

# Booking Records

Diese Seite enthält noch nicht alle Informationen bzw. befindet sich aktuell in der Überarbeitung

API Klassen und deren Funktionen können nur mit TecArt-System Version  $\geq 3.2$  genutzt werden

## 1. Methods

- 1.1. [addBookingRecord](#)
- 1.2. [searchBookingRecords](#)
- 1.3. [changeBookingRecord](#)
- 1.4. [deleteBookingRecord](#)
- 1.5. [getBookingRecord](#)
- 1.6. [getByIdExtId](#)
- 1.7. [getAllExtIds](#)

## 1. Methods

### 1.1. addBookingRecord

(Version  $\geq 4.0.9357$ )

#### Syntax:

```
int addBookingRecord(array $crm_booking_record)
```

#### Beschreibung:

Erzeugt eine neue Buchung im Buchungsjournal. Gibt im Erfolgsfall die ID des Datensatzes zurück.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

*crm\_booking\_record* kann ein assoziatives Array oder ein Objekt sein.

#### Eigenschaften:

Type (Length)	Eigenschaft	Beschreibung
int	\$payment_id	ID der Zahlung, zu der Buchungssatz zugewiesen werden soll
string	\$realnumber	Belegnummer
string	\$realnumber_activity	Belegnummer-ID
string	\$account_in	Gegenkonto
string	\$account_out	Buchungskonto
string	\$description	Beschreibung
string	\$notes	Notizen
double	\$amount	Umsatz
double	\$tax	Steuer
double	\$amount_brutto	Betrag
int	\$timestamp	Datum als Unix-Timestamp
int	\$ext_id	Externe ID ( ab Version 4.2 )

#### Fetter Text

#### Beispiel:

```

<?php

try {
    $api = crmapi::booking_records();

    $data = array(
        'payment_id'          => 1000,
        'realnumber'         => 100000,
        'realnumber_activity' => 1000,
        'account_in'         => '100000',
        'account_out'        => '100001',
        'description'        => 'API-Test',
        'notes'              => 'API-Test',
        'amount'             => 100,
        'tax'                => 19,
        'amount_brutto'      => 119,
        'tstamp'            => time()
    );
    $id = $api->addBookingRecord($data);
}
catch (CRMException $e) {
    echo $e->getBookingRecordMessage();
}

echo "Booking record id: $id<br>";

```

## 1.2. searchBookingRecords

(Version >= 4.0.9357)

### Syntax:

```
array searchBookingRecords(array $search_params)
```

### Beschreibung:

Sucht nach Buchungen anhand der übergebenen Suchparameter. Gibt bei Erfolg ein Array mit Buchungssätzen zurück.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

*search\_params* kann ein assoziatives Array oder ein Objekt sein.

### Eigenschaften:

Type (Length)	Eigenschaft	Beschreibung
string	\$search	Suchbegriff
string	\$posting_account	Buchungskonto
string	\$realnumber_activity	Belegnummer-ID
string	\$payment_id	Zahlung-ID
int	\$count	maximale Anzahl an Elementen, die zurückgegeben wird
int	\$offset	Position der Ergebnisse, ab der zurückgegeben werden soll
bool	\$sorting	Einstellung, ob Elemente sortiert zurückgegeben werden

### Eigenschaften der zurückgegebenen Array

Type(Length)	Eigenschaft	Beschreibung
int	\$id	ID des Datensatzes
int	\$payment_id	ID der Zahlung
string	\$realnumber	Belegnummer
string	\$realnumber_activity	Belegnummer-ID
string	\$account_in	Gegenkonto
string	\$account_out	Buchungskonto
string	\$description	Beschreibung
string	\$notes	Notizen
double	\$amount	Umsatz
double	\$tax	Steuer
double	\$amount_brutto	Betrag
int	\$tstamp	Datum als Unix-Timestamp
int	\$createuser	Benutzer-ID, der Buchung erzeugt hat
int	\$createtime	Zeitstempel, an dem Buchung erzeugt wurde
int	\$chguser	Benutzer-ID, der Buchung zuletzt geändert hat
int	\$chgtime	Zeitstempel, an dem zuletzt geändert wurde

#### Beispiel:

```
<?php
try {
    $api = crmapl::booking_records();

    $params = array(
        'search' => "description LIKE '%API-Test%'",
        'payment_id' => 1000
    );
    $results = $api->searchBookingRecords($params);
}
catch (CRMException $e) {
    echo $e->getBookingRecordMessage();
}

foreach($results as $result) {
    foreach($result as $key => $val) {
        echo "$key => $val<br> \n";
    }
}
echo "<hr/>";
}
```

## 1.3. changeBookingRecord

(Version >= 4.1)

#### Syntax:

```
int changeBookingRecord(int $booking_record_id, array $crm_booking_record)
```

#### Beschreibung:

Aktualisiert eine die Buchung mit der übergebenen ID anhand der übergebenen Daten.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

*crm\_booking\_record* kann ein assoziatives Array oder ein Objekt sein.

#### Eigenschaften:

Eigenschaften: sind die selben wie für [addBookingRecord](#)

#### Beispiel:

```
<?php

try {
    $data = array(
        'notes' => 'Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.'
    );

    echo $api->changeBookingRecord($booking_record_id, $data);
}
catch (CRMException $e) {
    echo $e->getMessage();
}
```

---

## 1.4. deleteBookingRecord

(Version >= 4.1)

### Syntax:

*int deleteBookingRecord(int \$booking\_record\_id)*

### Beschreibung:

Löscht die Buchung mit der übergebenen ID aus dem Buchungsjournal. Gibt im Erfolgsfall 1 zurück.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

### Beispiel:

```
<?php

try {
    echo $api->deleteBookingRecord($booking_record_id);
}
catch (CRMException $e) {
    echo $e->getMessage();
}
```

---

## 1.5. getBookingRecord

(Version >= 4.1)

### Syntax:

*array getBookingRecord(int \$booking\_record\_id)*

### Beschreibung:

Gibt den Buchungssatz mit der gewählten ID als Array zurück.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

### Eigenschaften der zurückgegebenen Array

Type(Length)	Eigenschaft	Beschreibung
int	\$id	ID des Datensatzes
int	\$ext_id	Externe ID ( ab Version 4.2 )
int	\$payment_id	ID der Zahlung
string	\$realnumber	Belegnummer
string	\$realnumber_activity	Belegnummer-ID
string	\$account_in	Gegenkonto
string	\$account_out	Buchungskonto
string	\$description	Beschreibung
string	\$notes	Notizen
double	\$amount	Umsatz
double	\$tax	Steuer
double	\$amount_brutto	Betrag
int	\$tstamp	Datum als Unix-Timestamp
int	\$createuser	Benutzer-ID, der Buchung erzeugt hat
int	\$createtime	Zeitstempel, an dem Buchung erzeugt wurde
int	\$chguser	Benutzer-ID, der Buchung zuletzt geändert hat
int	\$chgtime	Zeitstempel, an dem zuletzt geändert wurde

**Beispiel:**

```
<?php
try {
    $results = $api->getBookingRecord($booking_record_id);
}
catch (CRMException $e) {
    echo $e->getMessage();
}

foreach($results as $result) {
    foreach($result as $key => $val) {
        echo "$key => $val<br> \n";
    }
    echo "<hr/>";
}
}
```

## 1.6. getIdByExtId

(Version >= 4.2)

**Syntax:**

```
array getIdByExtId( string $external_id )
```

**Beschreibung**

Liefert interne ID(s) von Buchungen mit der übergebenen externen ID zurück.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

**Eigenschaften des zurückgegebenen *booking record ids* array:**

Type(Length)	Eigenschaft	Beschreibung
int	\$id	Task-Id

**Beispiel:**

```

<?php

try {
    $api      = crmapi::booking_records();
    $result = $api->getIdByExtId(12345);
}
catch (CRMException $e) {
    echo $e->getMessage();
}

foreach($result as $id) {
    echo $id."<br> ";
}

```

## 1.7. getAllExtIds

(Version >= 4.2)

### Syntax:

*array getAllExtIds( string \$external\_id )*

### Beschreibung

Liefert eine Liste aller externen IDs in Buchungen zurück.

Im Falle eines Fehlers wird eine CRMException mit mehr Fehlerdetails geworfen.

### Eigenschaften des zurückgegebenen *booking record ids* array:

Type(Length)	Eigenschaft	Beschreibung
int	\$id	externe Task-Id

### Beispiel:

```

<?php

try {
    $api      = crmapi::booking_records();
    $result = $api->getAllExtIds();
}
catch (CRMException $e) {
    echo $e->getMessage();
}

foreach($result as $id) {
    echo $id."<br> ";
}

```