SA - Creating a contract (createaccess)

Introduction

For processing a purchase transaction in connection with session management or with initiating a subscription.

This request is only available with the version "Access".

Important notes

- For the time-based settlement of goods or subscriptions you must first create offers in your merchant area.
- You have the option to submit values that overwrite the settings for the offer used (e.g. price, term, etc.). These optional submitted values will be used for all following periods of this contract. The original offer is not changed.
- If the time at which access is first granted is in the future or the initial term is free of charge, a reservation, which will later be cancelled automatically, is carried out. The amount for the reservation is one payment unit (e.g. 1 EUR). This procedure is simply used to verify the customer's payment data and to check the customer data using risk checks, provided that these are active. The first "real" booking will be effected at the start of the first period that requires a charge.
- The periods (defined by "period_unit_trail", "period_length_trail", "period_unit_recurring", "period_length_ recurring") are limited to 60 months.

Request "createaccess"

Common Pararameter	Required	Format	Comment
aid	+		Sub-Account ID, defined by PAYONE Format`NUMERIC(5..6)`

clearingtype	+		Format`LIST`		

value	Comment	Description / Comments
elv	Debit payment	According to the new regulation on the transfer of funds (Geldtransferverordnung) address data (name, street, zip, city) is mandatory for cross border bank transfers (EEA / EWR), e.g.: CH, SM, MC, PM, JE, GG. This is mandatory from 2017-11-19. Please use request "updateuser" to update customers address data. This means that the whole address for bank transfers needs to be provided.
cc	Credit card	
rec	Invoice	
cod	Cash on delivery	not allowed for vauthorization, createaccess, updateaccess
vor	Prepayment	not allowed for vauthorization, createaccess, updateaccess
sb	Online Bank Transfer	not allowed for vauthorization, createaccess, updateaccess
wlt	e-wallet	not allowed for vauthorization, createaccess, updateaccess
fnc	Financing	not allowed for vauthorization, createaccess, updateaccess
csh	Cash or Hybrid Payments	not allowed for vauthorization, createaccess, updateaccess

reference	+		Merchant reference number for the payment process (case insensitive) Format`CHAR(1..20)`Permitted Symbols `[0-9][a-z][A-Z], .,-,_,/`

productid	+		ID for the offer Format`NUMERIC(1..7)`

param	-		Individual parameter (per payment process) This parameter is passed on to all follow-up bookings of the same contract Format`CHAR(1..255)`

accessname	-		Customer's user name Format`CHAR(1..32)`

accesscode	-		Customer's password Format`CHAR(1..32)`

access_starttime	-		Access starting date as Unixtimestamp Format`Unixtimestamp`

access_expiretime	-		Can be
			
			• either used with creating a new access (createaccess)
			-> Set expiry date/time of the initial term as Unixtimestamp (e.g. date of first renewal)
			
			    • deprecated, should not be used. Please use "period_unit_trail", "period_length_trail" instead.
			    • do not use with "period_unit_trail", "period_length_trail"
			
			• or used with updating an existing access (updateaccess)
			-> Update expiry date/time of the current term as Unixtimestamp (e.g. overwrite date of next renewal)
			
			Format Unixtimestamp

access_canceltime	-		Date of cancellation as Unixtimestamp in the future (after this date no further renewals will be carried out) Format Unixtimestamp
amount_trail	o		Total gross amount for initial term Must equal the sum (quantity x price) of all items for the initial term. Required when item is submitted. Amount can be "0" (e.g. for test period). Format NUMERIC(1..8), max. value 999 999 99
period_unit_trail	-		Time unit for initial term Do not use with "access_expiretime". Do not exceed 5 years / 60 months. Format LIST {period_unit_trail table below}
period_length_trail	o		Duration of the initial term. Can only be used in combination with period_unit_trail. Required when period_unit_trail is submitted. Do not use with "access_expiretime" Format NUMERIC(1..4)
amount_recurring	o		Total gross amount of all items of one period during the subsequent term Must equal the sum (quantity x price) of all items during the subsequent term. Required when item is submitted. Amount must not be "0". Format NUMERIC(1..8), max. value 999 999 99

period_unit_trail list:

value	Comment
Y	Value "length" is in years
M	Value "length" is in months
D	Value "length" is in days

period_unit_recurring	-			Time unit for subsequent term Do not exceed 5 years / 60 months. Format`LIST`

Time unit for subsequent term

Do not exceed 5 years / 60 months.
Format`LIST`

value	Comment
Y	Value "length" is in years
M	Value "length" is in months
D	Value "length" is in days
N	No subsequent term given

period_length_recurring	o			Duration of the subsequent term. Can only be used in combination with period_unit_recurring. Required when period_length_recurring is submitted. Format`NUMERIC(1..4)`

Parameter (module billing)

vaccountname	+			ID/name of the merchant's billing account Format`CHAR(1..20)`Permitted Symbols `[0-9, a-z, A-Z, .,-,_,/]`

settle_period_length	+			Length of the accounting period Format`NUMERIC(1..3)`

settle_period_unit	+			Time unit of the accounting period Format`LIST`

Time unit of the accounting period
Format`LIST`

value	Comment
Y	Value "length" is in years
M	Value "length" is in months
D	Value "length" is in days

settletime	-			Date for the next invoicing Unixtimestamp in the future. Format`Unixtimestamp`

payout_open_balance	-			Credits will be booked on the most recent payment process. Format`LIST`

Credits will be booked on the most recent payment process.
Format`LIST`

value	Comment
yes	Amount will be credited immediately (default)
no	Credited amount will be offset with the subsequent transaction

Parameter (module Invoicing)

invoice_deliverymode	-		Parameter defines how documents like invoice, credit notes and reminders should be sent to the customer. Format`LIST` | Value	Comments
---	---
M	Postal Mail
P	PDF (via email)
N	no delivery	
invoiceappendix	-		Dynamic text on the invoice Format`CHAR(1..255)`
id_trail[n]	+		Product number, order number, etc. of this item (initial term) Format`CHAR(1..32)`Array Array elements [n] starting with [1]; serially numbered; max [100] Permitted Symbols`[0-9][A-Z][a-z][()[]{} +-_#/:]`
no_trail[n]	+		Quantity of this item (initial term) Format`NUMERIC(5)`Array Array elements [n] starting with [1]; serially numbered; max [100]
pr_trail[n]	+		Unit gross price of the item (initial term) in smallest unit. Format`NUMERIC(8) max. 999 999 99`Array Array elements [n] starting with [1]; serially numbered; max [100]
de_trail[n]	+		Description of this item (initial term) Format`CHAR(1..255)`Array Array elements [n] starting with [1]; serially numbered; max [100] **Example** ``` de[1]=Product 1 de[2]=Product 2 de[3]=Product 3 ... de[100]=Product 100 ```
va_trail[n]	-		VAT rate (% or bp) (first term) Format`NUMERIC(4)`Array Array elements [n] starting with [1]; serially numbered; max [100]
ti_trail[n]	-		Title (initial term) Format`CHAR(1..100)`Array Array elements [n] starting with [1]; serially numbered; max [100]
id_recurring[n]	+		Product number, order number, etc. of this item (subsequent term) Format`CHAR(1..32)`Array Array elements [n] starting with [1]; serially numbered; max [100] Permitted Symbols`[0-9][A-Z][a-z][()[]{} +-_#/:]`

no_recurring[n]	+		Quantity of this item (subsequent term) Format`NUMERIC(5)`Array Array elements [n] starting with [1]; serially numbered; max [100]
pr_recurring[n]	+		Unit gross price of the item (subsequent term) in smallest unit. Format`NUMERIC(8) max. 999 999 99`Array Array elements [n] starting with [1]; serially numbered; max [100]

de_recurring[n]	+		Description of this item (subsequent term)
			Format`CHAR(1..255)`Array
			
			Array elements [n] starting with [1]; serially numbered; max [100]
			
			Example
			<pre>de[1]=Product 1
			de[2]=Product 2
			de[3]=Product 3
			...
			de[100]=Product 100</pre>

va_recurring[n]	-		VAT rate (% or bp) (subsequent term) Format`NUMERIC(4)`Array Array elements [n] starting with [1]; serially numbered; max [100]
ti_recurring[n]	-		Title (subsequent term) Format`CHAR(1..100)`Array Array elements [n] starting with [1]; serially numbered; max [100]
Parameter (personal data)			
customerid	-		Merchant's customer ID, defined by you / merchant to refer to the customer record Format`CHAR(1..20)`Permitted Symbols `[0-9, a-z, A-Z, .,-,_,` `]`"customerid" can be used to identify a customer record. • If "customerid" is used then stored customer data are loaded automatically.
userid	-		PAYONE User ID, defined by PAYONE Format`NUMERIC(6..12)`
salutation	-		The customer's salutation Format`CHAR(1..10)`

title	-		Title of the customer
			Format`CHAR(1..20)`Samples`Dr`
			
			`Prof.`
			
			`Dr.-Ing.`

firstname	o		First name of customer; optional if company is used, i.e.: you may use • "company" • or "lastname" • or "firstname" plus "lastname" Format`CHAR(1..50)`

lastname	+		Last name of customer; optional if company is used, i.e.: you may use - "company" - or "lastname" - or "firstname" plus "lastname" Format`CHAR(2..50)`
company	-		Company name of customer; The company name is optional if lastname is used, i.e.: you may use - "company" - or "lastname" - or "firstname" plus "lastname" Format`CHAR(2..50)`
street	-		Street number and name (required: at least one character) Format`CHAR(1..50)`
addressaddition	-		Specifies an additional address line for the invoice address of the customer. Format`CHAR(1..50)`Samples`7th floor` `c/o Maier`
zip	-		Postcode Format`CHAR(2..10)`Permitted Symbols`[0-9][A-Z][a-z][` `_.-/ ]`
city	-		City of customer Format`CHAR(2..50)`

country	+		Specifies country of address for the customer
			Format`LIST`
			Permitted values`ISO 3166 2-`
			`letter-codes` Samples`DE`
			
			`GB`
			
			`US`
			
			Some countries require additional information in parameter "state"

email	-		email-address of customer Format`CHAR(5..254)`Permitted Symbols`RFC 5322`
telephonenumber	-		Phone number of customer Format`CHAR(1..30)`
birthday	-		Date of birth of customer Format`DATE(8), YYYYMMDD`Samples`20190101` `19991231`

language	-		Language indicator (ISO 639) to specify the language that should be presented to the customer (e.g. for error messages, frontend display). If the language is not transferred, the browser language will be used. For a non-supported language English will be used. Format `LIST` Permitted values `ISO 639-1 (Language)` `2-letter-codes`
vatid	-		VAT identification number. Used for b2b transactions to indicate VAT number of customer. Format `CHAR(1..50)`
ip	-		Customer's IP-V4-address (123.123.123.123) or IP-V6-address Format `CHAR(1..39)`
Parameter (debit payment)			
iban	o		IBAN to be used for payment or to be checked Format `CHAR(10..34)` Permitted Symbols `[0-9][A-Z]`

bic	o		Bank Identifier Code to be used for payment or to be checked
			Format
			`CHAR(8 or 11) Only capital letters and digits,`
			`no spaces`
			
			Permitted Symbols `[0-9][A-Z]` BIC is optional for all Bank transfers within SEPA. For Accounts from Banks outside of SEPA, BIC is still required.

bankcountry	o		Account type/ country for use with BBAN (i.e. bankcode, bankaccount): DE DE: Mandatory with bankcode, bankaccount, optional with IBAN For other countries than DE please use IBAN or IBAN/BIC Format `LIST`
bankaccount	o		Account number (BBAN) • DE: bankcountry, bankcode and bankaccount may be used. Then IBAN will be generated by PAYONE platform and used for SEPA transactions. • Not DE: Please use IBAN or IBAN / BIC. Format `NUMERIC(1..10)`
bankcode	o		Sort code (BBAN) (only in DE) • DE: bankcountry, bankcode and bankaccount may be used. Then IBAN will be generated by PAYONE platform and used for SEPA transactions. • Not DE: Please use IBAN or IBAN / BIC. Format `NUMERIC(8)`
bankaccountholder	-		Account holder Format `CHAR(1..50)`

| mandate_identification | o | | A SEPA mandate can be created if a payment is initiated (amount > 0). Can be used to enforce a merchant specific mandate identification. The mandate_identification has to be unique. Format`CHAR(1..35)` Permitted Symbols `[A-Z,a-z,0-9,+,-,.,,(,)]` If the mandate_identification is not set PAYONE will create an unique mandate identification (pattern: PO-nnnnnnnnnn).

PPS (PAYONE Payment Service): This parameter must not be used! For PPS the PAYONE platform defines the mandate_identification |

Parameter (credit card)

| cardpan | + | | Primary account number of credit card Format`NUMERIC(13..19)`

• if your system handles "cardpan" directly you can not be PCI DSS SAQ A compliant.
• for simple PCI DSS SAQ A compliance please use PAYONE hosted iFrames together with pseudocardpan. |

| cardtype | + | | Card type of credit card Format`LIST` |

value	Comment	BIN-Range for automatic card type detection
V	Visa	4
M	MasterCard	51-55,2221-2720
A	American Express	34, 37
D	Diners / Discover	300-305,3095,36,38,39 601, 64, 65
~~C~~	~~Discover~~	
J	JCB	3528-3589
O	Maestro International	50, 56-58,602, 61, 620, 627, 63, 67
P	China Union Pay	62212600-62299800,624-626, 6282-6288
U	UATP / Airplus	1220, 1920 -> coming soon; not available yet
G	girocard	68 *

*girocard is currently only viable for e-commerce-payments via Apple Pay.

| cardexpiredate | + | | Credit card expiry date YYMM

Format`NUMERIC(4), YYMM` |

cardcvc2	o		Credit card security number Format`NUMERIC(3..4)` For SAQ A compliance: [PAYONE Frontend hosted iFrame](#) must be used. This parameter must not be used.
cardissuenumber	-		Card issue number (only Maestro UK cards) Format`NUMERIC(1..2)`
cardholder	-		Cardholder of credit card. Format`CHAR(1..50)`
pseudocardpan	o		Pseudo primary account number for a credit card. Format`NUMERIC(16..19)`

Parameter (3-D Secure)

xid	+		3-D Secure transaction ID (if the request "3dscheck" was used previous transactions), i.e.: • value for "xid" is **returned** by request 3dscheck (optional request) • and then has to be used for request authorization, preauthorization or createaccess in request data Format`CHAR(1..60)`
cavv	-		3-D Secure authentication value Format`CHAR(1..40)`
eci	-		3-D Secure e-commerce indicator Format`CHAR(1..2)`
customer_is_present	o		Indicates whether customer is "present" and can enter their data in the shop (=yes). Or customer is not present and can not enter any data (=no). Format`LIST` Permitted Values`yes` `no`

successurl	o		• either "xid" (from previous 3dscheck)
			• or "successurl", "backurl" must be given
			• or default "successurl" and "backurl" for a portal are configured within PMI
			
			URL for "payment successful"
			Format`CHAR(2..255)`Scheme
			
			`<scheme>://<host>/<path>`
			
			`<scheme>://<host>/<path>?<query>`
			
			`scheme-pattern: [a-zA-Z]{1}[a-zA-Z0-9]{1,9}`

errorurl	-		URL for "faulty payment"
			Format`CHAR(2..255)`Scheme
			
			`<scheme>://<host>/<path>`
			
			`<scheme>://<host>/<path>?<query>`
			
			`scheme-pattern: [a-zA-Z]{1}[a-zA-Z0-9]{1,9}`

backurl	o		• either "xid" (from previous 3dscheck)
			• or "successurl", "backurl" must be given
			• or default "successurl" and "backurl" for a portal are configured within PMI
			
			URL for "Back" or "Cancel"
			Format`CHAR(2..255)`Scheme
			
			`<scheme>://<host>/<path>`
			
			`<scheme>://<host>/<path>?<query>`
			
			`scheme-pattern: [a-zA-Z]{1}[a-zA-Z0-9]{1,9}`

Response "createaccess"

Common Parameter	Required	Format	Comment
status	+	Default	APPROVED / REDIRECT / PENDING / ERROR
Parameter (APPROVED)			
txid	+		The txid specifies the payment process within the PAYONE platform Format`NUMERIC(9..12)`
userid	+		PAYONE User ID, defined by PAYONE Format`NUMERIC(6..12)`
accessid	+		Access ID, defined by PAYONE Format`NUMERIC(3..12)`
Parameter (REDIRECT)			
txid	+		The txid specifies the payment process within the PAYONE platform Format`NUMERIC(9..12)`
userid	+		PAYONE User ID, defined by PAYONE Format`NUMERIC(6..12)`
redirecturl	+		Redirect URL zMerchant system has to redirect customer to this URL to complete payment Format`CHAR(2..2000)`
Parameter (PENDING)			

txid	+		The txid specifies the payment process within the PAYONE platform Format`NUMERIC(9..12)`
userid	+		PAYONE User ID, defined by PAYONE Format`NUMERIC(6..12)`
accessid	+		Access ID, defined by PAYONE Format`NUMERIC(3..12)`
Parameter (ERROR)			
errorcode	+		In case of error the PAYONE Platform returns an error code for your internal usage. Format`NUMERIC(1..6)`
errormessage	+		In case of error the PAYONE Platform returns an error message for your internal usage. Format`CHAR(1..1024)`
customermessage	-		The customermessage is returned to your system in order to be displayed to the customer. (Language selection is based on the end customer's language, parameter "language") Format`CHAR(1..1024)`