







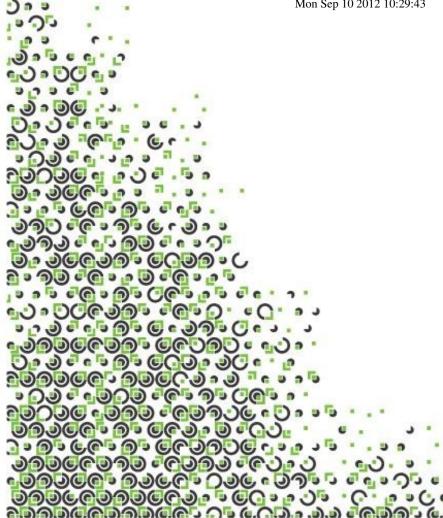




# Handbuch zur Implementierung des sofort Gateway

# SofortLib

1.5.1



Mon Sep 10 2012 10:29:43

# **Main Page**

Base class for Payment Network XML-Api This class implements basic http authentication and a xml-parser for parsing response messages

Requires libcurl and openssl

Copyright (c) 2012 SOFORT AG

#### Date:

2012-09-10 10:26:54 +0200 (Mon, 10 Sep 2012)

#### Version

SofortLib 1.5.0

Id:

sofortLib.php 5341 2012-09-10 08:26:54Z Niehoff

#### Author

SOFORT AG http://www.sofort.com(integration@sofort.com)

4 Main Page

# **Namespace Index**

2.1	<b>Package</b>	List
-----	----------------	------

Here are the pa	ack	age	es '	wit	h	br	ief	d	le	sc	ri	pt	io	ns	s (	if	av	/a	ila	ıbl	le)	:							
sofortLib																												1	1

# **Class Index**

# 3.1 Class Hierarchy

This inheritance list is sorted roughly, I	but not completely, alphabetically
--	------------------------------------

PnagInvoice
SofortLib
SofortLib_CancelSa
SofortLib_CancelSr
SofortLib_ClassicNotification
SofortLib_ConfirmSr
SofortLib_Debit
SofortLib_EditSr
SofortLib_iDeal_Banks
SofortLib_Multipay
SofortLib_Notification
SofortLib_Refund
SofortLib_SofortueberweisungClassic
SofortLib_iDealClassic
Safart Lib. Transaction Data

8 Class Index

# **Class Index**

# 4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

PnagInvoice
SofortLib
SofortLib_CancelSa
SofortLib_CancelSr
SofortLib_ClassicNotification
SofortLib_ConfirmSr
SofortLib_Debit
SofortLib_EditSr
SofortLib_iDeal_Banks
SofortLib_iDealClassic
SofortLib_Multipay
SofortLib_Notification
SofortLib_Refund
SofortLib_SofortueberweisungClassic
SofortLib TransactionData

10 Class Index

# **Namespace Documentation**

# 5.1 Package sofortLib

# **5.1.1 Detailed Description**

Copyright (c) 2012 SOFORT AG

Date:

2012-09-06 12:05:14 +0200 (Thu, 06 Sep 2012)

Version

Id:

class.invoice.inc.php 5320 2012-09-06 10:05:14Z dehn

Author

SOFORT AG http://www.sofort.com(integration@sofort.com)

# **Class Documentation**

# **6.1** PnagInvoice Class Reference

#### **Public Member Functions**

- \_\_construct (\$configKey, \$transactionId= ")
- getConstantById (\$id)
- getConstantByName (\$name)
- setBitmask (\$status, \$statusReason, \$invoiceStatus)
- setState (\$state, \$callback=")
- getState ()
- setTransactionId (\$transactionId)
- refreshTransactionData ()
- cancelInvoice (\$transactionId= ", \$creditNoteNumber= ")
- confirmInvoice (\$transactionId= ", \$invoiceNumber= ", \$customerNumber= ", \$orderNumber= ")
- updateInvoice (\$transactionId, \$items, \$comment, \$invoiceNumber= ", \$customerNumber= ", \$orderNumber= ")
- updateOrderNumber (\$transactionId, \$orderNumber)
- addItemToInvoice (\$itemId, \$productNumber, \$title, \$unitPrice, \$product-Type=0, \$description=", \$quantity=1, \$tax=19)
- removeItemfromInvoice (\$itemId)
- addShippingAddress (\$firstname, \$lastname, \$street, \$streetNumber, \$zipcode, \$city, \$salutation, \$country= 'DE', \$nameAdditive= ", \$streetAdditive= ", \$companyName=")
- addShippingAddresss (\$firstname, \$lastname, \$street, \$streetNumber, \$zipcode, \$city, \$salutation, \$country= 'DE', \$nameAdditive= ", \$streetAdditive= ")
- addInvoiceAddress (\$firstname, \$lastname, \$street, \$streetNumber, \$zipcode, \$city, \$salutation, \$country= 'DE', \$nameAdditive= ", \$streetAdditive= ", \$companyName=")
- setOrderId (\$arg)
- setCustomerId (\$arg)

- setPhoneNumberCustomer (\$arg)
- setEmailCustomer (\$arg)
- addUserVariable (\$arg)
- setNotificationUrl (\$arg)
- setAbortUrl (\$arg)
- setSuccessUrl (\$arg)
- setTimeoutUrl (\$arg)
- setTimeout (\$arg)
- setReason (\$reason1, \$reason2=")
- setAmount (\$arg, \$currency= 'EUR')
- getAmount ()
- setSofortrechnung ()
- setDebitorVatNumber (\$vatNumber)
- getPaymentUrl ()
- getInvoiceTye ()
- getTransactionId ()
- checkout ()
- getInvoice ()
- setInvoiceUrl (\$invoiceUrl)
- getInvoiceUrl ()
- setStatus (\$status)
- setStatusReason (\$statusReason)
- setStatusOfInvoice (\$invoiceStatus=")
- setLanguageCode (\$languageCode=")
- setTransaction (\$transaction)
- setTime (\$time)
- setVersion (\$arg)
- setPaymentMethod (\$paymentMethod)
- setInvoiceObjection (\$invoiceObjection)
- setInvoiceStatus (\$invoiceStatus)
- getInvoiceObjection ()
- getStatusOfInvoice ()
- getInvoiceStatus ()
- getPaymentMethod ()
- getStatusReason ()
- getStatus ()
- getLanguageCode ()
- getItems ()
- setItems (\$items)
- getTransactionData ()
- isError ()
- isWarning ()
- getError ()
- getErrors ()
- getErrorCodes (\$detailLevel=0)
- getWarnings ()

- enableLog()
- disableLog ()
- log (\$message)
- logError (\$message)
- logWarning (\$message)

# **6.1.1 Detailed Description**

Abstraction of an invoice Helper class to ease usage of "Rechnung by sofort" Encapsulates Multipay, TransactionData and ConfirmSr to handle everything there is about "Rechnung by sofort"

#### See also

```
SofortLib_Multipay
SofortLib_TransactionData
SofortLib_ConfirmSr
```

#### **6.1.2** Constructor & Destructor Documentation

```
6.1.2.1 __construct($ configKey, $ transactionId = ")
```

Constructor

# **Parameters**

```
string $apiKeystring $transactionIdstring $apiUrl
```

## **6.1.3** Member Function Documentation

```
6.1.3.1 addInvoiceAddress ( $ firstname, $ lastname, $ street, $ streetNumber, $ zipcode, $ city, $ salutation, $ country = 'DE', $ nameAdditive = ", $ streetAdditive = ", $ companyName = " )
```

Wrapper for SofortLib\_Multipay.setSofortrechnungInvoiceAddress

#### See also

```
SofortLib_Multipay
```

## **Parameters**

\$firstname

\$lastname

\$street

```
$streetNumber
    $zipcode
    $city
    $salutation
    $country (optional, default: DE)
    $nameAdditive (optional)
    $streetAdditive (optional)
    $companyName (optional)
6.1.3.2 addItemToInvoice ($ itemId, $ productNumber, $ title, $ unitPrice, $
        productType = 0, $ description = ", $ quantity = 1, $ tax = 19)
Wrapper for SofortLib_Multipay.addSofortrechnungItem
See also
    SofortLib_Multipay
Parameters
    $itemId
    $productNumber
    $title
    $unit_price - float precision 2
See also
    SofortLib_Multipay api
Parameters
    $productType
    $description
    $quantity - int
    $tax
6.1.3.3 addShippingAddress ($ firstname, $ lastname, $ street, $
        streetNumber, $ zipcode, $ city, $ salutation, $ country = 'DE', $
        nameAdditive = ", $ streetAdditive = ", $ companyName = " )
Wrapper for SofortLib_Multipay.setSofortrechnungShippingAddress
See also
    SofortLib_Multipay
```

#### **Parameters**

```
$firstname
```

\$lastname

\$street

\$streetNumber

\$zipcode

\$city

\$salutation

\$country (optional, default: DE)

\$nameAdditive (optional)

\$streetAdditive (optional)

\$companyName (optional)

# 6.1.3.4 addShippingAddresss (\$ firstname, \$ lastname, \$ street, \$ streetNumber, \$ zipcode, \$ city, \$ salutation, \$ country = 'DE', \$ nameAdditive = ", \$ streetAdditive = ")

 $Wrapper\ for\ SofortLib\_Multipay.setSofortrechnungShippingAddress$ 

#### See also

 $SofortLib\_Multipay$ 

#### **Parameters**

\$firstname

\$lastname

\$street

\$streetNumber

\$zipcode

\$city

\$salutation

\$country (optional, default: DE)

\$nameAdditive (optional)

\$streetAdditive (optional)

# 6.1.3.5 addUserVariable (\$ arg)

 $Wrapper\ function\ for\ SofortLib\_Multipay.addUserVariable$ 

# **Parameters**

\$arg

#### 6.1.3.6 cancelInvoice ( \$ transactionId = ", \$ creditNoteNumber = ")

Wrapper function for cancelling this invoice via SofortLib\_Multipay (SofortLib)

#### **Returns**

Ambigious boolean/Array

#### **6.1.3.7** checkout ( )

Validate your parameters against API

#### Returns

```
array - any validationerrors and -warnings send the order to pnag (-> buy your products)
```

empty array if ok ELSE array with errors and/or warnings

```
6.1.3.8 confirmInvoice ( $ transactionId = ", $ invoiceNumber = ", $ customerNumber = ", $ orderNumber = " )
```

Wrapper function for confirming this invoice via SofortLib\_Multipay (SofortLib)

#### **Parameters**

```
    $transactionId - optional parameter for confirming a transaction on the fly
    $invoiceNumer - optional parameter for own invoice number
    $customerNumber - optional parameter for own customer number
    $orderNumber - optional parameter for own order number
```

#### Returns

Ambigious boolean/Array

# **6.1.3.9 disableLog**( )

Disable logging for all encapsed SofortLib components

#### Returns

boolean

#### **6.1.3.10** enableLog ( )

Enabling logging for all encapsed SofortLib components

## Returns

boolean

```
6.1.3.11 getAmount ( )
current total amount of the given order-articles
Returns
    float - sum (price, total) of all articles
6.1.3.12 getConstantById ( $ id )
Getter for a class constant
Parameters
    int $id
Returns
    string
6.1.3.13 getConstantByName ( $ name )
Getter for a class constant
Parameters
    string $name
Returns
    int
6.1.3.14 getError ( )
returns one error (as String!)
6.1.3.15 getErrorCodes ( $ detailLevel = 0 )
Ouputs errors in a more convenient array to let users easily iterate
Parameters
    int $detailLevel
6.1.3.16 getErrors ( )
collect all errors and returns them
```

Returns

array - all errors

#### **6.1.3.17 getInvoice** ( )

Output the resulting invoice as pdf, if possible Function uses file\_get\_contents, if allow\_url\_fopen is allowed in php.ini (might be disabled on shared hosting) As a fall-back, downloading via cURL, when module cURL is available If neither file\_get\_contents nor cURL is available for downloading, a connection via socket is used to download.

#### **Returns**

boolean

#### 6.1.3.18 getInvoiceObjection ( )

Returns the reason for objecting this invoice

#### Returns

string

# 6.1.3.19 getInvoiceStatus ( )

Uses the statusMask to "calculate" the current invoice's payment status

# See also

Invoice. calcInvoiceStatusCode

#### Returns

int

#### 6.1.3.20 getInvoiceTye ( )

Wrapper function for SofortLib\_Multipay.geInvoiceType

#### Returns

(OR or LS)

# 6.1.3.21 getInvoiceUrl()

Getter for retrieving the invoice's url

#### Returns

url string

```
6.1.3.22 getItems ( )
Getter for items
Returns
    array
 \textbf{6.1.3.23} \quad \textbf{getLanguageCode} \left( \quad \right) 
Getter for language code
Returns
    string
6.1.3.24 getPaymentMethod ( )
Getter for payment_method
Returns
    string
6.1.3.25 getPaymentUrl()
Wrapper function for SofortLib_Multipay.getPaymentUrl
Returns
    url string
6.1.3.26 getState ( )
Getter for the current state
Returns
    int $this->state
6.1.3.27 getStatus ( )
Getter for status
Returns
    string
```

#### 6.1.3.28 getStatusOfInvoice()

Instead of calculated status, this method returns the invoice's staus (string)

#### **Returns**

string

# 6.1.3.29 getStatusReason()

Getter for status\_reason

#### Returns

string

#### 6.1.3.30 getTransactionData()

return TransactionData, the invoice is working with NOTICE: if status changed (removeArticle, InvoiceConfirmed etc.) it returns always the FRESH TransactionData from pnag-server

#### **Returns**

object

#### See also

\$this->refreshTransactionData();

# **6.1.3.31** getTransactionId ( )

 $Wrapper\ function\ for\ SofortLib\_Multipay.getPaymentUrl$ 

#### **Returns**

url string

# **6.1.3.32** getWarnings ( )

collects all warnings and returns them

#### Returns

array

#### **6.1.3.33** isError ( )

Check, if errors occured

#### Returns

boolean

## **6.1.3.34** isWarning ( )

Check, if warnings occured

#### Returns

boolean

# 6.1.3.35 log(\$ message)

Log the given String into log.txt Notice: logging must be enabled -> use enableLog();

#### **Parameters**

```
string $msg - Message to log
```

#### Returns

bool - true=logged ELSE false=logging failed

# **6.1.3.36** logError ( \$ message )

Log the given String into error\_log.txt Notice: logging must be enabled -> use enableLog();

# **Parameters**

```
string $msg - Message to log
```

### Returns

bool - true=logged ELSE false=logging failed

# 6.1.3.37 logWarning ( \$ message )

Log the given String into warning\_log.txt

#### **Parameters**

```
string $msg - Message to log
```

#### Returns

bool - true=logged ELSE false=logging failed

# 6.1.3.38 refreshTransactionData ( )

Refreshes the TransactionData with the data directly from the pnag-server

#### Returns

boolean

# **6.1.3.39** removeItemfromInvoice (\$ itemId )

Remove an item from the invoice

#### **Parameters**

\$itemId

#### **Returns**

boolean

# **6.1.3.40** setAbortUrl ( \$ arg )

Wrapper function for SofortLib\_Multipay.setAbortUrl

#### **Parameters**

\$arg

# **6.1.3.41** setAmount ( \$ arg, \$ currency = 'EUR' )

Wrapper function for SofortLib\_Multipay.setAmount

#### **Parameters**

\$arg float

\$currency string

#### 6.1.3.42 setBitmask (\$ status, \$ statusReason, \$ invoiceStatus )

Set the bitmask to a specific state

#### **Parameters**

```
string $status
string $status_reason
string $invoice_status
```

#### **Returns**

```
string pending - confirm_invoice - empty -> 4195329
```

# 6.1.3.43 setCustomerId (\$ arg)

 $Wrapper\ function\ for\ SofortLib\_Multipay.setSofortrechnungCustomerId$ 

#### **Parameters**

\$arg

#### **6.1.3.44** setDebitorVatNumber ( \$ vatNumber )

Wrapper function for SofortLib\_Multipay.setDebitorVatNumber

# 6.1.3.45 setEmailCustomer (\$ arg)

Wrapper function for SofortLib\_Multipay.setEmailCustomer

#### **Parameters**

\$arg

# **6.1.3.46** setInvoiceObjection ( \$ invoiceObjection )

Sets the reason for objecting this invoice

#### **Parameters**

```
string $invoiceObjection (40-50 chars max.)
```

#### Returns

object

# 6.1.3.47 setInvoiceStatus ( \$ invoiceStatus ) Sets the invoice status Parameters

string \$invoiceStatus

#### Returns

object

# 6.1.3.48 setInvoiceUrl ( \$ invoiceUrl )

Setter for invoiceUrl

#### **Parameters**

\$invoiceUrl

#### Returns

object

# **6.1.3.49** setItems ( \$ *items* )

Setter for invoice items, takes an array of PnagArticle objects

#### **Parameters**

array \$items

# 6.1.3.50 setLanguageCode ( \$ languageCode = " )

Setter for language\_code

### **Parameters**

string \$language\_code | fallback en

#### Returns

object

# 6.1.3.51 setNotificationUrl(\$ arg)

Wrapper function for SofortLib\_Multipay.setNotificationUrl

#### **Parameters**

\$arg

```
6.1.3.52 setOrderId ( $ arg )
```

 $Wrapper\ function\ for\ SofortLib\_Multipay.setSofortrechnungOrderId$ 

#### See also

SofortLib\_Multipay

#### **Parameters**

\$arg

# 6.1.3.53 setPaymentMethod ( \$ paymentMethod )

Setter for payment\_method

#### **Parameters**

\$paymentMethod

#### Returns

object

# 6.1.3.54 setPhoneNumberCustomer ( \$ arg )

Wrapper function for SofortLib\_Multipay.setPhoneNumberCustomer

# **Parameters**

\$arg

# 6.1.3.55 setReason( \*reason1, \*reason2 = ")

Wrapper function for SofortLib\_Multipay.setReason

#### **Parameters**

```
$reason1 string
$reason2 string
```

# 6.1.3.56 setSofortrechnung()

Wrapper function for SofortLib\_Multipay.setSofortrechnung

```
6.1.3.57 setState ( $ state, $ callback = " )
Set the state An optional callback can be registered
Parameters
    int $state
    function $callback
6.1.3.58 setStatus ( $ status )
Setter for status
Parameters
    $status
Returns
    object
6.1.3.59 setStatusOfInvoice ( $ invoiceStatus = " )
Setter for invoice_status
Parameters
    string $invoice_status | may be emtpy
Returns
    object
6.1.3.60 setStatusReason ( $ statusReason )
Setter for status_reason
Parameters
    $status_reason
Returns
    object
```

```
6.1.3.61 setSuccessUrl($ arg)
Wrapper function for SofortLib_Multipay.setSuccessUrl
Parameters
    $arg
6.1.3.62 setTime ( $ time )
Setter for time
Parameters
    $time return object
6.1.3.63 setTimeout ( $ arg )
Wrapper function for SofortLib_Multipay.setTimeout
Parameters
    $arg
6.1.3.64 setTimeoutUrl($ arg)
Wrapper function for SofortLib_Multipay.setTimeoutUrl
Parameters
    $arg
6.1.3.65 setTransaction ( $ transaction )
Setter for transaction
Parameters
    $transaction
```

Returns

object

#### **6.1.3.66** setTransactionId ( \$ transactionId )

Setter for transactionId

#### **Parameters**

\$transactionId

#### 6.1.3.67 setVersion ( \$ arg )

Setter for interface version Wrapper for class Multipay to set version according to shop module and it's interface version e.g. 'pn\_xtc\_5.0.0'

#### **Parameters**

**\$arg** string

# 6.1.3.68 updateInvoice ( \$ transactionId, \$ items, \$ comment, \$ invoiceNumber = ", \$ customerNumber = ", \$ orderNumber = " )

Wrapper function for removing an article via SofortLib\_Multipay (SofortLib)

#### **Parameters**

return array

```
$transactionId string
$PnagArticels array
$comment int
```

# 6.1.3.69 updateOrderNumber ( \$ transactionId, \$ orderNumber )

Wrapper function for updating order number after order hast been placed

# **Parameters**

```
string $transactionId
string $orderNumber
```

The documentation for this class was generated from the following file:

· class.invoice.inc.php

# 6.2 SofortLib Class Reference

# **Public Member Functions**

• \_\_construct (\$userId=", \$apiKey=", \$apiUrl=")

- getWarnings (\$paymentMethod= 'all', \$message= ")
- getErrors (\$paymentMethod= 'all', \$message= ")
- handleErrors (\$errors)
- getError (\$paymentMethod= 'all', \$message= ")
- isWarning (\$paymentMethod= 'all', \$message= ")
- isError (\$paymentMethod= 'all', \$message= ")
- deleteAllWarnings ()
- deleteAllErrors ()
- error (\$message, \$fatal=false)
- fatalError (\$message)
- enableLog()
- disableLog ()
- setLogEnabled ()
- setLogDisabled ()
- setLogger (\$SofortLibLogger)
- logWarning (\$message)
- logError (\$message)
- log (\$message)
- setApiVersion (\$version)

#### **Protected Member Functions**

• \_sendMessage (\$message)

# **6.2.1 Detailed Description**

Include any available helper here

Basic PHP Library for communication with multipay API and related products of so-fort.com

## Author

payment-network.com

#### **6.2.2** Constructor & Destructor Documentation

6.2.2.1 \_\_construct ( 
$$userId = ", apiKey = ", apiUrl = " )$$

Constructor

#### **Parameters**

string \$apiKey

# **6.2.3** Member Function Documentation

```
6.2.3.1 _sendMessage($ message) [protected]
```

internal send-method, will check http-errorcode and return body

#### **Parameters**

String \$message message to post

#### Returns

string error or body

# 6.2.3.2 deleteAllErrors ( )

delete all errors

# 6.2.3.3 deleteAllWarnings ( )

delete all warnings

#### **6.2.3.4 disableLog**()

See also

SofortLib setLogDisabled

# **6.2.3.5** enableLog()

See also

SofortLib setLogEnabled

# **6.2.3.6** error ( \$ message, \$ fatal = false )

#### **Parameters**

```
unknown_type $message
unknown_type $fatal
```

# 6.2.3.7 fatalError ( \$ message )

error while parsing xml

## **Parameters**

unknown\_type \$message

#### 6.2.3.8 getError ( \$ paymentMethod = 'all', \$ message = ")

returns one error message

#### See also

getErrors() for more detailed errors

#### **Parameters**

array \$message response array

#### Returns

string errormessage ELSE false

# 6.2.3.9 getErrors ( \$ paymentMethod = 'all', \$ message = ")

Getter for errors

#### **Parameters**

(optional) array \$message response array

#### Returns

emtpy array if no error exist ELSE array with error-codes and error-messages

# 6.2.3.10 getWarnings ( \$ paymentMethod = 'all', \$ message = ")

Getter for warnings

## Returns

empty array if no warnings exists ELSE array with warning-codes and warning-messages

#### **6.2.3.11** handleErrors ( \$ errors )

Alter error array and set error message and error code together as one

#### **Parameters**

array \$errors

#### 6.2.3.12 isError ( \$paymentMethod = 'all', \$message = ")

checks (response)-array for error

#### **Parameters**

```
array $message response array
string $paymentMethod - 'all', 'sr', 'su', 'sv', 'sa', 'ls', 'sl', 'sf' (if unknown then
it uses "all")
```

#### Returns

boolean true if errors found (in given payment-method or in 'global') ELSE false

#### 6.2.3.13 is Warning ( \$paymentMethod = 'all', \$message = ")

checks (response)-array for warnings

#### **Parameters**

```
array $message response array
string $paymentMethod - 'all', 'sr', 'su', 'sv', 'sa', 'ls', 'sl', 'sf' (if unknown then
it uses "all")
```

#### Returns

boolean true if warnings found ELSE false

#### 6.2.3.14 log ( \$ message )

log the given string into log.txt use \$this->enableLog(); to enable logging before!

#### **Parameters**

string \$message

#### **6.2.3.15** logError ( \$ message )

log the given string into error\_log.txt use \$this->enableLog(); to enable logging before!

#### **Parameters**

string \$message

#### **6.2.3.16** logWarning ( \$ message )

log the given string into warning\_log.txt use \$this->enableLog(); to enable logging before!

#### **Parameters**

```
string $message
```

# 6.2.3.17 setApiVersion ( \$ version )

sets the api version which should be used

#### **Parameters**

```
float $version
```

# 6.2.3.18 setLogDisabled ( )

```
disableLog();
```

### 6.2.3.19 setLogEnabled()

enableLog();

#### 6.2.3.20 setLogger ( \$ SofortLibLogger )

Set the SofortLibLogger object

## Parameters

```
object $SofortLibLogger
```

The documentation for this class was generated from the following file:

· sofortLib.php

# 6.3 SofortLib\_CancelSa Class Reference

# **Public Member Functions**

- \_\_construct (\$configKey=")
- removeSofortDauerauftrag (\$transaction)
- setSuccessUrl (\$arg)
- setAbortUrl (\$arg)
- setTimeoutUrl (\$arg)
- setTransaction (\$arg)

# **6.3.1** Detailed Description

Copyright (c) 2012 SOFORT AG

The base class for cancelling SofortDauerauftrag

#### Date:

```
2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)
```

#### Version

SofortLib 1.5.0

#### Id:

sofortLib\_cancel\_sa.inc.php 5301 2012-09-05 12:27:56Z dehn

#### **Author**

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

#### **6.3.2** Constructor & Destructor Documentation

#### 6.3.2.1 $\_$ construct ( \$ configKey = " )

create new cancel object

# **Parameters**

String \$apikey your API-key

# **6.3.3** Member Function Documentation

# **6.3.3.1** removeSofortDauerauftrag ( \$ transaction )

remove SofortDauerauftrag

#### **Parameters**

String \$transaction Transaction ID

# Returns

SofortLib\_CancelSa

#### 6.3.3.2 setAbortUrl (\$ arg)

the customer will be redirected to this url if he uses the abort link on the payment form, should redirect him back to his cart or to the payment selection page

# **Parameters**

string \$arg url for aborting the transaction

#### Returns

SofortLib\_Multipay

# 6.3.3.3 setSuccessUrl(\$ arg)

the customer will be redirected to this url after a successful transaction, this should be a page where a short confirmation is displayed

#### **Parameters**

string \$arg the url after a successful transaction

#### Returns

SofortLib\_Multipay

# 6.3.3.4 setTimeoutUrl(\$ arg)

if the customer takes too much time or if your timeout is set too short he will be redirected to this page

#### **Parameters**

string \$arg url

#### Returns

SofortLib\_Multipay

# 6.3.3.5 setTransaction (\$ arg)

Set the transaction you want to confirm/change

#### **Parameters**

String \$arg Transaction Id

#### Returns

SofortLib\_CancelSa

The documentation for this class was generated from the following file:

• sofortLib\_cancel\_sa.inc.php

# 6.4 SofortLib\_CancelSr Class Reference

# **Public Member Functions**

- \_\_construct (\$configKey= ")
- setTransaction (\$transaction, \$invoice=0)
- setComment (\$comment, \$invoice=0)
- cancelInvoice (\$transaction=", \$invoice=0)
- getInvoiceUrl ()

# **6.4.1 Detailed Description**

```
This class is for confirming and changing statuses of invoices eg: $confirmObj = new SofortLib_CancelSr('yourapikey');
```

\$confirmObj->confirmInvoice('1234-456-789654-31321')->sendRequest();

Copyright (c) 2012 SOFORT AG

#### Date:

2012-05-21 16:53:26 +0200 (Mo, 21 Mai 2012)

# Version

SofortLib 1.5.0

Id:

sofortLib\_cancel\_sr.inc.php 4191 2012-05-21 14:53:26Z niehoff

#### Author

SOFORT AG http://www.sofort.com(integration@sofort.com)

# 6.4.2 Constructor & Destructor Documentation

```
6.4.2.1 \_construct ( $ configKey = " )
```

create new confirm object

# **Parameters**

String \$apikey your API-key

#### **6.4.3** Member Function Documentation

# 6.4.3.1 cancelInvoice ( \$ transaction = ", \$ invoice = 0)

cancel the invoice

#### **Parameters**

string \$transaction the transaction id

#### Returns

SofortLib\_ConfirmSr

# 6.4.3.2 getInvoiceUrl()

after you you changed/confirmed an invoice you can download the new invoice-pdf with this function

#### Returns

string url

# 6.4.3.3 setComment (\$ comment, \$ invoice = 0)

set a comment for refunds

#### **Parameters**

string \$arg

# 6.4.3.4 setTransaction ( \$ transaction, \$ invoice = 0 )

Set the transaction you want to confirm/change

# Parameters

String \$transaction Transaction Id

#### Returns

 $SofortLib\_ConfirmSr$ 

The documentation for this class was generated from the following file:

• sofortLib\_cancel\_sr.inc.php

# 6.5 SofortLib\_ClassicNotification Class Reference

# **Protected Member Functions**

• \_getHashHexValue (\$data, \$hashFunction= 'sha1')

# **6.5.1** Detailed Description

Instance of this class handles the callback of SOFORT to notify about a status change, the classic way to do so

Copyright (c) 2012 SOFORT AG

#### Date:

```
2012-09-05 14:28:13 +0200 (Wed, 05 Sep 2012)
```

#### Version

SofortLib 1.5.0

#### Id:

sofortLib\_classic\_notification.inc.php 5302 2012-09-05 12:28:13Z dehn

#### **Author**

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

#### **6.5.2** Member Function Documentation

# 6.5.2.1 \_getHashHexValue( \$ data, \$ hashFunction = 'sha1') [protected]

#### **Parameters**

string \$data string to be hashed

#### Returns

string the hash

The documentation for this class was generated from the following file:

• sofortLib\_classic\_notification.inc.php

# 6.6 SofortLib\_ConfirmSr Class Reference

# **Public Member Functions**

- \_\_construct (\$configKey=")
- setTransaction (\$transaction, \$invoice=0)
- setComment (\$comment)
- addItem (\$itemId, \$productNumber, \$productType, \$title, \$description, \$quantity, \$unitPrice, \$tax)
- removeItem (\$productId, \$quantity=0)

- updateCart (\$cartItems=array())
- cancelInvoice (\$transaction=")
- confirmInvoice (\$transaction=")
- getInvoiceUrl (\$i=0)

# 6.6.1 Detailed Description

```
This class is for confirming and changing statuses of invoices
```

```
eg: $confirmObj = new SofortLib_ConfirmSr('yourapikey');
```

\$confirmObj->confirmInvoice('1234-456-789654-31321')->sendRequest();

Copyright (c) 2012 SOFORT AG

#### Date:

```
2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)
```

#### Version

SofortLib 1.5.0

#### Id:

sofortLib\_confirm\_sr.inc.php 5301 2012-09-05 12:27:56Z dehn

# Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

# 6.6.2 Constructor & Destructor Documentation

```
6.6.2.1 __construct ( $ configKey = " )
```

create new confirm object

# **Parameters**

String \$apikey your API-key

# **6.6.3** Member Function Documentation

# 6.6.3.1 addItem ( \$ itemId, \$ productNumber, \$ productType, \$ title, \$ description, \$ quantity, \$ unitPrice, \$ tax )

add one item to the cart if you want to change the invoice just useable with api version 1.0

#### See also

SofortLib\_EditSr

#### **Parameters**

```
string $productNumber product number, EAN code, ISBN number or similar
string $title description of this title
double $unit_price gross price of one item
int $productType product type number see manual
string $description additional description of this item
int $quantity default 1
int $tax tax in percent, default 19
```

# **6.6.3.2** cancelInvoice (\$ transaction = ")

cancel the invoice just useable with api version 1.0

#### See also

```
SofortLib_EditSr
```

# **Parameters**

string \$transaction the transaction id

#### Returns

SofortLib\_ConfirmSr

# **6.6.3.3** confirmInvoice ( \$ transaction = ")

confirm the invoice

#### **Parameters**

string \$transaction the transaction id

#### Returns

SofortLib\_ConfirmSr

# 6.6.3.4 getInvoiceUrl ( \$ i = 0 )

after you you changed/confirmed an invoice you can download the new invoice-pdf with this function

# Returns

string url

```
6.6.3.5 removeItem ( $ productId, $ quantity = 0 )
TODO: implement removal of items
See also
    SofortLib_EditSr
Parameters
    $productId
    $quantity
6.6.3.6 setComment ( $ comment )
set a comment for refunds just useable with api version 1.0
See also
    SofortLib_EditSr
Parameters
    string $arg
6.6.3.7 setTransaction ( \$ transaction, \$ invoice = 0 )
Set the transaction you want to confirm
Parameters
    String $transaction Transaction Id
Returns
    SofortLib ConfirmSr
6.6.3.8 updateCart($ cartItems = array())
See also
    SofortLib_EditSr
Parameters
    array $cartItems
The documentation for this class was generated from the following file:
```

• sofortLib\_confirm\_sr.inc.php

# 6.7 SofortLib\_Debit Class Reference

# **Public Member Functions**

- sendRequest ()
- setProjectId (\$id)
- setDate (\$date=")
- setSenderAccount (\$bankCode, \$accountNumber, \$holder)
- setSenderAccountNumber (\$accountNumber)
- setSenderBankCode (\$bankCode)
- setSenderHolder (\$name)
- setAmount (\$amount)
- addUserVariable (\$userVariable)
- addReason (\$reason)
- setReason (\$reason1, \$reason2=")
- getTransactionId()

# 6.7.1 Detailed Description

class for handling debit/lastschrift

Copyright (c) 2012 SOFORT AG

# Date:

```
2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)
```

# Version

SofortLib 1.5.0

#### Id:

sofortLib\_debit.inc.php 5301 2012-09-05 12:27:56Z dehn

# Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

# **6.7.2** Member Function Documentation

# 6.7.2.1 addReason (\$ reason)

adds another reason (verwendungszweck) only first two can be used, 27 characters each

#### **Parameters**

*\$reason* string

# Returns

SofortLib\_Debit \$this

#### **6.7.2.2** addUserVariable (\$ userVariable)

add another user-variable to this transfer this variable could be a customer-number or similar and will help you identify this transfer later

# **Parameters**

String \$userVariable max 255 characters

#### **Returns**

SofortLib\_Debit \$this

# 6.7.2.3 getTransactionId()

get Transaction-Id of this Transfer

#### Returns

String transaction-id

# **6.7.2.4** sendRequest ( )

send this debitpay and get response

## Returns

boolean true if transaction was accepted, false otherwise

# 6.7.2.5 setAmount ( \$ amount )

set amount of this transfer needs to be a decimal e.g. 2.24

#### **Parameters**

*float* \$amount of this transfer

# 6.7.2.6 setDate ( \$ date = " )

sets date of this debitpay automatically called in constructor and set to today

#### **Parameters**

String \$date date in Format Y-m-d (eg: 2011-01-20), default: today

#### Returns

SofortLib\_Debit \$this

# **6.7.2.7** setProjectId (\$ *id* )

Project id Id of your Sofortlastschrift project

#### **Parameters**

int \$id project id

#### Returns

SofortLib\_Debit \$this

# 6.7.2.8 setReason ( \$ reason1, \$ reason2 = ")

set reason (verwendugszweck) of this transfer two lines possible, 27 characters each

#### **Parameters**

\$reason1

\$reason2

# Returns

SofortLib\_Debit \$this

# 6.7.2.9 setSenderAccount(\$ bankCode, \$ accountNumber, \$ holder)

set data of account

# **Parameters**

String \$bankCode bank code of bank

String \$accountNumber account number

String \$holder Name/Holder of this account

## Returns

SofortLib\_Debit \$this

# **6.7.2.10** setSenderAccountNumber ( \$ accountNumber )

set data of account

## **Parameters**

String \$accountNumber account number

#### **Returns**

SofortLib\_Debit \$this

#### 6.7.2.11 setSenderBankCode ( \$ bankCode )

set data of account

# **Parameters**

String \$bankCode bank code of bank

#### Returns

SofortLib\_Debit \$this

# 6.7.2.12 setSenderHolder (\$ name)

set data of account

#### **Parameters**

String \$name Name/Holder of this account

#### Returns

SofortLib\_Debit \$this

The documentation for this class was generated from the following file:

• sofortLib\_debit.inc.php

# 6.8 SofortLib\_EditSr Class Reference

# **Public Member Functions**

- \_\_construct (\$configKey=")
- setComment (\$comment, \$invoice=0)
- addItem (\$itemId, \$productNumber, \$productType, \$title, \$description, \$quantity, \$unitPrice, \$tax, \$invoice=0)
- getInvoiceUrl ()

# 6.8.1 Detailed Description

This class is for confirming and changing statuses of invoices

```
eg: $confirmObj = new SofortLib_EditSr('yourapikey');
```

\$confirmObj->confirmInvoice('1234-456-789654-31321')->sendRequest();

Copyright (c) 2012 SOFORT AG

#### Date:

```
2012-05-21 16:53:26 +0200 (Mo, 21 Mai 2012)
```

#### Version

```
SofortLib 1.5.0
```

#### Id:

```
sofortLib_edit_sr.inc.php 4191 2012-05-21 14:53:26Z niehoff
```

## Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

#### **6.8.2** Constructor & Destructor Documentation

```
6.8.2.1 \_construct ( $ configKey = " )
```

create new confirm object

#### **Parameters**

String \$apikey your API-key

# **6.8.3** Member Function Documentation

6.8.3.1 addItem ( \$ itemId, \$ productNumber, \$ productType, \$ title, \$ description, \$ quantity, \$ unitPrice, \$ tax, \$ invoice = 0 )

add one item to the cart if you want to change the invoice

## **Parameters**

```
string $itemId itemId
double $unit_price gross price of one item
int $quantity default 1
```

# 6.8.3.2 getInvoiceUrl()

after you you changed/confirmed an invoice you can download the new invoice-pdf with this function

#### Returns

string url

#### 6.8.3.3 setComment (\$ comment, \$ invoice = 0)

set a comment for refunds just useable with api version 1.0

#### **Parameters**

```
string $arg
```

The documentation for this class was generated from the following file:

• sofortLib\_edit\_sr.inc.php

# 6.9 SofortLib\_iDeal\_Banks Class Reference

# **6.9.1 Detailed Description**

This class encapsulates retrieval of listed banks of the Netherlands Copyright (c) 2012 SOFORT AG

# Date:

```
2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)
```

# Version

SofortLib 1.5.0

Id:

sofortLib\_ideal\_banks.inc.php 5301 2012-09-05 12:27:56Z dehn

# Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

The documentation for this class was generated from the following file:

• sofortLib\_ideal\_banks.inc.php

# 6.10 SofortLib\_iDealClassic Class Reference

Inherits SofortLib\_SofortueberweisungClassic.

## **Public Member Functions**

- setSenderCountryId (\$senderCountryId= 'NL')
- setSenderBankCode (\$senderBankCode)
- getRelatedBanks ()

# **6.10.1** Detailed Description

iDeal\_Classic extends Sofortueberweisung\_Classic, implementing payment via iDeal Setup a session within iDeal using the classic api You get the so called payment-url after successful configuration Payment is enabled with this url being sent to iDeal

eg: \$sofort = \$sofortLib\_iDealClassic = new SofortLib\_iDealClassic (\$configurationKey, \$password, \$hashfunction = 'sha1'); \$sofort->getRelatedBanks(); //get all iDEAL-Banks \$sofort->getPaymentUrl(); //returns paymentUrl including (including ...&hash=1234567890&...)

Copyright (c) 2012 SOFORT AG

#### Date:

```
2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)
```

#### Version

SofortLib 1.5.0

#### Id:

sofortLib\_ideal\_classic.php 5301 2012-09-05 12:27:56Z dehn

## Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

## **6.10.2** Member Function Documentation

# 6.10.2.1 getRelatedBanks ( )

Get related banks of iDeal

#### Returns

array

# 6.10.2.2 setSenderBankCode ( \$ senderBankCode )

Set sender's bank code

## **Parameters**

string \$senderBankCode

#### **Returns**

instance

#### 6.10.2.3 setSenderCountryId ( \$ senderCountryId = 'NL')

Set sender's country id

#### **Parameters**

string \$senderCountryId

#### Returns

instance

The documentation for this class was generated from the following file:

• sofortLib\_ideal\_classic.php

# 6.11 SofortLib\_Multipay Class Reference

# **Public Member Functions**

- \_\_construct (\$configKey=")
- setLanguageCode (\$arg)
- setTimeout (\$arg)
- setEmailCustomer (\$arg)
- setPhoneNumberCustomer (\$arg)
- addUserVariable (\$arg)
- setSenderAccount (\$bankCode, \$accountNumber, \$holder)
- setAmount (\$arg, \$currency= 'EUR')
- setReason (\$arg, \$arg2=")
- setSuccessUrl (\$arg)
- setAbortUrl (\$arg)
- setTimeoutUrl (\$arg)
- setNotificationUrl (\$arg)
- addNotificationUrl (\$arg)
- setNotificationEmail (\$arg)
- 1.1NI-4'C --4'--E---'1 (\$----'
- addNotificationEmail (\$arg)
- setVersion (\$arg)
- setSofortueberweisung (\$amount=")
- setSofortueberweisungCustomerprotection (\$customerProtection=true)
- setSofortlastschrift (\$amount=")
- setSofortlastschriftAddress (\$firstname, \$lastname, \$street, \$streetNumber, \$zipcode, \$city, \$salutation, \$country= 'DE')
- setLastschrift (\$amount=")
- setLastschriftAddress (\$firstname, \$lastname, \$street, \$streetNumber, \$zipcode, \$city, \$salutation, \$country= 'DE')
- setSofortDauerauftrag ()
- setSofortDauerauftragStartDate (\$arg)

- setSofortDauerauftragTotalPayments (\$arg)
- setSofortDauerauftragMinimumPayments (\$arg)
- setSofortDauerauftragInterval (\$arg)
- setSofortrechnung ()
- setSofortvorkasse (\$amount=")
- setSofortvorkasseCustomerprotection (\$customerProtection=true)
- setSofortrechnungCustomerId (\$arg)
- setSofortrechnungOrderId (\$arg)
- setDebitorVatNumber (\$vatNumber)
- setSofortrechnungInvoiceAddress (\$firstname, \$lastname, \$street, \$streetNumber, \$zipcode, \$city, \$salutation, \$country= 'DE', \$nameAdditive= ", \$streetAdditive= ", \$companyName= ")
- setSofortrechnungShippingAddress (\$firstname, \$lastname, \$street, \$street. Number, \$zipcode, \$city, \$salutation, \$country= 'DE', \$nameAdditive= ", \$streetAdditive= ", \$companyName= ")
- addSofortrechnungItem (\$itemId, \$productNumber, \$title, \$unitPrice, \$product-Type=0, \$description=", \$quantity=1, \$tax=19)
- removeSofortrechnungItem (\$itemId)
- updateSofortrechnungItem (\$itemId, \$quantity, \$unitPrice)
- sendValidationRequest ()
- getPaymentUrl ()
- getTransactionId ()

# 6.11.1 Detailed Description

Setup a multipay payment session after the configuration of multiple payment methods you will receive an url and a transaction id, your customer should be redirected to this url you can use the transaction id for future reference of this payment

example by usage: \$objMultipay = new SofortLib\_Multipays('my-API-KEY'); \$objMultipay->setSofortueberweisung(); //OR setSofortrechnung(), setSofortvorkasse() etc. \$objMultipay->set...(\$param); //set params for PNAG-API (watch API-documentation for needed params) \$objMultipay->add...(\$param); //add params for PNAG-API (watch API-documentation for needed params) \$errorsAnd-Warnings = \$objMultipay->validateRequest(); //send param against the PNAG-API without setting an order ... make own validation of \$errorsAndWarnings and if ok ... \$objMultipay->sendRequest(); //set the order at PNAG \$errorsAndWarnings = \$objMultipay->getErrors(); //should not occur, if validation was ok ... make own validation of \$errorsAndWarnings and if ok ... ... finish order in the shopsystem

Copyright (c) 2012 SOFORT AG

#### Date:

2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)

#### Version

SofortLib 1.5.0

Id:

sofortLib\_multipay.inc.php 5301 2012-09-05 12:27:56Z dehn

#### Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

# 6.11.2 Constructor & Destructor Documentation

```
6.11.2.1 __construct ( $ configKey = " )
```

create a new payment object

#### **Parameters**

```
string $apikey your API key
int $projectId your project id
```

# **6.11.3** Member Function Documentation

# 6.11.3.1 addNotificationEmail (\$ arg)

you can set set multiple emails for receiving notifications

# Parameters

string \$arg email

# Returns

SofortLib\_Multipay

# 6.11.3.2 addNotificationUrl(\$ arg)

you can set set multiple urls for receiving notifications this might be helpfull if you have several systems for processing an order (e.g. an ERP system)

#### **Parameters**

string \$arg url

#### **Returns**

SofortLib\_Multipay

```
6.11.3.3 addSofortrechnungItem ( $ itemId, $ productNumber, $ title, $ unitPrice, $ productType = 0, $ description = ", $ quantity = 1, $ tax = 19 )
```

add one item to the cart

#### **Parameters**

```
int $itemId unique item id
string $productNumber product number, EAN code, ISBN number or similar
string $title description of this title
double $unit_price gross price of one item
int $productType product type number see manual (0=other, 1=shipping, ...)
string $description additional description of this item
int $quantity default 1
int $tax tax in percent, default 19
```

# 6.11.3.4 addUserVariable (\$ arg)

add another variable this can be your internal order id or similar

# **Parameters**

string \$arg the contents of the variable

# Returns

SofortLib\_Multipay

# 6.11.3.5 getPaymentUrl()

after configuration and sending this request you can use this function to redirect the customer to the payment form

#### Returns

string url of payment form

# 6.11.3.6 getTransactionId()

use this id to track the transaction

#### Returns

string transaction id

# 6.11.3.7 removeSofortrechnungItem ( \$ itemId )

Remove one item from cart

#### **Parameters**

\$itemId

#### Returns

boolean

# 6.11.3.8 sendValidationRequest ( )

makes a request against the pnag-API and returns all API-Fault/Warnings it doesnt result in an order at pnag!

# Returns

bool

# 6.11.3.9 setAbortUrl(\$ arg)

the customer will be redirected to this url if he uses the abort link on the payment form, should redirect him back to his cart or to the payment selection page

# **Parameters**

string \$arg url for aborting the transaction

# Returns

SofortLib\_Multipay

# 6.11.3.10 setAmount ( \$ arg, \$ currency = 'EUR' )

amount of this payment

#### **Parameters**

```
double $arg
```

string \$currency currency of this transaction, default EUR

# Returns

SofortLib\_Multipay \$this

# **6.11.3.11** setDebitorVatNumber ( \$ vatNumber )

set debitor vat number for invoice

#### **Parameters**

string \$vatNumber

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.12 setEmailCustomer ( \$ arg )

set the email address of the customer this will be used for sofortvorkasse and sofortrechnung

#### **Parameters**

string \$arg email address

#### **Returns**

SofortLib\_Multipay

# 6.11.3.13 setLanguageCode (\$ arg)

the language code will help in determing what language to use when displaying the payment form, other data like browser settings and ip will be used as well

## **Parameters**

```
string $arg de|en|n1|fr ...
```

# Returns

SofortLib\_Multipay

# **6.11.3.14** setLastschrift ( \$ amount = " )

add lastschrift as payment method

#### **Parameters**

double \$amount this amount only applies to this payment method

#### Returns

SofortLib\_Multipay \$this

```
6.11.3.15 setLastschriftAddress ( $ firstname, $ lastname, $ street, $ streetNumber, $ zipcode, $ city, $ salutation, $ country = 'DE' )
```

set the address of the customer for address validation, this should be the invoice address of the customer

#### **Parameters**

```
string $firstname
string $lastname
string $street
string $streetNumber
string $zipcode
string $city
int $salutation [2|3] 2=Mr. 3=Mrs.
string $country country code, only DE allowed at the moment
```

#### Returns

SofortLib\_Multipay object

# 6.11.3.16 setNotificationEmail (\$ arg)

set the email address where you want notification about status changes being sent to.

#### **Parameters**

string \$arg email address

#### Returns

SofortLib\_Multipay

# 6.11.3.17 setNotificationUrl(\$ arg)

set the url where you want notification about status changes being sent to. Use SofortLib\_Notification and SofortLib\_TransactionData to further process that notification

# **Parameters**

string \$arg url

#### Returns

SofortLib\_Multipay

# 6.11.3.18 setPhoneNumberCustomer ( \$ arg )

set the phone number of the customer

#### **Parameters**

string \$arg phone number

# Returns

SofortLib\_Multipay

# 6.11.3.19 setReason ( \$ arg, \$ arg2 = ")

set the reason values of this transfer

#### **Parameters**

```
string $arg max 27 charactersstring $arg2 max 27 characters
```

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.20 setSenderAccount ( \$ bankCode, \$ accountNumber, \$ holder )

set data of account

#### **Parameters**

```
string $bank_code bank code of bankstring $account_number account numberstring $holder Name/Holder of this account
```

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.21 setSofortDauerauftrag ( )

add SofortDauerauftrag as a payment method

# Returns

SofortLib\_Multipay object

# 6.11.3.22 setSofortDauerauftragInterval ( \$ arg ) set the interval for payment **Parameters** \$arg Returns object 6.11.3.23 setSofortDauerauftragMinimumPayments ( \$ arg ) set the minimum amount for Dauerauftrag **Parameters** \$arg Returns object $6.11.3.24 \quad setSofortDauerauftragStartDate \left( \ \$ \ arg \ \right)$ Set the date to start "Dauerauftrag" date must be compliant to ISO 8601 (e.g. 2011-06-21) **Parameters** \$arg - string date 6.11.3.25 setSofortDauerauftragTotalPayments (\$ arg ) Enter description here ... **Parameters** \$arg Returns object

# 6.11.3.26 setSofortlastschrift ( \$ amount = ")

add sofortlastschrift as payment method

#### **Parameters**

double \$amount this amount only applies to this payment method

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.27 setSofortlastschriftAddress ( \$ firstname, \$ lastname, \$ street, \$ streetNumber, \$ zipcode, \$ city, \$ salutation, \$ country = 'DE' )

set the address of the customer for address validation, this should be the invoice address of the customer

#### **Parameters**

```
string $firstname
string $lastname
string $street
string $streetNumber
string $zipcode
string $city
int $salutation [2|3] 2=Mr. 3=Mrs.
string $country country code, only DE allowed at the moment
```

# Returns

SofortLib\_Multipay \$this

# 6.11.3.28 setSofortrechnung ( )

add sofortrechnung as payment method if you use this payment method you have to provide the customer address and cart as well the total amount of this payment method will be determined by the total of the cart

#### **Returns**

SofortLib\_Multipay object

## **6.11.3.29** setSofortrechnungCustomerId (\$ arg )

set the customer id which will appear on top of the invoice

# **Parameters**

int \$arg

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.30 setSofortrechnungInvoiceAddress (\$ firstname, \$ lastname, \$ street, \$ streetNumber, \$ zipcode, \$ city, \$ salutation, \$ country = 'DE', \$ nameAdditive = ", \$ streetAdditive = ", \$ companyName = ")

set the invoice address of the customer

#### **Parameters**

```
string $firstname
string $lastname
string $street
string $streetNumber
string $zipcode
string $city
int $salutation [2|3] 2 = Mr. 3 = Mrs.
string $country country code, only DE allowed at the moment
```

# Returns

SofortLib\_Multipay \$this

# 6.11.3.31 setSofortrechnungOrderId ( \$ arg )

set the order id which will appear on top of the invoice

#### **Parameters**

int \$arg

# Returns

SofortLib\_Multipay \$this

```
6.11.3.32 setSofortrechnungShippingAddress ( $ firstname, $ lastname, $ street, $ streetNumber, $ zipcode, $ city, $ salutation, $ country = 'DE', $ nameAdditive = ", $ streetAdditive = ", $ companyName = ")
```

set the shipping address of the customer

#### **Parameters**

```
string $firstname
string $lastname
string $street
string $streetNumber
string $zipcode
string $city
int $salutation [2|3] 1 = Mr. 2 = Mrs.
string $country country code, only DE allowed at the moment
```

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.33 setSofortueberweisung ( \$ amount = ")

add sofortueberweisung as payment method

### **Parameters**

double \$amount this amount only applies to this payment method

# Returns

SofortLib\_Multipay \$this

# 6.11.3.34 setSofortueberweisungCustomerprotection ( \$ customerProtection = true )

add sofortueberweisung as payment method adds customer protection

# **Parameters**

double \$amount this amount only applies to this payment method

# Returns

SofortLib\_Multipay \$this

# **6.11.3.35** setSofortvorkasse ( \$ *amount* = " )

add sofortvorkasse as payment method

#### **Parameters**

double \$amount this amount only applies to this payment method

#### Returns

SofortLib\_Multipay objet

# 6.11.3.36 setSofortvorkasseCustomerprotection ( \$ customerProtection = true )

add sofortvorkasse as payment method adds customer protection

#### **Parameters**

double \$amount this amount only applies to this payment method

#### Returns

SofortLib\_Multipay \$this

# 6.11.3.37 setSuccessUrl(\$ arg)

the customer will be redirected to this url after a successful transaction, this should be a page where a short confirmation is displayed

# **Parameters**

string \$arg the url after a successful transaction

# Returns

SofortLib\_Multipay

# **6.11.3.38** setTimeout ( \$ arg )

timeout how long this transaction configuration will be valid for this is the time between the generation of the payment url and the user completing the form, should be at least two to three minutes defaults to unlimited if not set

# Parameters

int \$arg timeout in seconds

#### Returns

SofortLib\_Multipay

# 6.11.3.39 setTimeoutUrl(\$ arg)

if the customer takes too much time or if your timeout is set too short he will be redirected to this page

#### **Parameters**

string \$arg url

#### Returns

SofortLib\_Multipay

# **6.11.3.40** setVersion (\$ arg )

set the version of this payment module this is helpfull so the support staff can easily find out if someone uses an outdated module

#### **Parameters**

string \$arg version string of your module

# Returns

SofortLib\_Multipay

# 6.11.3.41 updateSofortrechnungItem (\$ itemId, \$ quantity, \$ unitPrice )

Update one item in cart

### **Parameters**

\$itemId
\$quantity
\$unit\_price

## Returns

boolean

The documentation for this class was generated from the following file:

• sofortLib\_multipay.inc.php

# 6.12 SofortLib\_Notification Class Reference

# **Public Member Functions**

- \_\_construct ()
- getNotification (\$source= 'php://input')

# **6.12.1** Detailed Description

Instance of this class handles the callback of Payment Network to notify about a status change

In rare cases notifications might be doubled or even wrong alltogether (if send by a malicious user). So don't use this to change your status but instead use the transaction id to query the webservice for detailed data (SofortLib\_TransactionData)

```
eg: $notificationObj = new SofortLib_Notification();
```

\$transactionId = \$notificationObj->getNotification();

Copyright (c) 2012 SOFORT AG

#### Date:

2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)

#### Version

SofortLib 1.5.0

#### Id:

sofortLib\_notification.inc.php 5301 2012-09-05 12:27:56Z dehn

# **Author**

SOFORT AG http://www.sofort.com(integration@sofort.com)

# **6.12.2** Constructor & Destructor Documentation

```
6.12.2.1 __construct( )
```

creates a new notification object for receiving notifications

# **6.12.3** Member Function Documentation

```
6.12.3.1 getNotification ( $ source = 'php://input')
```

reads the input and tries to read the transaction id

#### Returns

array transactionid=>status

The documentation for this class was generated from the following file:

• sofortLib\_notification.inc.php

# 6.13 SofortLib\_Refund Class Reference

# **Public Member Functions**

- sendRequest ()
- addRefund (\$transaction, \$amount, \$comment=")
- setSenderAccount (\$bankCode, \$accountNumber, \$holder=")

# **6.13.1 Detailed Description**

class for refund/rueckbuchung

Copyright (c) 2012 SOFORT AG

## Date:

2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)

#### Version

SofortLib 1.5.0

#### Id:

sofortLib\_refund.inc.php 5301 2012-09-05 12:27:56Z dehn

# Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

# **6.13.2** Member Function Documentation

# 6.13.2.1 addRefund ( \$ transaction, \$ amount, \$ comment = " )

add a new refund to this message

#### **Parameters**

string \$transaction transaction id of transfer you want to refund

**float** \$amount amount of money to refund, less or equal to amount of original transfer

string \$comment comment that will be displayed in admin-menu later

## Returns

SofortLib\_Refund \$this

## **6.13.2.2** sendRequest ( )

send this message and get response

#### Returns

array transactionid=>status

# 6.13.2.3 setSenderAccount(\$ bankCode, \$ accountNumber, \$ holder = ")

set data of account

#### **Parameters**

```
string $bank_code bank code of bankstring $account_number account numberstring $holder Name/Holder of this account
```

#### Returns

SofortLib\_Multipay \$this

The documentation for this class was generated from the following file:

• sofortLib\_refund.inc.php

# 6.14 SofortLib\_SofortueberweisungClassic Class Reference

Inherited by SofortLib\_iDealClassic.

# **Public Member Functions**

- setReason (\$arg, \$arg2=")
- setSuccessUrl (\$arg)
- setAbortUrl (\$arg)
- setNotificationUrl (\$arg)
- getHashHexValue (\$data, \$hashFunction= 'sha1')
- getSupportedHashAlgorithm ()
- getSupportedHashAlgorithms ()

# **Static Public Member Functions**

• static generatePassword (\$length=24)

# **6.14.1** Detailed Description

Setup a sofortueberweisung.de session using the classic api after the configuration of the configuration you will receive an url and a transaction id, your customer should be redirected to this url

Called by the sofortLib.php/sofortLib\_ideal\_classic.php etc. \$sofort->new SofortLib\_SofortueberweisungClassic( \$userid, \$projectid, \$password [, \$hashfunction='sha1']); \$sofort->set...(); //set params for Hashcalculation \$sofort->set...(); //set more params for Hashcalculation \$sofort->getPaymentUrl(); Notice: sometimes getPaymentUrl() must be overwritten by calling class because of changed hash-params

Copyright (c) 2012 SOFORT AG

#### Date:

```
2012-09-05 14:27:56 +0200 (Wed, 05 Sep 2012)
```

#### Version

SofortLib 1.5.0

Id:

sofortLib\_sofortueberweisung\_classic.php 5301 2012-09-05 12:27:56Z dehn

# **Author**

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

# **6.14.2** Member Function Documentation

# 6.14.2.1 static generatePassword ( \$ length = 24 ) [static]

## **Parameters**

int [optional] \$length length of return value, default 24

# Returns

string

# 6.14.2.2 getHashHexValue ( \$ data, \$ hashFunction = 'sha1')

#### **Parameters**

string \$data string to be hashed

#### Returns

string the hash

# **6.14.2.3** getSupportedHashAlgorithm ( )

checks wich hash algorithms are supported by the server and returns the best one

# Returns

sha512|sha256|sha1|md5|empty string

#### **6.14.2.4** getSupportedHashAlgorithms ( )

checks wich hash algorithms are supported by the server

#### Returns

array with all supported algorithms, preferred as first one (index 0)

# 6.14.2.5 setAbortUrl ( \$ arg )

the customer will be redirected to this url if he uses the abort link on the payment form, should redirect him back to his cart or to the payment selection page

#### **Parameters**

string \$arg url for aborting the transaction

#### Returns

SofortLib\_Multipay

# 6.14.2.6 setNotificationUrl (\$ arg)

set the url where you want notification about status changes being sent to. Use SofortLib\_Notification and SofortLib\_TransactionData to further process that notification

# **Parameters**

string \$arg url

# Returns

SofortLib\_Multipay

```
6.14.2.7 setReason ( $ arg, $ arg2 = " )
```

Set the reason (Verwendungszweck) for sending money

#### **Parameters**

```
string $arg
string $arg2
```

# 6.14.2.8 setSuccessUrl (\$ arg)

the customer will be redirected to this url after a successful transaction, this should be a page where a short confirmation is displayed

#### **Parameters**

string \$arg the url after a successful transaction

#### Returns

```
SofortLib_Multipay
```

The documentation for this class was generated from the following file:

• sofortLib\_sofortueberweisung\_classic.php

# 6.15 SofortLib\_TransactionData Class Reference

# **Public Member Functions**

- setTransaction (\$arg)
- addTransaction (\$arg)
- setTime (\$from, \$to)
- setNumber (\$number, \$page= '1')
- getConsumerProtection (\$i=0)
- getInvoiceAddress (\$i=0)
- getShippingAddress (\$i=0)
- getStatus (\$i=0)
- getStatusReason (\$i=0)
- getStatusModifiedTime (\$i=0)
- getLanguageCode (\$i=0)
- getAmount (\$i=0)
- getAmountRefunded (\$i=0)
- getCurrency (\$i=0)
- getPaymentMethod (\$i=0)
- getSofortdauerauftragPayments (\$i=0)

- getSofortdauerauftragStartDate (\$i=0)
- getSofortdauerauftragInterval (\$i=0)
- getSofortdauerauftragTotalPayments (\$i=0)
- getSofortdauerauftragPaymentsSum (\$i=0)
- getSofortdauerauftragNumberOfPaymentsPending (\$i=0)
- getSofortdauerauftragNumberOfPaymentsReceived (\$i=0)
- getSofortdauerauftragMinimumPayments (\$i=0)
- getSofortdauerauftragPaymentsReceived (\$i=0)
- getSofortdauerauftragPaymentCount (\$i=0)
- getSofortdauerauftragPaymentReceived (\$i=0, \$i=0)
- getSofortdauerauftragAllPaymentsReceived (\$i=0)
- getSofortdauerauftragLastPaymentReceived (\$i=0)
- getTransaction (\$i=0)
- getItems (\$i=0)
- getReason (\$i=0)
- getUserVariable (\$n, \$i=0)
- getTime (\$i=0)
- getProjectId (\$i=0)
- getInvoiceUrl (\$i=0)
- getInvoiceStatus (\$i=0)
- getInvoiceObjection (\$i=0)
- isTest (\$i=0)
- isSofortueberweisung (\$i=0)
- isSofortvorkasse (\$i=0)
- isSofortlastschrift (\$i=0)
- isLastschrift (\$i=0)
- isSofortdauerauftrag (\$i=0)
- isSofortrechnung (\$i=0)
- isReceived (\$i=0)
- isLoss (\$i=0)
- isPending (\$i=0)
- isRefunded (\$i=0)
- getRecipientHolder (\$i=0)
- getRecipientAccountNumber (\$i=0)
- getRecipientBankCode (\$i=0)
- getRecipientCountryCode (\$i=0)
- getRecipientBankName (\$i=0)
- getRecipientBic (\$i=0)
- getRecipientIban (\$i=0)
- getSenderHolder (\$i=0)
- getSenderAccountNumber (\$i=0)
- getSenderBankCode (\$i=0)
- getSenderCountryCode (\$i=0)
- getSenderBankName (\$i=0)
- getSenderBic (\$i=0)
- getSenderIban (\$i=0)

- getInvoiceReason (\$n=0, \$i=0)
- getInvoiceDebitorText (\$i=0)
- getInvoiceDate (\$i=0)
- getInvoiceDueDate (\$i=0)
- getInvoiceNumber (\$i=0)
- getInvoiceBankHolder (\$i=0)
- getInvoiceBankAccountNumber (\$i=0)
- getInvoiceBankCode (\$i=0)
- getInvoiceBankName (\$i=0)
- getInvoiceType (\$i=0)

# **6.15.1** Detailed Description

This class is for retrieving information about transactions, you can search by transaction-id or by date

```
eg: $transactionDataObj = new SofortLib_TransactionData('yourapikey');
```

\$transactionDataObj->setTransaction('1234-456-789654-31321')->sendRequest();

echo \$transactionDataObj->getStatus();

Copyright (c) 2012 SOFORT AG

#### Date:

```
2012-09-10 10:26:54 +0200 (Mon, 10 Sep 2012)
```

# Version

SofortLib 1.3.0

### Id:

sofortLib\_transaction\_data.inc.php 5341 2012-09-10 08:26:54Z Niehoff

# Author

```
SOFORT AG http://www.sofort.com(integration@sofort.com)
```

# **6.15.2** Member Function Documentation

# 6.15.2.1 addTransaction (\$ arg)

use this function if you want to request detailed information about several transactions at once

#### **Parameters**

String \$arg

# Returns

SofortLib\_TransactionData \$this

# **6.15.2.2 getAmount** ( \$ *i* = 0 )

returns the total amount of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

double amount

# 6.15.2.3 getAmountRefunded ( i = 0 )

refund, if a transaction was refundend. amount = amountRefunded if everything was refunded

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

double amount

# **6.15.2.4** getConsumerProtection ( \$ i = 0 )

returns the state of consumer\_protection if set

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

boolean

# 

returns the currency of a transaction

## Parameters

int \$i if you request multiple transactions at once you can set the number here

## Returns

string EUR|USD|GBP....

# 6.15.2.6 getInvoiceAddress ( i = 0 )

returns the InvoiceAddress

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

array

# 6.15.2.7 getInvoiceBankAccountNumber (\$ i = 0)

invoice receiving bank account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string

# 6.15.2.8 getInvoiceBankCode ( i = 0 )

invoice receiving bank account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string

# 6.15.2.9 getInvoiceBankHolder ( i = 0 )

invoice receiving bank account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string

# 6.15.2.10 getInvoiceBankName (\$ i = 0)

invoice receiving bank account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string

# 6.15.2.11 getInvoiceDate ( i = 0 )

date of the invoice

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string

# 6.15.2.12 getInvoiceDebitorText ( i = 0 )

get debitor text (Forderungsabtretung)

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string

# **6.15.2.13 getInvoiceDueDate** ( \$ i = 0 )

due date of the invoice, only available for confirmed invoices

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string

# **6.15.2.14** getInvoiceNumber ( \$ i = 0 )

invoice number

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string

# 6.15.2.15 getInvoiceObjection ( \$ i = 0 )

returns the status of an invoice

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string the status can be pending|received|reminder\_1|reminder\_2|reminder\_-3|encashment

# **6.15.2.16** getInvoiceReason ( n = 0, i = 0 )

returns the reason the customer needs to use when paying for "Rechnung by sofort"

## **Parameters**

int \$n specify reason linenumber, can be 1 or 2; us 0 for an array with all reasonsint \$i if you request multiple transactions at once you can set the number here

#### Returns

string array reason

# **6.15.2.17** getInvoiceStatus ( \$ i = 0 )

returns the status of an invoice

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string the status can be pending|received|reminder\_1|reminder\_2|reminder\_-3|encashment

# 6.15.2.18 getInvoiceType ( \$ i = 0 )

get the invoice type

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string (OR or LS)

# 6.15.2.19 getInvoiceUrl ( \$ i = 0 )

you can request the url to the pdf of a sr-invoice with this function

# Parameters

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string url to the pdf

# **6.15.2.20** getItems ( \$ i = 0 )

Returns an array containing all items of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here @ return array transactions items

# **6.15.2.21** getLanguageCode ( \$ i = 0 )

returns the language code of a transaction

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string language\_code

# **6.15.2.22** getPaymentMethod ( \$ i = 0 )

returns the payment method of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string su|sr|sl|sv|ls

# **6.15.2.23** getProjectId ( i = 0 )

returns the project id of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### **Returns**

int project id

# 6.15.2.24 getReason ( i = 0 )

Returns an array containing reason of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here @ return array transaction reason

# 6.15.2.25 getRecipientAccountNumber ( i = 0 )

returns the account number of the receiving account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### **Returns**

# 6.15.2.26 getRecipientBankCode ( i = 0 )

returns the bank code of the receiving account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string value

# 6.15.2.27 getRecipientBankName ( i = 0 )

returns the bank name of the receiving account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string value

# **6.15.2.28** getRecipientBic ( \$ i = 0 )

returns the BIC of the receiving account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string value

# **6.15.2.29** getRecipientCountryCode (\$i = 0)

returns the country code of the receiving account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

# 6.15.2.30 getRecipientHolder ( \$ i = 0 )

returns the holder of the receiving account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string value

# **6.15.2.31** getRecipientIban (\$ i = 0)

returns the IBAN of the receiving account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### **Returns**

string value

# 6.15.2.32 getSenderAccountNumber ( \$ i = 0 )

returns the account number of the sending account

## **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string value

# 6.15.2.33 getSenderBankCode ( i = 0 )

returns the bank code of the sending account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

# 6.15.2.34 getSenderBankName (\$ i = 0)

returns the bank name of the sending account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string value

# 6.15.2.35 getSenderBic ( i = 0 )

returns the BIC of the sending account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string value

# **6.15.2.36** getSenderCountryCode (\$i = 0)

returns the country code of the sending account

## **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string value

# **6.15.2.37** getSenderHolder ( \$ i = 0 )

returns the holder of the sending account

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

# 6.15.2.38 getSenderIban ( i = 0 )

returns the IBAN of the sending account

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string value

# 6.15.2.39 getShippingAddress (\$ i = 0)

returns the ShippingAddress

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

array

# 6.15.2.40 getSofortdauerauftragAllPaymentsReceived (\$i = 0)

Only payments made in the past (or the current day) are respected ...

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean

# 6.15.2.41 getSofortdauerauftragInterval ( i = 0 )

returns the interval for an abo payment

## **Parameters**

 $\emph{int}$  \$i if you request multiple transactions at once you can set the number here

## Returns

int

# 6.15.2.42 getSofortdauerauftragLastPaymentReceived (\$ i = 0)

Get the status (received true false) of the last payment of the currenct transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean

# 6.15.2.43 getSofortdauerauftragMinimumPayments (\$ i = 0)

returns the count of minimum payments for an abo payment

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

int

# 6.15.2.44 getSofortdauerauftragNumberOfPaymentsPending (\$i = 0)

returns the number of received payments of an abo payment

## **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

int

# 6.15.2.45 getSofortdauerauftragNumberOfPaymentsReceived (\$i = 0)

returns total sum of received payments of an abo payment

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

int

# 6.15.2.46 getSofortdauerauftragPaymentCount (\$ i = 0)

Enter description here ...

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# 6.15.2.47 getSofortdauerauftragPaymentReceived (\$ i = 0,\$ j = 0)

Get the \$j'th payment of \$i'th transaction

# **Parameters**

int \$i if you request multiple transactions at once you can set the number hereint \$j int current payment

## Returns

boolean

# 6.15.2.48 getSofortdauerauftragPayments (\$ i = 0)

returns the payments array for an abo payment

## **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

array

# 6.15.2.49 getSofortdauerauftragPaymentsReceived ( i = 0 )

returns the count of received payments for an abo payment

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

int

# 6.15.2.50 getSofortdauerauftragPaymentsSum ( i = 0 )

returns total sum of received payments of an abo payment

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

int

# 6.15.2.51 getSofortdauerauftragStartDate (\$ i = 0)

returns the start date for an abo payment

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

array

# 6.15.2.52 getSofortdauerauftragTotalPayments (\$ i = 0)

returns total of payments of an abo payment

## **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

int

# 6.15.2.53 getStatus ( i = 0 )

returns the status of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

string pending|received|loss|refunded

# 6.15.2.54 getStatusModifiedTime ( \$ i = 0 )

returns the time of the last status-change so you can check if sth. changed

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string time e.g. 2011-01-01T12:35:09+01:00 use strtotime() to convert it to unix-time

# 6.15.2.55 getStatusReason (\$ i = 0)

returns the detailed status description of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string message

# 6.15.2.56 getTime ( \$ i = 0 )

returns the time of a transaction

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

string time e.g. 2011-01-01T12:35:09+01:00 use strtotime() to convert it to unix-time

# **6.15.2.57 getTransaction** ( \$ i = 0 )

returns the transaction id of a transaction

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

string transaction id

# 6.15.2.58 getUserVariable (\$ n, \$ i = 0)

returns the user variable of a transaction

#### **Parameters**

int \$n number of the variable

int \$i if you request multiple transactions at once you can set the number here

#### Returns

string the content of this variable

# 6.15.2.59 isLastschrift (\$ i = 0)

check if the transaction was a lastschrift by sofort

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean true false

# 6.15.2.60 isLoss (\$ i = 0)

check if status of transaction is loss

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean true|false

# 6.15.2.61 is Pending ( i = 0 )

check if status of transaction is pending

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean true false

# 6.15.2.62 is Received ( \$ i = 0 )

check if status of transaction is received

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

boolean true false

# 6.15.2.63 is Refunded (\$ i = 0)

check if status of transaction is refunded

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

boolean true false

# 6.15.2.64 is Sofort dauerauftrag (\$ i = 0)

check if the transaction was a sofortdauerauftrag

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

# Returns

boolean true|false

# 6.15.2.65 is Sofortlast schrift (\$ i = 0)

check if the transaction was a sofortlastschrift

## **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

boolean true false

# 6.15.2.66 is Sofortrechnung ( \$ i = 0 )

check if the transaction was a sofortrechnung

# **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean true false

# 6.15.2.67 is Sofortueberweisung (\$ i = 0)

check if the transaction was a sofortueberweisung

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean true false

# **6.15.2.68** is Sofortvorkasse ( \$ i = 0 )

check if the transaction was a sofortvorkasse

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

#### Returns

boolean true false

# 6.15.2.69 is Test ( i = 0 )

checks if the transaction was a test

#### **Parameters**

int \$i if you request multiple transactions at once you can set the number here

## Returns

bool true false

```
6.15.2.70 setNumber (  number,  page = '1' )
```

you can limit the number of results

#### **Parameters**

```
int $number number of results [0-100]int $page result page
```

## Returns

SofortLib\_TransactionData \$this

#### See also

setTime()

# **6.15.2.71** setTime (\$ from, \$ to )

you can request all transactions of a certain time period use setNumber() to limit the results

## **Parameters**

```
string $from date possible formats: 2011-01-25 or 2011-01-25T19:01:02+02:00 string $to date possible formats: 2011-01-25 or 2011-01-25T19:01:02+02:00
```

#### Returns

SofortLib\_TransactionData \$this

# See also

setNumber()

# 6.15.2.72 setTransaction (\$ arg)

use this function if you want to request detailed information about a single transaction

# **Parameters**

String \$arg

#### **Returns**

SofortLib\_TransactionData \$this

The documentation for this class was generated from the following file:

• sofortLib\_transaction\_data.inc.php

# **Index**

construct	cancelInvoice
PnagInvoice, 15	PnagInvoice, 17
SofortLib, 31	SofortLib_CancelSr, 38
SofortLib_CancelSa, 36	SofortLib_ConfirmSr, 42
SofortLib_CancelSr, 38	checkout
SofortLib_ConfirmSr, 41	PnagInvoice, 18
SofortLib_EditSr, 48	confirmInvoice
SofortLib_Multipay, 53	PnagInvoice, 18
SofortLib_Notification, 65	SofortLib_ConfirmSr, 42
_getHashHexValue	
SofortLib_ClassicNotification, 40	deleteAllErrors
_sendMessage	SofortLib, 32
SofortLib, 32	deleteAllWarnings
	SofortLib, 32
addInvoiceAddress	disableLog
PnagInvoice, 15	PnagInvoice, 18
addItem	SofortLib, 32
SofortLib_ConfirmSr, 41	
SofortLib_EditSr, 48	enableLog
addItemToInvoice	PnagInvoice, 18
PnagInvoice, 16	SofortLib, 32
addNotificationEmail	error
SofortLib_Multipay, 53	SofortLib, 32
addNotificationUrl	fatalError
SofortLib_Multipay, 53	
addReason	SofortLib, 32
SofortLib_Debit, 44	generatePassword
addRefund	SofortLib
SofortLib_Refund, 66	SofortueberweisungClassic
addShippingAddress	68
PnagInvoice, 16	getAmount
addShippingAddresss	PnagInvoice, 18
PnagInvoice, 17	SofortLib_TransactionData, 72
addSofortrechnungItem	getAmountRefunded
SofortLib_Multipay, 53	SofortLib_TransactionData, 73
addTransaction	getConstantById
SofortLib_TransactionData, 72	PnagInvoice, 19
addUserVariable	getConstantByName
PnagInvoice, 17	PnagInvoice, 19
SofortLib_Debit, 44	getConsumerProtection
SofortLib_Multipay, 54	SofortLib_TransactionData, 73

getCurrency	SofortLib_EditSr, 48
SofortLib_TransactionData, 73	SofortLib_TransactionData, 77
getError	getItems
PnagInvoice, 19	PnagInvoice, 20
SofortLib, 32	SofortLib_TransactionData, 77
getErrorCodes	getLanguageCode
PnagInvoice, 19	PnagInvoice, 21
getErrors	SofortLib_TransactionData, 77
PnagInvoice, 19	getNotification
SofortLib, 33	SofortLib_Notification, 65
getHashHexValue	getPaymentMethod
SofortLib	PnagInvoice, 21
SofortueberweisungClassic,	SofortLib_TransactionData, 77
68	getPaymentUrl
getInvoice	PnagInvoice, 21
PnagInvoice, 19	SofortLib_Multipay, 54
getInvoiceAddress	getProjectId
SofortLib_TransactionData, 73	SofortLib_TransactionData, 78
getInvoiceBankAccountNumber	getReason
SofortLib_TransactionData, 74	SofortLib_TransactionData, 78
getInvoiceBankCode	getRecipientAccountNumber
SofortLib_TransactionData, 74	SofortLib_TransactionData, 78
getInvoiceBankHolder	getRecipientBankCode
SofortLib_TransactionData, 74	SofortLib_TransactionData, 78
getInvoiceBankName	getRecipientBankName
SofortLib_TransactionData, 74	SofortLib_TransactionData, 79
getInvoiceDate	getRecipientBic
SofortLib_TransactionData, 75	SofortLib_TransactionData, 79
getInvoiceDebitorText	getRecipientCountryCode SofortLib TransactionDate 70
SofortLib_TransactionData, 75	SofortLib_TransactionData, 79
getInvoiceDueDate	getRecipientHolder
SofortLib_TransactionData, 75	SofortLib_TransactionData, 79
getInvoiceNumber	getRecipientIban
SofortLib_TransactionData, 75	SofortLib_TransactionData, 80
getInvoiceObjection	getRelatedBanks
PnagInvoice, 20	SofortLib_iDealClassic, 50
SofortLib_TransactionData, 76	getSenderAccountNumber
getInvoiceReason	SofortLib_TransactionData, 80
SofortLib_TransactionData, 76	getSenderBankCode
getInvoiceStatus	SofortLib_TransactionData, 80
PnagInvoice, 20	getSenderBankName
SofortLib_TransactionData, 76	SofortLib_TransactionData, 80
getInvoiceTye	getSenderBic
PnagInvoice, 20	SofortLib_TransactionData, 81
getInvoiceType	getSenderCountryCode
SofortLib_TransactionData, 76	SofortLib_TransactionData, 81
getInvoiceUrl	getSenderHolder
PnagInvoice, 20	SofortLib_TransactionData, 81
SofortLib_CancelSr, 39	getSenderIban
SofortLib ConfirmSr 42	SofortLib TransactionData 81

getShippingAddress	getTransaction
SofortLib_TransactionData, 82	SofortLib_TransactionData, 86
getSofortdauerauftragAllPaymentsReceived	
SofortLib_TransactionData, 82	PnagInvoice, 22
getSofortdauerauftragInterval	getTransactionId
SofortLib_TransactionData, 82	PnagInvoice, 22
getSofortdauerauftragLastPaymentReceived	•
SofortLib_TransactionData, 82	SofortLib_Multipay, 54
getSofortdauerauftragMinimumPayments	getUserVariable
SofortLib_TransactionData, 83	SofortLib_TransactionData, 86
get So fort dauer auf trag Number Of Payments	egetiWgrnings
SofortLib_TransactionData, 83	PnagInvoice, 22
get So fort dauer auf trag Number Of Payments Research to the following property of the prop	Receiv@dfortLib, 33
SofortLib_TransactionData, 83	
getSofortdauerauftragPaymentCount	handleErrors
SofortLib_TransactionData, 83	SofortLib, 33
getSofortdauerauftragPaymentReceived	
SofortLib_TransactionData, 84	isError
get So fort dauer auftrag Payments	PnagInvoice, 22
SofortLib_TransactionData, 84	SofortLib, 33
getSofortdauerauftragPaymentsReceived	isLastschrift
SofortLib_TransactionData, 84	SofortLib_TransactionData, 87
getSofortdauerauftragPaymentsSum	isLoss
SofortLib_TransactionData, 84	SofortLib_TransactionData, 87
getSofortdauerauftragStartDate	isPending
SofortLib_TransactionData, 85	SofortLib_TransactionData, 87
getSofortdauerauftragTotalPayments	isReceived
SofortLib_TransactionData, 85	SofortLib_TransactionData, 87
getState	isRefunded
PnagInvoice, 21	SofortLib_TransactionData, 88
getStatus PnagInvoice, 21	isSofortdauerauftrag SofortLib_TransactionData, 88
SofortLib_TransactionData, 85	isSofortlastschrift
getStatusModifiedTime	SofortLib_TransactionData, 88
SofortLib_TransactionData, 85	isSofortrechnung
getStatusOfInvoice	SofortLib_TransactionData, 88
PnagInvoice, 21	isSofortueberweisung
getStatusReason	SofortLib_TransactionData, 89
PnagInvoice, 22	isSofortvorkasse
SofortLib_TransactionData, 86	SofortLib_TransactionData, 89
getSupportedHashAlgorithm	isTest
SofortLib	SofortLib_TransactionData, 89
SofortueberweisungClassic,	isWarning
68	PnagInvoice, 23
getSupportedHashAlgorithms	SofortLib, 34
SofortLib	
SofortueberweisungClassic,	log
69	PnagInvoice, 23
getTime	SofortLib, 34
SofortLib_TransactionData, 86	logError

PnagInvoice, 23	setCustomerId, 25
SofortLib, 34	setDebitorVatNumber, 25
logWarning	setEmailCustomer, 25
PnagInvoice, 23	setInvoiceObjection, 25
SofortLib, 34	setInvoiceStatus, 25
	setInvoiceUrl, 26
PnagInvoice, 13	setItems, 26
construct, 15	setLanguageCode, 26
addInvoiceAddress, 15	setNotificationUrl, 26
addItemToInvoice, 16	setOrderId, 26
addShippingAddress, 16	setPaymentMethod, 27
addShippingAddresss, 17	setPhoneNumberCustomer, 27
addUserVariable, 17	setReason, 27
cancelInvoice, 17	setSofortrechnung, 27
checkout, 18	setState, 27
confirmInvoice, 18	setStatus, 28
disableLog, 18	setStatusOfInvoice, 28
enableLog, 18	setStatusReason, 28
getAmount, 18	setSuccessUrl, 28
getConstantById, 19	setTime, 29
getConstantByName, 19	setTimeout, 29
getError, 19	setTimeoutUrl, 29
getErrorCodes, 19	setTransaction, 29
getErrors, 19	setTransactionId, 29
getInvoice, 19	setVersion, 30
getInvoiceObjection, 20	updateInvoice, 30
getInvoiceStatus, 20	updateOrderNumber, 30
getInvoiceTye, 20	updateOrder(variiber, 30
getInvoiceUrl, 20	refreshTransactionData
getItems, 20	PnagInvoice, 24
getLanguageCode, 21	removeItem
getPaymentMethod, 21	SofortLib_ConfirmSr, 42
getPaymentUrl, 21	removeItemfromInvoice
getState, 21	PnagInvoice, 24
getStatus, 21	removeSofortDauerauftrag
getStatusOfInvoice, 21	SofortLib_CancelSa, 36
getStatusReason, 22	
getTransactionData, 22	removeSofortrechnungItem
getTransactionId, 22	SofortLib_Multipay, 54
getWarnings, 22	sendRequest
isError, 22	SofortLib_Debit, 45
isWarning, 23	SofortLib Refund, 66
log, 23	sendValidationRequest
logError, 23	SofortLib_Multipay, 55
	± •
logWarning, 23	setAbortUrl
refreshTransactionData, 24	PnagInvoice, 24
removeItemfromInvoice, 24	SofortLib_CancelSa, 36
setAbortUrl, 24	SofortLib_Multipay, 55
setAmount, 24	SofortLib
setBitmask, 24	SofortueberweisungClassic,

69	SofortLib
setAmount	SofortueberweisungClassic,
PnagInvoice, 24	69
SofortLib_Debit, 45	setNumber
SofortLib_Multipay, 55	SofortLib_TransactionData, 89
setApiVersion	setOrderId
SofortLib, 35	PnagInvoice, 26
setBitmask	setPaymentMethod
PnagInvoice, 24	PnagInvoice, 27
setComment	setPhoneNumberCustomer
SofortLib_CancelSr, 39	PnagInvoice, 27
	SofortLib_Multipay, 57
SofortLib_ConfirmSr, 43	setProjectId
SofortLib_EditSr, 48	SofortLib_Debit, 45
setCustomerId	setReason
PnagInvoice, 25	PnagInvoice, 27
setDate	SofortLib_Debit, 46
SofortLib_Debit, 45	SofortLib_Multipay, 58
setDebitorVatNumber	SofortLib
PnagInvoice, 25	SofortueberweisungClassic,
SofortLib_Multipay, 55	69
setEmailCustomer	setSenderAccount
PnagInvoice, 25	SofortLib_Debit, 46
SofortLib_Multipay, 56	SofortLib_Multipay, 58
setInvoiceObjection	SofortLib_Refund, 67
PnagInvoice, 25	setSenderAccountNumber
setInvoiceStatus	SofortLib_Debit, 46
PnagInvoice, 25	setSenderBankCode
setInvoiceUrl	SofortLib_Debit, 46
PnagInvoice, 26	SofortLib_iDealClassic, 50
setItems	setSenderCountryId
PnagInvoice, 26	SofortLib_iDealClassic, 50
setLanguageCode	setSenderHolder
PnagInvoice, 26	SofortLib_Debit, 47
SofortLib_Multipay, 56	setSofortDauerauftrag
setLastschrift	SofortLib_Multipay, 58
SofortLib_Multipay, 56	setSofortDauerauftragInterval
setLastschriftAddress	SofortLib_Multipay, 58
SofortLib_Multipay, 56	setSofortDauerauftragMinimumPayments
setLogDisabled	SofortLib_Multipay, 59
SofortLib, 35	setSofortDauerauftragStartDate
setLogEnabled	SofortLib_Multipay, 59
SofortLib, 35	setSofortDauerauftragTotalPayments
setLogger	SofortLib_Multipay, 59
SofortLib, 35	setSofortlastschrift
setNotificationEmail	SofortLib_Multipay, 59
SofortLib_Multipay, 57	setSofortlastschriftAddress
setNotificationUrl	SofortLib_Multipay, 60
PnagInvoice, 26	setSofortrechnung
SofortLib_Multipay, 57	PnagInvoice, 27

C. C. A. H. M. M. L	
SofortLib_Multipay, 60	setVersion
setSofortrechnungCustomerId	PnagInvoice, 30
SofortLib_Multipay, 60	SofortLib_Multipay, 64
setSofortrechnungInvoiceAddress	SofortLib, 30
SofortLib_Multipay, 61	construct, 31
setSofortrechnungOrderId	_sendMessage, 32
SofortLib_Multipay, 61	deleteAllErrors, 32
setSofortrechnungShippingAddress	deleteAllWarnings, 32
SofortLib_Multipay, 61	disableLog, 32
setSofortueberweisung	enableLog, 32
SofortLib_Multipay, 62	error, 32
set So for tue berweisung Customer protection	fatalError, 32
SofortLib_Multipay, 62	getError, 32
setSofortvorkasse	getErrors, 33
SofortLib_Multipay, 62	getWarnings, 33
set So for tvork as se Customer protection	handleErrors, 33
SofortLib_Multipay, 63	isError, 33
setState	isWarning, 34
PnagInvoice, 27	log, 34
setStatus	logError, 34
PnagInvoice, 28	logWarning, 34
setStatusOfInvoice	setApiVersion, 35
PnagInvoice, 28	setLogDisabled, 35
setStatusReason	setLogEnabled, 35
PnagInvoice, 28	setLogger, 35
setSuccessUrl	sofortLib, 11
PnagInvoice, 28	SofortLib_CancelSa, 35
SofortLib_CancelSa, 37	construct, 36
SofortLib_Multipay, 63	removeSofortDauerauftrag, 36
SofortLib	setAbortUrl, 36
SofortueberweisungClassic,	setSuccessUrl, 37
70	setTimeoutUrl, 37
setTime	setTransaction, 37
PnagInvoice, 29	SofortLib_CancelSr, 38
SofortLib_TransactionData, 90	construct, 38
setTimeout	cancelInvoice, 38
PnagInvoice, 29	getInvoiceUrl, 39
SofortLib_Multipay, 63	setComment, 39
setTimeoutUrl	setTransaction, 39
PnagInvoice, 29	SofortLib ClassicNotification, 39
SofortLib_CancelSa, 37	_getHashHexValue, 40
SofortLib_Multipay, 63	SofortLib_ConfirmSr, 40
setTransaction	construct, 41
PnagInvoice, 29	addItem, 41
SofortLib_CancelSa, 37	cancelInvoice, 42
SofortLib_CancelSr, 39	confirmInvoice, 42
SofortLib_ConfirmSr, 43	getInvoiceUrl, 42
SofortLib_TransactionData, 90	removeItem, 42
setTransactionId	setComment, 43
PnagInvoice, 29	setTransaction, 43
i naginvoice, 27	sections, TS

updateCart, 43	setSofortDauerauftragStartDate, 59
SofortLib_Debit, 44	setSofortDauerauftragTotalPay-
addReason, 44	ments, 59
addUserVariable, 44	setSofortlastschrift, 59
getTransactionId, 45	setSofortlastschriftAddress, 60
sendRequest, 45	setSofortrechnung, 60
setAmount, 45	setSofortrechnungCustomerId, 60
setDate, 45	setSofortrechnungInvoiceAddress,
setProjectId, 45	61
setReason, 46	setSofortrechnungOrderId, 61
setSenderAccount, 46	setSofortrechnungShippingAddress,
setSenderAccountNumber, 46	61
setSenderBankCode, 46	setSofortueberweisung, 62
setSenderHolder, 47	setSofortueberweisungCustomerpro-
SofortLib_EditSr, 47	tection, 62
construct, 48	setSofortvorkasse, 62
addItem, 48	setSofortvorkasseCustomerprotec-
getInvoiceUrl, 48	tion, 63
setComment, 48	setSuccessUrl, 63
SofortLib_iDeal_Banks, 49	setTimeout, 63
SofortLib_iDealClassic, 49	setTimeoutUrl, 63
getRelatedBanks, 50	setVersion, 64
setSenderBankCode, 50	updateSofortrechnungItem, 64
setSenderCountryId, 50	SofortLib_Notification, 64
SofortLib_Multipay, 51	construct, 65
construct, 53	getNotification, 65
addNotificationEmail, 53	SofortLib_Refund, 66
addNotificationUrl, 53	addRefund, 66
addSofortrechnungItem, 53	sendRequest, 66
addUserVariable, 54	setSenderAccount, 67
getPaymentUrl, 54	SofortLib_SofortueberweisungClassic,
getTransactionId, 54	67
removeSofortrechnungItem, 54	generatePassword, 68
sendValidationRequest, 55	getHashHexValue, 68
setAbortUrl, 55	getSupportedHashAlgorithm, 68
setAmount, 55	getSupportedHashAlgorithms, 69
setDebitorVatNumber, 55	setAbortUrl, 69
setEmailCustomer, 56	setNotificationUrl, 69
setLanguageCode, 56	setReason, 69
setLastschrift, 56	setSuccessUrl, 70
setLastschriftAddress, 56	SofortLib_TransactionData, 70
setNotificationEmail, 57	addTransaction, 72
setNotificationUrl, 57	getAmount, 72
setPhoneNumberCustomer, 57	getAmountRefunded, 73
setReason, 58	getConsumerProtection, 73
setSenderAccount, 58	getCurrency, 73
setSofortDauerauftrag, 58	getInvoiceAddress, 73
setSofortDauerauftragInterval, 58	getInvoiceBankAccountNumber, 74
setSofortDauerauftragMinimumPay-	getInvoiceBankCode, 74
ments, 59	getInvoiceBankHolder, 74

getInvoiceBankName, 74	getSofortdauerauftragStartDate, 85
getInvoiceDate, 75	getSofortdauerauftragTotalPay-
getInvoiceDebitorText, 75	ments, 85
getInvoiceDueDate, 75	getStatus, 85
getInvoiceNumber, 75	getStatusModifiedTime, 85
getInvoiceObjection, 76	getStatusReason, 86
getInvoiceReason, 76	getTime, 86
getInvoiceStatus, 76	getTransaction, 86
getInvoiceType, 76	getUserVariable, 86
getInvoiceUrl, 77	isLastschrift, 87
getItems, 77	isLoss, 87
getLanguageCode, 77	isPending, 87
getPaymentMethod, 77	isReceived, 87
getProjectId, 78	isRefunded, 88
getReason, 78	isSofortdauerauftrag, 88
getRecipientAccountNumber, 78	isSofortlastschrift, 88
getRecipientBankCode, 78	isSofortrechnung, 88
getRecipientBankName, 79	isSofortueberweisung, 89
getRecipientBic, 79	isSofortvorkasse, 89
getRecipientCountryCode, 79	isTest, 89
getRecipientHolder, 79	setNumber, 89
getRecipientIban, 80	setTime, 90
getSenderAccountNumber, 80	setTransaction, 90
getSenderBankCode, 80	
getSenderBankName, 80	updateCart
getSenderBic, 81	SofortLib_ConfirmSr, 43
getSenderCountryCode, 81	updateInvoice
getSenderHolder, 81	PnagInvoice, 30
getSenderIban, 81	updateOrderNumber
getShippingAddress, 82	PnagInvoice, 30
getSofortdauerauftragAllPay-	updateSofortrechnungItem
mentsReceived, 82	SofortLib_Multipay, 64
getSofortdauerauftragInterval, 82	
getSofortdauerauftragLastPaymen-	
tReceived, 82	
getSofortdauerauftragMinimumPay-	
ments, 83	
getSofortdauerauftragNumberOf-	
PaymentsPending, 83	
getSofortdauerauftragNumberOf-	
PaymentsReceived, 83	
getSofortdauerauftragPayment-	
Count, 83	
getSofortdauerauftragPaymentRe-	
ceived, 84	
getSofortdauerauftragPayments, 84	
getSofortdauerauftragPaymentsRe-	
ceived, 84	
get So fort dauer auf trag Payments Sum,	
84	