAMETER_INVALID	The parameter value is not valid

List of Client's Standing Orders response elements:

Level	Message element	Occurence	Format type	Presentation
+	standingOrders	[1..1]	±	Standing orders collection
++	standingOrderIdentification	[1..1]	Max35Text	Standing order identification
+++	instructionIdentification	[0..1]	Max35Text	Instruction identification
+++	transactionIdentification	[1..1]	Max35Text	Transaction identification
++	amount	[1..1]	STDO - AmountType3CZ ESST - AmountType3CZ XBST - AmountType3Choice	Payment amount
+++	instructedAmount	[1..1]	CurrencyAndAmount	Amount and currency in the instruction
++++	value	[1..1]	Amount	Amount of the transfer
++++	currency	[1..1]	CurrencyCode	Transfer Currency
++	standingOrder	[1..1]	Structure	Structure describing the parameters of the standing orders.
+++	alias	[0..1]	Max250Text	Description resp. the user-defined standing order payment name, taken from payment description (remittanceInformation.unstructured). If the payment description is not filled by user then alias is not included in the response.
+++	execution	[1..1]	±	Features of execution a standing order for payment.
++++	mode	[1..1]	Max35Text	The execution mode defines when or how standing order will be cancelled, processed the last time. Possible values: UNTIL_DATE (standing order is valid until specific date - field lastExecutionDate), UNTIL_CANCELLATION (standing order is valid forever and must be cancelled by client), AFTER_MAX_ITERATION_EXCEEDED (certain count of executions is specified - field maxIterations). Value

				MAX_AMOUNT_EXCEEDED is not supported.
++++	modeDue	[1..1]	Max35Text	The execution due mode defines how the date when order should be executed is specified. Possible values: DUE_DAY_OF_MONTH (specific number of day in the month is defined). Other values are not supported.
++++	interval	[1..1]	Max10Text	Execution interval defines how often order is executed. Possible values: DAILY, WEEKLY, BI-WEEKLY (twice in a month), MONTHLY, BI_MONTHLY, QUARTERLY, HALFYEARLY, YEARLY, IRREGULAR.
++++	intervalDue	[0..0]	Number	Not supported
++	debtorAccount	[1..1]	±	Payer's account
+++	id	[1..1]	Text	API Payment account identifier
+++	identification	[1..1]	±	Payer account identifier
++++	iban	[1..1]	IBAN2007Identifier	IBAN
+++	currency	[1..1]	CurrencyCode, ISO 4217	Payer account currency
++	creditorAccount	[1..1]	±	Payee's account
+++	identification	[1..1]	±	Payee account identifier
++++	iban	[1..1]	IBAN2007Identifier	IBAN
+++	currency	[1..1]	CurrencyCode, ISO 4217	Payee account currency
++	remittanceInformation	[1..1]	±	
+++	unstructured	[0..1]	Max140Text, CERTIS supported	Structured message for the creditor. Also this field is used as Alias if filled by user.
+++	structured	[1..1]	StructuredRemittanceInformation7CZ	Structured message for the creditor – variable symbol
++++	creditorReferenceInformation	[1..1]	CreditorReferenceInformation2CZ	Creditor reference information
+++++	reference	[1..1]	CreditorReferenceInformation2CZ	KS value The variable symbol value is recorded as VS:max.10 digits (e.g. VS:3451859072). If not filled by user value „0“ is provided.

JSON example of response:


```
{
  "pageNumber": 0,
  "pageCount": 2,
  "pageSize": 10,
  "standingOrders":    [
        {
      "standingOrderIdentification": {"transactionIdentification": "221010"},
      "amount": {"instructedAmount":        {
        "value": 999,
        "currency": "CZK"
      }},
      "standingOrder":        {
        "alias": "ČERPÁNÍ_TEST",
        "execution":          {
          "mode": "UNTIL_CANCELLATION",
          "modeDue": "DUE_DAY_OF_MONTH",
          "interval": "MONTHLY"
        }
      },
      "debtorAccount":        {
        "id":
"Gsuil3A_Vcg9Ww2IP5zLWUjK7AmAos8b0xU09CfN9iB6s9q4C3sGrTKNGBQaH_c6QQAPDapJhcAjX6Wd5q3b1i8EK2k4H
6r5Lq6BlnaZWmY",
        "identification": {"iban": "CZ1101000900930763990217"},
        "currency": "CZK"
      },
      "creditorAccount":        {
        "identification": {"iban": "CZ8401000000000009121507"},
        "currency": "CZK"
      },
      "remittanceInformation":        {
        "unstructured": "ČERPÁNÍ_TEST",
        "structured": {"creditorReferenceInformation": {"reference": ["VS:9999"]}}
      }
    },
        {
      "standingOrderIdentification": {"transactionIdentification": "545617"},
      "amount": {"instructedAmount":        {
        "value": 1.69,
        "currency": "CZK"
      }},
      "standingOrder": {"execution":        {
        "mode": "UNTIL_CANCELLATION",
        "modeDue": "DUE_DAY_OF_MONTH",
        "interval": "MONTHLY"
      }},
      "debtorAccount":        {
        "id": "Gsuil3A_Vcg9Ww2IP5zLWUjK7AmAos8b0xU09CfN9iCwyvRZCOZAMlsZnh3Q3famlXQNEE3KJi6kA3P-2gvU-
eK5rmB3I-bOhtrf2JPqkdQ",
        "identification": {"iban": "CZ1101000900930763990217"},
        "currency": "CZK"
      },
      "creditorAccount":        {
        "identification": {"iban": "CZ2101000900930778340217"},
        "currency": "CZK"
      },
      "remittanceInformation": {"structured": {"creditorReferenceInformation": {"reference": ["VS:0"]}}}
```

```
    },
```

6. Standing order details

Resource for get detail about the standing order.

It is at the choice of each bank whether to return details of any client's standing order or just those that the client has established through a particular TPP.

Resource characteristics

URI:	/standingorders/{transactionIdentification}
HTTP Method:	GET
Request URL:	https://api.kb.cz/serverapi/aisp/v2/my/standingorders/{transactionIdentification}
Authorization:	the request requires an authorisation by the user/client as part of the API call.
Certification:	the request requires the use of the third party qualified certificate as part of establishing two-way TSL communication with the server. The third party is identified by verifying the validity and content of this certificate.
Pagination:	no
Sorting:	no
Filtration:	no

Query parameters of the request

Parameter	Type	Manda-tory	Purpose
transactionIdentification	Max35Text	Yes	Identifier of established standing order instruction

Request header parameters

Parameter	Type	Manda-tory	Purpose
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the application/json format is primarily supported.
API-key	Text	No	An optional string issued to a communicating third party as the call identifier of that party primarily serving as the configuration element of communication.
Authorisation	Text	Yes	A parameter used for forwarding the authenticated user's access token along with its type.
TPP-Name	Text	Yes	The name of the original TPP that created the request. Eg. 'Star corporation, a.s.'. In this field, only characters with no diacritics are supported.
TPP-Identification	Text	No	The identification (licence number) of the original TPP that created the request. Eg. 'CZ013574-15'

Response header parameters

Parameter	*Type*	*Manda-tory*	*Purpose*
Content-Type	Text	Yes	A specification of the required transfer format. Based on the prerequisites of the technical specification of this API standard, in this case the application/json format is primarily supported.

Error codes defined for the service GET - Standing order details:

HTTP status code	Error code	Purpose
401	UNAUTHORISED	Invalid/missing access token = user is not authenticated
401	UNAUTHORISED	Invalid/missing certificate = provider is not authenticated
501	NOT_IMPLEMENTED	Method not implemented
404	TRANSACTION_MISSING	The requested transaction ID does not exist.

Standing order details report elements

Level	Message element	Occurrence	Payment type	Format type	Presentation
+	standingOrderIdentification	[1..1]	ALL	Max35Text	Standing order identification
++	instructionIdentification	[1..1]	ALL	Max35Text	Instruction identification
++	endToEndIdentification	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	Max35Text	End to end identification SEPA – Not supported
++	transactionIdentification	[0..0]	ALL	Max35Text	Transaction identification
+	paymentTypeInformation	[1..1]	ALL	The same object as to initiate payments chapter …	
+	amount	[1..1]	ALL	STDO - AmountType3CZ ESST - AmountType3CZ XBST - AmountType3Choice	Payment amount
++	instructedAmount	[1..1]	ALL	CurrencyAndAmount	Amount and currency in the instruction
+++	value	[1..1]	ALL	Amount	Amount of the transfer
+++	currency	[1..1]	ALL	CurrencyCode	Transfer Currency
+	requestedExecutionDate	[1..1]	ALL	ISODate	Requested date of payment
+	standingOrder	[1..1]	ALL	Structure	Structure describing the parameters of the standing orders.
++	alias	[0..1]	ALL	Max250Text	Description resp. the user-defined standing

					order payment name. In the alias field always display information from the remittanceinfromation field.
++	execution	[1..1]	ALL		Features of execution a standing order for payment.
+++	mode	[1..1]	ALL	Max35Text	The execution mode defines when or how standing order will be cancelled, processed the last time. Possible values: UNTIL_DATE (standing order is valid until specific date - field lastExecutionDate), UNTIL_CANCELLATION (standing order is valid forever and must be cancelled by client), AFTER_MAX_ITERATION_EXCEEDED (certain count of executions is specified - field maxIterations) In KB the date of the last payment is determined by the number of repetitions and added or MAX_AMOUNT_EXCEEDED (maximum amount which can be transferred for this order is specified, if next iteration would exceed this amount it is not executed - field maxAmount). Not supported
+++	modeDue	[1..1]	ALL	Max35Text	The execution due mode defines how the date when order should be executed is specified. Possible values: DUE_DAY_OF_MONTH (specific number of day in the month is defined) or DUE_OR_BEFORE_DAY_OF_MONTH (is defined as a specific day of the month or

					the previous day if the day falls on a non-banking day) or DUE_OR_NEXT_DAY_OF_MONTH (is defined as a specific day of the month or the next day if the day falls on a non-banking day) or DUE_LAST_DAY_OF_MONTH (order is executed on last day of particular month). Not supported
+++	interval	[1..1]	ALL	Max10Text	Execution interval defines how often order is executed. Possible values: DAILY, WEEKLY, BI-WEEKLY, MONTHLY, BI_MONTHLY, QUARTERLY, HALFYEARLY, YEARLY, ~~IRREGULAR~~ – Not supported.
+++	intervalDue	[0..0]	ALL	Number	Value represents order number of the day within particular period when the standing order will be requlary executed. Possible values: 1-7 (for WEEKLY interval), 1-28 for MONTHLY, 1-2 for BI_MONTHLY, 1-3 for QUARTERLY, 1-6 for HALFYEARLY, 1-12 for YEARLY. Not supported.
++	validity	[1..1]	ALL		Elements restricting the validity of the standing order. If the final due date is not filled in, an empty field is always returned.
+++	lastExecutionDate	[0..1]	ALL	ISODate	Date when the last order will be processed. Null value responds to ending by the user.
+++	maxIterations	[0..0]	ALL	Number	Maximum number of iterations - processing of the standing order. Only applicable in

					combination with executionMode AFTER_MAX_ITERATION_EXCEEDED. Not supported
+++	maxAmount	[0..0]	ALL	Amount	Maximum amount to be transferred using the standing order. Only applicable in combination with executionMode AFTER_MAX_AMOUNT_EXCEEDED. Not supported.
++++	value	[1..1]	ALL	Amount	Maximum amount of the transfer
++++	currency	[1..1]	ALL	CurrencyCode	Transfer Currency of the maximum amount
++	exceptions	[0..1]	ALL		Elements defining exceptions to executing a standing order.
+++	stoppages	[0..0]	ALL	Array of Max20Text	List of months where there is no payment (only applicable with interval IRREGULAR). Possible values: JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER. Not supported
+++	breaks	[0..1]	ALL	Array of Intervals	Break periods.
++++	validFromDate	[0..1]	ALL	ISODate	Start date of one break period. Standing order will not be processed from this date.
++++	validToDate	[0..1]	ALL	ISODate	End date of one break period. Standing order will not be processed to this date.
++	openDate	[0..1]	ALL	ISODate	Date when the standing order was created.
++	lastDate	[0..1]	ALL	ISODate	Date of the last execution of the standing order.

+	exchangeRateInformation	[0..0]	ALL	The same object as to initiate payments chapter …	
+	chargeBearer	[0..0] [0..0] [0..0] [0..1]	TUZEM SEPA EHP NONEHP	ChargeBearerType1Code	Fee payer. EHP - Not supported
+	chargesAccount	[0..0]	ALL	The same object as to initiate payments chapter …	
+	ultimateDebtor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter …	SEPA - Not supported
+	debtor	[0..0]	ALL	The same object as to initiate payments chapter …	Payer
+	debtorAccount	[1..1]	ALL	The same object as to initiate payments chapter …	Payer's account
+	intermediaryAgent1	[0..0]	ALL	The same object as to initiate payments chapter …	
+	creditorAgent	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter …	SEPA, EHP, NONEHP - Not supported
+	creditor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter …	SEPA, EHP, NONEHP - Not supported
+	creditorAccount	[1..1]	ALL	The same object as to initiate payments chapter …	beneficiary's account
+	ultimateCreditor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter …	SEPA - Not supported
+	purpose	[0..0]	ALL	The same object as to initiate payments chapter …	Purpose of payment
+	instructionForNextAgent	[0..0]	ALL	Instruction code	Instructions for the next bank
+	remittanceInformation	[1..1]	ALL	The same object as to initiate payments chapter …	It is always returned even if nothing has been entered.
+	signInfo	[1..1]	ALL	±	Status information and id of unauthorized transactions
++	state	[1..1]	ALL	StateCode	Transaction

					authorization status
++	signId	[1..1]	ALL	Text	Identifier of the authorization process of a particular transaction. Always returned
+	instructionStatus	[1..1]	ALL	Status Code set	Transaction status identifier
+	statusChangeInfo	[0..1]	ALL	Text	Specifies the change which was made to the instruction. Used with istructionStatus "ACWC". Not supported.

JSON příklad response:


```
{
  "standingOrderIdentification":  {
    "instructionIdentification": "NOTPROVIDED",
    "transactionIdentification": "33838828"
  },
  "amount": {"instructedAmount":  {
    "value": 1,
    "currency": "CZK"
  }},
  "requestedExecutionDate": "2021-10-05",
  "standingOrder":  {
    "alias": "TESTOVACI - AT",
    "execution":  {
      "mode": "UNTIL_DATE",
      "modeDue": "DUE_DAY_OF_MONTH",
      "interval": "MONTHLY"
    },
    "validity": {"lastExecutionDate": "2022-11-30"},
    "exceptions": {},
    "openDate": "2021-06-21",
    "lastDate": "2021-09-03"
  },
  "debtorAccount":  {
    "identification": {"iban": "CZ4101000000115XXXXXXXXXX"},
    "currency": "CZK"
  },
  "creditorAccount":  {
    "identification": {"iban": "CZ0801000000123XXXXXXXXXX"},
    "currency": "CZK"
  },
  "remittanceInformation":  {
    "unstructured": "TESTOVACI",
    "structured": {"creditorReferenceInformation": {"reference":  [
      "VS:111",
      "SS:333",
      "KS:222"
    ]}}
  },
  "paymentTypeInformation": {"serviceLevel": {"code": "DMCT"}},
  "instructionStatus": "ACSP",
```

```
  "signInfo":   {
    "state": "CLOSE",
    "signId": "33838828"
  }
}
```