

Projekt Bonusclub

Projekt Bonusclub

Inhalt

Coding Conventions

Tabellen und Spalten

bonus_customers

CUSTOMER_ID

name

surname

phone_number

date_of_birth

date_of_signup

email

card_no

bonus_tax_rates

TAX_RATE_ID

designation

percentage

bonus_articles

ARTICLE_ID

designation

unit

net_price

tax_rate_id

sold_units

bonus_positions

INVOICE_ID

ARTICLE_ID

unit_price

quantity

position

bonus_invoices

INVOICE_ID

customer_id

subtotal_net

purchase_timestamp

branch_office_id

bonus_branch_offices

BRANCH_OFFICE_ID

name

adress

postcode

bonus_postcodes

PLZ

name

Trigger

Neuberechnung der Zwischensumme

Berechnung der verkauften Artikel

Views

Trigger

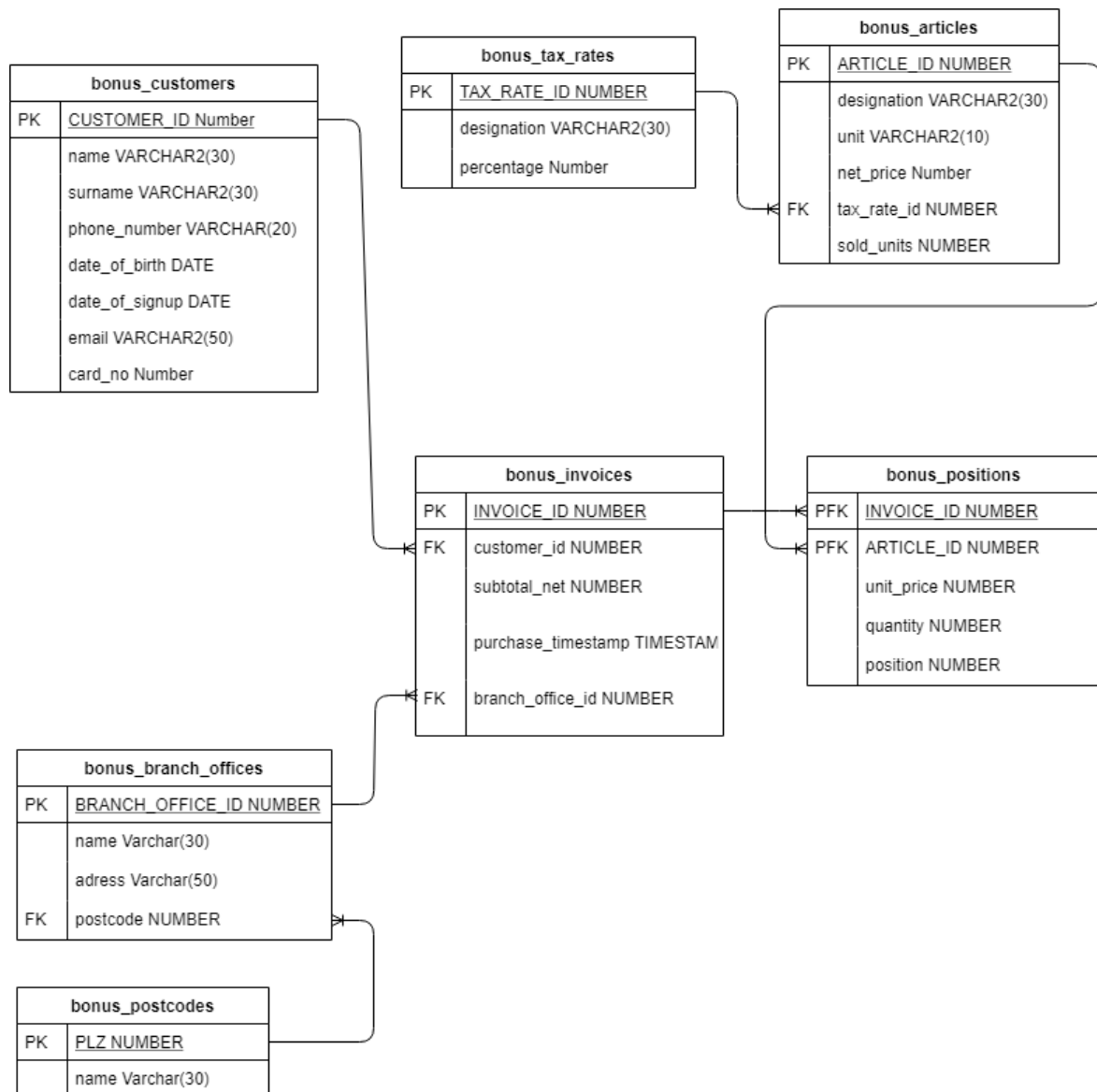
Calculate subtotal

- Delete Customer
- Update sold units
- Packages
 - pa_bonus_customers
 - sp_add_customer
 - Parameter
 - Rückgabewert
 - sp_update_customer
 - Parameter
 - Rückgabewert
 - f_get_customer_turnover_n
 - Parameter
 - Rückgabewert
 - f_get_customers_rc
 - Parameter
 - Rückgabewert
 - f_get_customers_rc
 - Parameter
 - Rückgabewert
 - Articles
 - P: Add
 - P: Update
 - P: List
 - pa_bonus_branch
 - f_get_branches
 - Parameter
 - Rückgabewert
 - f_get_branch
 - Parameter
 - Rückgabewert
 - pa_bonus_invoices
 - sp_add_invoice
 - Parameter
 - Rückgabewert
 - sp_update_invoice
 - Parameter
 - Rückgabewert
 - sp_add_article
 - Parameter
 - Rückgabewert
 - sp_delete_article
 - Parameter
 - Rückgabewert
 - f_get_invoices_rc
 - Parameter
 - Rückgabewert
 - f_get_invoice_rc
 - Parameter
 - Rückgabewert
 - f_get_invoice_articles_rc
 - Parameter
 - Rückgabewert
 - pa_bonus_stats
 - f_get_most_bought_article_rt
 - Parameter
 - Rückgabewert
 - f_get_customer_with_highest_turnover_n
 - Parameter

Rückgabewert
ps_reset_sold_units
Parameter
Rückgabewert
Description

Inhalt

Das Projekt Bonusclub besteht aus den beiden Projektmitgliedern Kurosh Mehdipour Moghaddam und Manuel Szecsenyi. Das Projekt soll ein vereinfachtes Kundenkartensystem sein. Das Kundeneinkaufsverhalten soll erfasst und analysiert werden. In der GUI sollen Kunden, Rechnungen, Artikel und Positionen verwaltet werden. Das Löschen von Kunden oder Rechnungen wird in der GUI aufgrund der gesetzlichen Aufbewahrungspflicht nicht möglich sein. Des Weiteren wird in der GUI das aufrufen der Statistiken möglich sein.



Coding Conventions

Es wurden keine zusätzlichen Coding Conventions erhoben. Es wird auf die der Angabe zurückgegriffen.

Tabellen und Spalten

bonus_customers

Name: bonus_customers

Inhalt: Stamminformationen der Kunden

Indizes: CUSTOMER_ID, card_no

CUSTOMER_ID

Name: Customer_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID des Kunden

name

Name: name

Datentyp: VARCHAR2(30)

Constraints: NOT NULL

Inhalt: Der Vorname eines Kunden

surname

Name: surname

Datentyp: VARCHAR2(30)

Constraints: NOT NULL

Inhalt: Der Nachname eines Kunden

phone_number

Name: phone_number

Datentyp: VARCHAR(20)

Constraints: /

Inhalt: Die Telefonnummer eines Kunden

date_of_birth

Name: date_of_birth

Datentyp: DATE

Constraints: NOT NULL

Inhalt: Das Geburtsdatum eines Kunden

date_of_signup

Name: date_of_signup

Datentyp: DATE

Constraints: NOT NULL

Inhalt: Das Registrierdatum eines Kunden

email

Name: email

Datentyp: VARCHAR2(50)

Constraints: NOT NULL

Inhalt: Die Emailadresse eines Kunden

card_no

Name: card_no

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Die Kartenummer eines Kunden

bonus_tax_rates

Name: bonus_tax_rates

Inhalt: Steuersätze der Artikel

Indizes: ?

TAX_RATE_ID

Name: TAX_RATE_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Steuersatzid

designation

Name: designation

Datentyp: VARCHAR2(30)

Constraints: NOT NULL

Inhalt: Bezeichnung des Steuersatzes

percentage

Name: percentage

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Prozentsatz, welcher dann später verrechnet wird

bonus_articles

Name: bonus_articles

Inhalt: Artikelstammdaten

Indizes: ?

ARTICLE_ID

Name: ARTICLE_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID des Artikels

designation

Name: designation

Datentyp: VARCHAR2(30)

Constraints: NOT NULL

Inhalt: Bezeichnung des Artikels

unit

Name: unit

Datentyp: VARCHAR2(10)

Constraints: NOT NULL

Inhalt: Einheit des Produktes (z.B. kg)

net_price

Name: net_price

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Nettopreis des Artikels

tax_rate_id

Name: tax_rate_id

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID des Steuersatzes

sold_units

Name: sold_units

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Verkaufte Menge

bonus_positions

Name: bonus_positions

Inhalt: Position des Artikels auf der Rechnung

Indizes: ?

INVOICE_ID

Name: INVOICE_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID der Rechnung

ARTICLE_ID

Name: ARTICLE_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID des Artikels

unit_price

Name: unit_price

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Stückpreis des Artikels

quantity

Name: quantity

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Anzahl des Artikels auf der Rechnung

position

Name: position

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Position des Artikels auf der Rechnung

bonus_invoices

Name: bonus_invoices

Inhalt: Rechnung

Indizes: ?

INVOICE_ID

Name: INVOICE_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID der Rechnung

customer_id

Name: customer_id

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID des Kunden

subtotal_net

Name: subtotal_net

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Zwischensumme der Rechnung

purchase_timestamp

Name: purchase_timestamp

Datentyp: TIMESTAMP

Constraints: NOT NULL

Inhalt: Gespeicherte Zeit der abgeschlossenen Rechnung

branch_office_id

Name: branch_office_id

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID der Filiale

bonus_branch_offices

Name: bonus_branch_offices

Inhalt: Filiale wo die Artikel verkauft werden

Indizes: ?

BRANCH_OFFICE_ID

Name: BRANCH_OFFICE_ID

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: ID der Filiale

name

Name: name

Datentyp: VARCHAR(30)

Constraints: NOT NULL

Inhalt: Name der Filiale

adress

Name: adress

Datentyp: VARCHAR(50)

Constraints: NOT NULL

Inhalt: Adresse der Filiale

postcode

Name: postcode

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Postleitzahl der Filiale

bonus_postcodes

Name: bonus_postcodes

Inhalt: Postleitzahlen der verschiedenen Filialen

Indizes: ?

PLZ

Name: PLZ

Datentyp: NUMBER

Constraints: NOT NULL

Inhalt: Postleitzahl der Filiale

name

Name: name

Datentyp: VARCHAR(30)

Constraints: NOT NULL

Inhalt: Name des Ortes

Trigger

Das Projekt beinhaltet zwei Trigger.

1. Neuberechnung der Zwischensumme
2. Aktualisieren der verkauften Stückzahl eines Produktes

Neuberechnung der Zwischensumme

Name: `tr_ar_dlu_update_subtotal`

Zweck: Die Zwischensumme einer Rechnung muss bei jedem neuen oder entfernten Artikel neu berechnet werden. Sonst ist sie inkorrekt.

Art: `After Delete, Insert, Update For Each Row`

Tabellen und Attribute: Es werden immer die Menge und Preise multipliziert. Bei einem `Insert` werden sie mit der alten Zwischensumme summiert. Bei einem `Delete` subtrahiert. Bei einem `Update` wird das alte Produkt subtrahiert und das neue addiert. Bei einem Wechsel der Rechnungs ID passiert dies in den jeweiligen Tabellen. **Attribute:** `UNIT_PRICE` und `QUANTITY` jeweils mit dem Präfix `:new` oder `:old`. **Tabellen** die verändert werden: `BONUS_INVOICES`

Der erste Ansatz war nach jeder Änderung alle Positionen der Rechnung in der Tabelle `BONUS_POSITIONS` durchzugehen und zu addieren. Leider führte das durch den Fehler 04091 nicht zum gewünschten Erfolg. Ein Trigger kann nämlich kein `SELECT`, `DELETE` oder `UPDATE` gegen eine Tabelle ausführen, wenn in dieser der Row Trigger ist. Deshalb dann der Ansatz mit addieren und subtrahieren der Zwischensummen.

Berechnung der verkauften Artikel

Name: `tr_ar_lu_update_sold_units`

Zweck: Um die Verkauften Einheiten eines Produktes dokumentieren zu können. Änderungen bei der Artikelanzahl werden auch berücksichtigt.

Art: `After Insert or Update, for each row`

Tabellen und Attribute: Es wird immer die Anzahl der Artikel zu den Verkauften Einheiten dazugezählt. Bei einem Insert wird die Anzahl des Artikels dem derzeitigen Stand der verkauften Einheiten hinzugefügt. Bei einem Update Befehl wird zuerst geprüft ob die verkauften Einheiten über 0 liegen. Falls dies erfüllt wird, dann wird der alte Wert abgezogen und der neue Wert daraufhin dazu addiert. Wenn dies nicht der Fall ist, wird der Wert nur dazu addiert.

Bei dem Trigger wurden die **Tabellen:** `BONUS_ARTICLES` und `BONUS_POSITIONS` verwendet.

Attribute: `QUANTITY` und `SOLD_UNITS`

Views

Trigger

Calculate subtotal

Delete Customer

(Before delete müssen Positionen und Rechnungen gelöscht werden.) oder in Invoices updaten zu einer anderen Zahl

Update sold units

Packages

pa_bonus_customers

Package name: `pa_bonus_customer`

Entwickler: Manuel Szecsenyi

Beschreibung: Ermöglicht das Aufrufen von Kundendaten.

sp_add_customer

Fügt einen neuen Kunden hinzu.

```
procedure sp_add_customer(v_name_in IN VARCHAR,  
                          v_surname_in IN VARCHAR,  
                          v_phone_number IN VARCHAR,  
                          d_date_of_birth_in DATE,  
                          v_email_in VARCHAR,  
                          n_card_no_in Number);
```

Parameter

`v_name_in IN VARCHAR` - Der Vorname des Kunden

`v_surname_in IN VARCHAR` - Der Nachname des Kunden

`v_phone_number IN VARCHAR` - Die Telefonnummer des Kunden

`d_date_of_birth_in DATE` - Das Geburtsdatum des Kunden

`v_email_in VARCHAR` - Die Email-Adresse des Kunden

`n_card_no_in Number` - Die Kartennummer des Kunden

Rückgabewert

Die Prozedur gibt nichts zurück.

sp_update_customer

Aktualisiert die Daten eines Kunden in der Datenbank

```
procedure sp_update_customer( n_customer_id_in IN NUMBER,  
                             v_name_in IN VARCHAR,  
                             v_surname_in IN VARCHAR,  
                             v_phone_number IN VARCHAR,  
                             d_date_of_birth_in DATE,  
                             v_email_in VARCHAR,  
                             n_card_no_in Number)
```

Parameter

`n_customer_id_in IN NUMBER` - Die ID der Person. Wird zur Identifizierung verwendet.

`v_name_in IN VARCHAR` - Der Vorname des Kunden

`v_surname_in IN VARCHAR` - Der Nachname des Kunden

`v_phone_number IN VARCHAR` - Die Telefonnummer des Kunden

`d_date_of_birth_in DATE` - Das Geburtsdatum des Kunden

`v_email_in VARCHAR` - Die Email-Adresse des Kunden

`n_card_no_in Number` - Die Kartennummer des Kunden

Rückgabewert

Die Prozedur gibt nichts zurück.

f_get_customer_turnover_n

Nimmt die KundenID entgegen und gibt den Gesamtumsatz den der Kunde in den Geschäften hinterlassen hat zurück.

```
function f_get_customer_turnover_n(n_CUSTOMER_ID_in IN NUMBER) RETURN NUMBER;
```

Parameter

`n_CUSTOMER_ID_in IN NUMBER` - Die ID des Kunden

Rückgabewert

`NUMBER` - Der Umsatz des Kunden. Bei einem Fehler wird `-1` zurück gegeben.

f_get_customers_rc

Gibt alle Kunden zurück.

```
function f_get_customers_rc RETURN SYS_REFCURSOR
```

Parameter

Keine Parameter.

Rückgabewert

`SYS_REFCURSOR` - Alle Kunden

f_get_customers_rc

Gibt einen Kunden zurück mit der angegeben ID.

```
function f_get_customers_rc(n_customer_id_in NUMBER) RETURN SYS_REFCURSOR
```

Parameter

`n_customer_id_in NUMBER` - Die KundenID

Rückgabewert

`SYS_REFCURSOR` - Kunde mit der angegebenen ID

Articles

P: Add

P: Update

P: List

pa_bonus_branch

Package name: `pa_bonus_branch`

Entwickler: Manuel Szecsenyi

Beschreibung: Ermöglicht das Aufrufen von Filialdaten.

f_get_branches

Gibt alle Filialen zurück.

```
function f_get_branches RETURN SYS_REFCURSOR
```

Parameter

Keine Parameter.

Rückgabewert

`SYS_REFCURSOR` - Alle abgespeicherten Filialen.

f_get_branch

Gibt eine die Filiale zurück mit der angegebenen ID.

```
function f_get_branch(n_branch_id_in IN NUMBER) RETURN SYS_REFCURSOR
```

Parameter

`n_branch_id_in NUMBER` - Eine Filialid

Rückgabewert

`SYS_REFCURSOR` - Die Filiale mit der angegebenen ID.

pa_bonus_invoices

Package name: `pa_bonus_invoices`

Entwickler: Manuel Szecsenyi

Beschreibung: Hinzufügen und aktualisieren von Rechnungen ist mit diesem Package möglich. Es können auch Artikel hinzugefügt und gelöscht werden. Aber Rechnungsdaten können abgerufen werden.

sp_add_invoice

Fügt eine neue Rechnung hinzu. Das System ermittelt selbständig die nächste Rechnungsnummer und speichert den aktuellen Zeitstempel ab. Die Rechnungen sind leer und haben Anfangs einen Nettobetrag von Null (0€).

```
PROCEDURE sp_add_invoice(  
    n_customer_id_in IN NUMBER,  
    n_branch_office_id_in IN NUMBER,  
)
```

Parameter

`n_customer_id_in` IN NUMBER - Der Kunde zu dem die Rechnung abgespeichert werden soll.

`n_branch_office_id_in` - Die Filial-ID in der der Einkauf getätigt wurde.

Rückgabewert

Kein Rückgabewert

sp_update_invoice

Aktualisiert die wichtigsten Metainformationen einer Rechnung. Die Rechnungs-ID wird nicht verändert.

```
PROCEDURE sp_update_invoice(  
    n_invoice_id_in IN NUMBER,  
    n_customer_id_in IN NUMBER,  
    d_purchase_timestamp_in IN TIMESTAMP,  
    n_branch_office_id_in IN NUMBER  
)
```

Parameter

`n_invoice_id_in` - Die Rechnungs-ID der Rechnung welche geändert werden soll. Die ID kann nicht verändert werden.

`n_customer_id_in` - Der Kunde zu dem die Rechnung gehört.

`n_purchase_timestamp_in` - Der Timestamp des Einkaufs.

`n_branch_office_id_in` - Die Filial-ID in der der Einkauf getätigt wurde.

Rückgabewert

Kein Rückgabewert

sp_add_article

Fügt einen Artikel mit der angegebenen Stückzahl zu einer Rechnung hinzu. Es wird der im Artikel hinterlegte Preis verrechnet.

```
PROCEDURE sp_add_article(  
    n_invoice_id_in IN NUMBER,  
    n_article_id_in IN NUMBER,  
    n_quantity_in IN NUMBER  
)
```

Parameter

`n_invoice_id_in` - Die Rechnungs-ID der Rechnung.

`n_article_id_in` - Die Artikel-ID eines Artikels

`n_quantity_in` - Wie viel Stück des Produkte soll verrechnet werden?

Rückgabewert

Kein Rückgabewert.

sp_delete_article

Der angegebene Artikel auf der Rechnung wird entfernt. Die Rechnungspositionen werden nachgerückt.

```
PROCEDURE sp_delete_article(  
    n_invoice_id_in IN NUMBER,  
    n_article_id_in IN NUMBER  
)
```

Parameter

`n_invoice_id_in` - Die Rechnungs-ID der Rechnung.

`n_article_id_in` - Die Artikel-ID eines Artikels

Rückgabewert

Kein Rückgabewert.

f_get_invoices_rc

Gibt alle Rechnungen zurück.

```
function f_get_invoices_rc RETURN SYS_REFCURSOR
```

Parameter

Keine Parameter.

Rückgabewert

`rc_cursor_out` - Alle vorhandenen Rechnungen

f_get_invoice_rc

Gibt die Metadaten einer Rechnung zurück.

```
function f_get_invoice_rc(n_invoice_id_in in NUMBER) RETURN SYS_REFCURSOR
```

Parameter

`n_invoice_id_in` - Die Rechnungs-ID der Rechnung.

Rückgabewert

`rc_cursor_out` - Eine Rechnung mit der angegebenen ID

f_get_invoice_articles_rc

Gibt alle Positionen einer Rechnung zurück

```
function f_get_invoice_articles_rc(n_invoice_id_in in NUMBER) RETURN  
SYS_REFCURSOR
```

Parameter

`n_invoice_id_in` - Die Rechnungs-ID der Rechnung.

Rückgabewert

`rc_cursor_out` - Alle vorhandenen Positionen

pa_bonus_stats

Package name: `pa_bonus_stats`

Entwickler: Manuel Szecsenyi

Beschreibung: Mit diesem Package können verschiedene Statistken ausgelesen und zurückgesetzt werden.

f_get_most_bought_article_rt

Gibt den Artikel zurück mit dem größten `sold_units` Attribut. Zählt **nicht** die Häufigkeit eines Artikels in den Positionen. Diese Statistik kann mit dem zurücksetzen der verkauften Einheiten eines Produktes verändert werden.

```
function f_get_most_bought_article_rt RETURN BONUS_ARTICLES%ROWTYPE;
```

Parameter

Keine Parameter

Rückgabewert

`rt_article_row_out` - Ein Eintrag aus der Tabelle Artikel.

f_get_customer_with_highest_turnover_n

Gibt den Kunden mit dem größten Umsatz zurück.

```
function f_get_customer_with_highest_turnover_rt RETURN NUMBER;
```

Parameter

Diese Funktion nimmt keine Parameter entgegen.

Rückgabewert

`n_customer_id_out` `NUMBER` - Die Id des Kunden mit dem größten Umsatz.

ps_reset_sold_units

```
PROCEDURE ps_reset_sold_units(  
    n_article_id_in in NUMBER  
)
```


Parameter

`n_article_id_in` - Der Artikel welcher zurückgesetzt werden soll.

Rückgabewert

Kein Rückgabewert.

Description

Setzt die `sold_units` des Artikels auf null (0).