



# Komfortkasse API

Version: 1.3

## Inhalt

Allgemeines .....	2
Konfiguration.....	3
Bestellung.....	3
Formatierung.....	4
API-Aufrufe .....	4
Senden einer neuen Bestellung an Komfortkasse.....	4
Rückgabewerte der API .....	5
Senden einer neuen Rückerstattung an Komfortkasse.....	6
Aufrufe durch Komfortkasse .....	7
Auslesen von bestehenden Bestellungen durch Komfortkasse .....	7
Schritt 1: Bestellnummern.....	8
Schritt 2: Bestellinformationen (optional) .....	8
Update des Zahlungsstatus durch Komfortkasse.....	9
Auslesen von Rückerstattungen durch Komfortkasse .....	10
Update von Informationen zur Rückerstattungen durch Komfortkasse.....	10
Beispiel-Implementierungen .....	10
PHP .....	10

## Allgemeines

Mit der Komfortkasse API können Sie Ihr eigenes Shopsystem an Komfortkasse anbinden.

Die Komfortkasse API basiert auf HTTP-Aufrufen, ähnlich einem Webservice. Die Verwendung der API ist grundsätzlich nur möglich, wenn Ihr Server HTTPS unterstützt. (API-Aufrufe ohne HTTPS-Verbindung sind mit erhöhtem Aufwand auf Anfrage realisierbar. In diesem Fall muss eine eigene Verschlüsselung, z.B. auf Basis von OpenSSL oder mcrypt, umgesetzt werden.)

Die API unterstützt derzeit folgende Aufrufe:

Bestellungen:

1. Senden einer neuen Bestellung an Komfortkasse (`neworder`)
2. Auslesen von bestehenden Bestellungen durch Komfortkasse (2-stufig) (`readorders`)
3. Update des Zahlstatus durch Komfortkasse (`updateorder`)
4. Abrufen von Bestelldaten an Komfortkasse (`getorder`)\*

Rückerstattungen:

1. Senden einer Rückerstattung an Komfortkasse (`newrefund`)
2. Auslesen von Rückerstattungen durch Komfortkasse (`readrefunds`)
3. Rückmeldung zu Rückerstattungen durch Komfortkasse (`updaterefund`)

\* auf Anfrage

Die API-Aufrufe durch Ihren Shop (Fall 1.) erfolgen mit einem **API-Key**, der von Komfortkasse generiert wird.

Die Aufrufe Ihres Shopsystems durch Komfortkasse (Fall 2. und 3.) erfolgen mit einem **Zugriffsschlüssel** (Access Code), der ebenfalls von Komfortkasse generiert wird. Sie sollten in Ihrem Shopsystem diesen Zugriffsschlüssel immer prüfen, um unbefugten Zugriff zu verhindern.

Um Übertragungsprobleme durch unterschiedliche Zeichenkodierungen zu verhindern, erfolgt jegliche Kommunikation Base64-kodiert.

## Konfiguration

Im Komfortkasse Händlerbereich hinterlegen Sie den Pfad, der von Komfortkasse verwendet wird, um auf Ihr Shopsystem zuzugreifen. Hier finden Sie auch den API-Key und Zugriffsschlüssel sowie die URL der Komfortkasse API.

Einstellungen für API-Zugriff	
API Key:	6O8qsgnjR5XIMH6PkWWDOSNhzczWIZWw
Access Code:	52d75f8a26f8790925dff2b9e2a04c8
API URL:	https://ssl.komfortkasse.eu
Pfad:	/kk/komfortkasse_receive.php
<input type="button" value="✓ Einstellungen speichern"/>	

Für die folgenden Beispiele wird die API URL `api.komfortkasse.eu` verwendet. **Achtung!** Verwenden Sie unbedingt die in Ihrem Händlerbereich angegebene API URL!

## Bestellung

Eine Bestellung in Komfortkasse besteht aus folgenden Informationen:

Bestellnummer	Pflicht
Bestelldatum	Pflicht
E-Mail Adresse des Käufers	Pflicht
Betrag Währung	Pflicht
Kundennummer des Käufers	Optional
Zahlungsmethode	Optional
Wechselkurs (zu EUR)	Optional*
Bestellsprache	Pflicht
Land des Käufers**	Optional
Firmenname, Vorname, Nachname der Rechnungsadresse	Pflicht
Firmenname, Vorname, Nachname der Lieferadresse	Pflicht
Artikelnummern	Optional
Straße, PLZ, Ort von Rechnungs- und Lieferadresse***	Optional

\* Pflicht, wenn die Wahrung nicht in Ihrem Konto verwaltet wird. Beispiel: Sie verwenden EUR und CHF in Ihrem Handlerkonto. Fur diese Wahrungen mussen Sie keinen Wechselkurs ubertragen. Wenn Sie eine Bestellung in einer anderen Wahrung, z.B. USD, ubertragen, mussen Sie einen Wechselkurs mit ubertragen.

\*\* Wird fur die Lokalisierung der E-Mail-Texte verwendet. Sollte das Land der Rechnungsadresse, nicht der Lieferadresse sein.

\*\*\* Empfohlen, wenn Sie Kunden in der Schweiz haben

## Formatierung

Dezimalzahlen werden durch Punkt getrennt und ohne Tausender-Trennzeichen angegeben. Beispiel: 1200.34

Listen werden als Array mit fortlaufendem Index (beginnend bei 0) formatiert. Beispiel:

```
products[0]=ABC  
products[1]=XYZ
```

## API-Aufrufe

Der API Aufruf erfolgt per http-Request an die URL

`https://API-URL/api/SECTION/ACTION.jsf`

Die API-URL wird in Ihrem Handlerbereich angezeigt.

Die SECTION ist der Teilbereich der API, z.B. `shop` fur API-Aufrufe von Webshops.

Die ACTION spezifiziert die Aktion die ausgefuhrt werden soll, z.B. `neworder` fur das Anlegen einer neuen Bestellung.

## Senden einer neuen Bestellung an Komfortkasse

Dieser Aufruf wird von Ihrem Shopsystem abgesetzt, sobald eine neue Bestellung eingeht. Der Aufruf wird nur fur neue Bestellungen abgesetzt, nicht, wenn sich eine bestehende Bestellung andert.

```
SECTION: shop  
ACTION: neworder
```

Beispiel-URL: <https://api.komfortkasse.eu/api/shop/neworder.jsf>

Daten fur die Bestellung:

<code>number</code>	Bestellnummer	Alphanumerisch, eindeutig*
<code>date</code>	Bestelldatum	Datum, Format TT.MM.JJJJ
<code>email</code>	E-Mail Adresse des Kaufers	Alphanumerisch
<code>amount</code>	Betrag	Dezimal mit 2 Nachkommastellen
<code>currency_code</code>	Wahrung	Wahrungscod, 3-stellig (ISO 4712)
<code>customer_number</code>	Kundennummer des Kaufers	Alphanumerisch
<code>payment_method</code>	Zahlungsmethode	Alphanumerisch
<code>exchange_rate</code>	Wechselkurs (zu EUR)	Dezimal mit 4 Nachkommastellen

language_code	Bestellsprache und Land des Käufers **	Sprachcode, 2-stellig (ISO 639-1) Ländercode, 3-stellig (ISO-3166-1 ALPHA-3)
billing_firstname	Vorname (Rechnungsadresse)	Alphanumerisch
billing_lastname	Nachname (Rechnungsadresse)	Alphanumerisch
billing_company	Firma (Rechnungsadresse) ***	Alphanumerisch
billing_street	Straße und Hausnummer (Rechnungsadresse)	Alphanumerisch
billing_postcode	Postleitzahl (Rechnungsadresse)	Alphanumerisch
billing_city	Ort (Rechnungsadresse)	Alphanumerisch
delivery_firstname	Vorname (Lieferadresse)	Alphanumerisch
delivery_lastname	Nachname (Lieferadresse)	Alphanumerisch
delivery_company	Firma (Lieferadresse) ***	Alphanumerisch
delivery_street	Straße und Hausnummer (Lieferadresse)	Alphanumerisch
delivery_postcode	Postleitzahl (Lieferadresse)	Alphanumerisch
delivery_city	Ort (Lieferadresse)	Alphanumerisch
products	Artikelnummern	Liste (jeweils alphanumerisch)

\* Die Bestellnummer ist die eindeutige Identifikation einer Bestellung. Wenn eine Bestellung übertragen wird, deren Bestellnummer bereits vorhanden ist, wird ein Fehler ausgegeben. Änderungen zu bestehenden Bestellungen werden nicht über die API übertragen, sondern von Komfortkasse ausgelesen (Fall 2.).

\*\* getrennt durch Bindestrich „-“, z.B. de-CH, en-UK. Wenn nur der Sprachcode angegeben wird, kann der Bindestrich entfallen, z.B. de.

\*\*\* Parameter nicht oder leer übertragen, wenn kein Firmenname angegeben wurde

Für den API-Aufruf werden die Parameter URL-kodiert und Base64-kodiert.

Beispiel für URL-Kodierung:

number=1234&date=31.12.2014&amount=1200.34&currency\_code=EUR

Base64-Kodierung des Beispiels (kein Zeilenumbruch):

bnVtYmVyPTEyMzQmZGF0ZT0zMS4xMi4yMDE0JmFtb3VudD0xMjAwLjM0JmN1cnJlbnN5X2NvZGU9RVVS

Parameter für API-Aufruf:

q	Base64-kodierte Parameter	Alphanumerisch
hash	Zugriffsschlüssel (Access Code)	Alphanumerisch
key	API-Key	Alphanumerisch

Der API-Aufruf sollte als POST erfolgen. Die Verwendung von GET ist möglich, kann aber zu Fehlern führen wenn die maximale Länge eines GET-Requests überschritten wird. Daher wird dringend empfohlen, POST Requests zu verwenden.

## Rückgabewerte der API

Die API gibt ein XML-Dokument der folgenden Form zurück:

```
<result>
  <code>CODE</code>
  <message>MESSAGE</message>
</result>
```

code	Rückgabewert. Mögliche Werte: OK, ERROR	Alphanumerisch
message	Im Falle von ERROR wird eine Fehlermeldung ausgegeben	Alphanumerisch

Wenn erfolgreich ein Objekt (Bestellung oder Rückerstattung) erzeugt wurde, werden auch Daten zum Objekt mit ausgegeben:

```
<result>
  <code>CODE</code>
  <message>MESSAGE</message>
  <object type="order">
    <referenceNumber>K1234X</referenceNumber>
    <amount currency="EUR">12.34</amount>
    <remittee>meinshop.de</remittee>
    <account>
      <iban>DE1234567890123456</iban>
      <bic>BANKDEX1</bic>
    </account>
  </object>
</result>
```

### Senden einer neuen Rückerstattung an Komfortkasse

Dieser Aufruf wird von Ihrem Shopsystem abgesetzt, um eine Rückerstattung auszulösen.

SECTION: shop  
ACTION: newrefund

Beispiel-URL: <https://api.komfortkasse.eu/api/shop/newrefund.jsf>

Daten für die Rückerstattung:

external_reference	Referenznummer	Alphanumerisch, eindeutig
number	Bestellnummer*	Alphanumerisch
amount	Betrag**	Dezimal mit 2 Nachkommastellen
currency_code	Währung**	Währungscode, 3-stellig (ISO 4712)
remittee	Zahlungsempfänger**	Alphanumerisch
email	E-Mail Adresse des Zahlungsempfängers***	Alphanumerisch
language	Sprache des Zahlungsempfängers****	Sprachcode, 2-stellig (ISO 639-1)

\* In den Einstellungen wird festgelegt, ob auch Rückerstattungen erfasst werden können, die nicht zuvor über Komfortkasse bezahlt wurden. Wenn ja, kann dieses Feld leer sein.

\*\* Kann leer sein, wenn Bestellnummern angegeben. In diesem Fall wird der gesamte Betrag aus der Bestellung zurück erstattet.

\*\*\* Wird ignoriert, wenn Bestellnummer angegeben. E-Mail Adresse und Sprache werden benötigt, um den Zahlungsempfänger nach Bankverbindung zu fragen.

## Aufrufe durch Komfortkasse

API-Aufrufe führt Ihr System bei Komfortkasse durch.

Aufrufe durch Komfortkasse werden von Komfortkasse auf Ihrem System durchgeführt („callback“).

Die Aufrufe erfolgen per http POST Request an folgende URL:

`https://<Shop-URL>/<Pfad>`

Die Shop-URL und der Pfad werden im Komfortkasse Händlerbereich hinterlegt:

The screenshot shows two sections of the merchant interface. The top section, titled 'Shop www.meinshop.de', contains the following fields: 'URL' (www.meinshop.de), 'Shopsystem' (Sonstiges (generisches callback)), 'Status' (Nicht verbunden), 'Fehler (letztes Monat)' (0), and 'Verschlüsselung' (HTTPS). There are buttons for 'Bestellungen einlesen' and 'Update durchführen'. The bottom section, titled 'Einstellungen für API-Zugriff', contains 'API Key' (608qsgnjR5XIMH6PKWWDOSNhzcZWIZWw), 'Access Code' (52d75f8a26f9790925dfd2b9e2a04c8), and 'Pfad' (/kk/komfortkasse\_receive.php). There is a button for 'Einstellungen speichern'.

Folgende Parameter werden übermittelt:

action	Aktion	Alphanumerisch
o	Base64-kodierte Informationen	Alphanumerisch
accesscode	Zugriffsschlüssel (Access Code)	Alphanumerisch

Prüfen Sie immer, ob der übertragene Zugriffsschlüssel mit Ihrem hinterlegten Zugriffsschlüssel übereinstimmt.

Für den Parameter `action` sind folgende Werte möglich:

readorders	Auslesen von Bestellinformationen
updateorders	Update des Zahlungsstatus von Bestellungen
readrefunds	Auslesen von im Shop erfassten Rückerstattungen
updaterefunds	Update des Status von Rückerstattungen

Im Parameter `o` werden Base64-kodierte Informationen übertragen. Die Informationen unterscheiden sich je `action`.

## Auslesen von bestehenden Bestellungen durch Komfortkasse

Wenn in Ihrem System eine neue Bestellung erfasst wird, wird diese durch den Aufruf „neworder“ an Komfortkasse übertragen (siehe oben).

Zusätzlich fragt Komfortkasse in regelmäßigen Abständen in Ihrem System die vorhandenen Bestellungen ab. Dies hat folgende Gründe:

- Falls ein API-Aufruf von `neworder` nicht durchgeführt werden konnte, z.B. aufgrund von Netzwerkproblemen
- Falls eine Bestellung in Ihrem Shopsystem manuell erledigt, storniert oder gelöscht wurde
- Falls eine Bestellung in Ihrem Shopsystem verändert wurde (z.B. nachträglicher Rabatt, Änderung von Bestellpositionen)

Die Abfrage erfolgt in zwei Stufen: zuerst wird von Komfortkasse angefragt, welche Bestellnummern noch offen sind. Anschließend werden zu jeder Bestellnummer die Bestellinformationen übertragen.

#### Schritt 1: Bestellnummern

Als `action` wird `readorders` übertragen.

Als Parameter `o` wird der fixe Wert `all` übergeben (Base64-kodiert).

Als Ausgabe erwartet Komfortkasse eine Liste der Bestellnummern, die im Shopsystem für Komfortkasse relevant sind. Die Liste wird in einer Zeile, Semikolon-getrennt, ausgegeben und anschließend Base64-kodiert.

Beispiel:

```
1000123;1000456;1000789
```

#### Schritt 2: Bestellinformationen (optional)

*Wenn Sie diese Funktion nicht implementieren, werden keine Änderungen an Komfortkasse übertragen. Wenn Sie diese Funktion nicht implementieren, müssen Sie sicherstellen, dass jede Bestellung durch `neworder` an Komfortkasse übertragen wird. Wenn der Aufruf von `neworder` fehlschlägt (z.B. durch Netzwerkprobleme), muss dieser solange wiederholt werden, bis der Aufruf erfolgreich war. Wenn Sie Schritt 2 nicht implementieren, muss beim Aufruf ein leerer String zurückgegeben werden.*

Als `action` wird `readorders` übertragen.

Als Parameter `o` wird eine Liste von Bestellnummern übergeben, zu denen Bestellinformationen ausgelesen werden (Base64-kodiert).

Beispiel:

```
1000123;1000456;1000789
```

Als Ausgabe erwartet Komfortkasse URL-kodierte Bestellinformationen (Aufbau gleich wie beim Senden einer neuen Bestellung). Pro Zeile wird Bestellung ausgegeben. Die gesamte Ausgabe wird Base64-kodiert.

Beispiel:

```
number=1000123&date=31.12.2014&amount=1200.34&currency_code=EUR&...  
number=1000456&date=05.01.2015&amount=748.22&currency_code=EUR&...  
number=1000789&date=01.01.2015&amount=18.87&currency_code=CHF&...
```

Die Reihenfolge der Ausgabe ist beliebig.

## Update des Zahlungsstatus durch Komfortkasse

Wenn eine Zahlung eingeht oder eine Bestellung storniert wird, wird der Zahlungsstatus an Ihr Shopsystem übertragen.

Als `action` wird `updateorders` übertragen.

Als Parameter `o` werden Bestellnummern und der zu setzende Status übergeben (jeweils eine Bestellung pro Zeile, Semikolon-getrennt).

Folgende Informationen werden übergeben:

Bestellnummer	Pflicht	Alphanumerisch
Zahlungsstatus	Pflicht	Alphanumerisch
Callback-ID	Optional	Alphanumerisch

Der Zahlungsstatus kann folgende Werte annehmen:

PAID	Vollständig bezahlt
PAID_PARTIAL	Teilweise bezahlt
CANCELLED	Storniert
REFUNDED	Vollständig zurück erstattet
REFUNDED_PARTIAL	Teilweise zurück erstattet

Als Callback-ID wird eine von Komfortkasse gewählte ID mit übergeben. Dies ist bei einem Zahlungseingang z.B. die ID des Zahlungseingangs oder `MANUAL`, wenn die Zahlung manuell in Komfortkasse bearbeitet wurde (z.B. storniert wurde oder auf bezahlt gesetzt wurde). Die Callback-ID kann leer sein oder nicht mit übergeben werden.

Beispiel:

```
1000123;PAID;RC65HS765FE  
1000456;CANCELLED;MANUAL  
1000789;REFUNDED;
```

Als Antwort erwartet Komfortkasse eine Liste von Bestellnummern, bei denen der Status erfolgreich aktualisiert wurde. Wenn der Status nicht aktualisiert werden konnte (z.B. wegen Software-Fehler oder weil die Bestellung in der Zwischenzeit gelöscht wurde) wird Komfortkasse 7 Tage lang weiter versuchen, den Zahlungsstatus zu aktualisieren.

Nicht alle Zahlungsstatus müssen vom Shopsystem unterstützt werden. So ist es z.B. möglich, nur `PAID` zu unterstützen, aber `PAID_PARTIAL` zu ignorieren. In diesem Fall sollte die Bestellnummer ebenfalls nicht ausgegeben werden. Es sollte mindestens `PAID` und `CANCELLED` unterstützt werden. (Theoretisch ist es auch möglich, gar keinen Zahlungsstatus zu unterstützen und nur die Benachrichtigungs-Funktionen von Komfortkasse zu verwenden.)

Beispiel:

```
1000123;1000456
```

(Bedeutung: Die Bestellungen 1000123 und 1000456 wurden aktualisiert. Die Bestellung 1000789 wurde nicht aktualisiert, da der Status `REFUNDED` nicht unterstützt wird.)

## Auslesen von Rückerstattungen durch Komfortkasse

Wenn in Ihrem System eine neue Rückerstattung erfasst wird, wird diese durch den Aufruf „newrefund“ an Komfortkasse übertragen (siehe oben).

Zusätzlich fragt Komfortkasse in regelmäßigen Abständen in Ihrem System die vorhandenen Rückerstattungen ab. **Änderungen an Rückerstattungen werden nicht übertragen.**

Als `action` wird `readrefunds` übertragen. Die Übertragung erfolgt analog zum Aufruf von `readorders` in zwei Schritten (siehe dort).

## Update von Informationen zur Rückerstattungen durch Komfortkasse

Wenn sich der Status der Rückerstattung ändert oder ein Problem damit aufgetreten ist, wird der Status an Ihr Shopsystem übertragen.

Als `action` wird `updaterefunds` übertragen.

Als Parameter `o` werden Bestellnummern und eine Information übergeben (jeweils eine Bestellung pro Zeile, Semikolon-getrennt).

Folgende Informationen werden übergeben:

Referenznummer	Pflicht	Alphanumerisch
Information	Pflicht	Alphanumerisch

Beispiel:

```
R100123;Durchgeführt  
R100456;Storniert  
R100789;Fehler: ...
```

Fehlermeldungen beginnen immer mit „Fehler:“.

## Beispiel-Implementierungen

Wir empfehlen dringend, vor der eigenen Implementierung eine Beispiel-Implementierung zu installieren um den Ablauf durchzutesten. Sie können auch Ihre eigene Implementierung auf diesen Beispiel-Dateien aufbauen. Sie finden alle Beispiel-Implementierungen zum Download unter [komfortkasse.eu/info/api](http://komfortkasse.eu/info/api).

Derzeit steht eine Beispiel-Implementierung in PHP zur Verfügung.

### PHP

Für die Beispiel-Implementierung benötigen Sie einen Webserver mit PHP-Unterstützung. Sie benötigen keinen Datenbank-Server (die Beispiel-Implementierung verwendet Textfiles zur Speicherung von Daten).

Sie benötigen einen gültigen Komfortkasse Account (Sie können sich zu Testzwecken einen *free*-Account erstellen).

Die Beispiel-Implementierung enthält folgende Dateien:

README.txt	Anleitung zur Installation und Verwendung
komfortkasse_send.php	API-Aufrufe
komfortkasse_receive.php	Endpunkt für die Aufrufe durch Komfortkasse
komfortkasse_settings.php	Einstellungen (API Key und Accesscode)
komfortkasse_buildorder.php	erzeugt Bestellungen (mit zufälligen Bestellnummern)

Sie können die gesamte Funktionalität mit der Beispiel-Implementierung testen. Folgen Sie dazu einfach den Anweisungen in der README.txt.