

# Order Webservice

- [Create / easy.affiliate REST-API \(Orders\)](#)
- [Update / easy.affiliate REST-API \(Orders\)](#)

# Create / easy.affiliate REST-API (Orders)

## ? Einleitung

Um neue Orders automatisiert im easy.affiliate-System anzulegen, steht ein REST-konformer Webservice zur Verfügung. Über die HTTP-Methode **POST** können neue Transaktionen übermittelt werden. Jeder User erhält hierfür einen Authentifizierungs-Token und eine Login-ID, die im Frontend einsehbar sind.

---

## ? Authentifizierung

Für den Zugriff sind folgende Header erforderlich:

Header	Beschreibung	Typ
<b>Content-Type</b>	Muss auf <code>application/json</code> gesetzt sein	String
<b>X-Network-ID</b>	Netzwerk-ID: meist <code>-1</code> für mandantenübergreifende API	Integer
<b>X-Auth-Token</b>	Dein API-Token (im Frontend sichtbar)	String
<b>X-Auth-ID</b>	Deine Login-ID (im Frontend sichtbar)	Integer

---

## ? Endpunkte

- **Admin:** `https://SUBDOMAIN.de/ws/V6/admin/JSON/Orders`
  - **Advertiser:** `https://SUBDOMAIN.de/ws/V6/advertiser/JSON/Orders`
- 

## ? Beispiel (cURL)

```

curl -X POST \
  -H "Content-Type: application/json" \
  -H "X-Network-ID: -1" \
  -H "X-Auth-Token: ADMIN_APIUSER_TOKEN" \
  -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" \
  -d '{
    "ordertoken": "testorder001",
    "campaign_id": 1,
    "trigger_id": 2,
    "status": 1,
    "turnover": 199.99,
    "commission": 20.00
  }' \
  https://SUBDOMAIN.de/ws/V6/admin/JSON/Orders

```

## ? Body-Parameter

Parameter	Beschreibung	Datentyp	Pflichtfeld
<b>ordertoken</b>	Eindeutiger Identifier für die Order. Muss systemweit eindeutig sein.	String	<input type="checkbox"/> Ja
<b>campaign_id</b>	ID der Kampagne, der die Order zugeordnet wird.	Integer	<input type="checkbox"/> Ja
<b>trigger_id</b>	ID des Triggers, der bei Nutzung von <code>ordertoken</code> zwingend angegeben werden muss.	Integer	<input type="checkbox"/> Ja
<b>status</b>	Status der Order: <code>0</code> = offen, <code>1</code> = bestätigt, <code>2</code> = storniert	Integer	<input type="checkbox"/> Ja
<b>turnover</b>	Bestellwert in Kampagnenwährung	Float	<input type="checkbox"/> Ja
<b>original_turnover</b>	Bestellwert in Fremdwährung	Float	Nein
<b>commission</b>	Provisionswert in Kampagnenwährung. Kann übergeben werden, falls selbst berechnet	Float	Nein
<b>cancel_reason</b>	Angabe eines Stornogrundes bei Status = 2 (storniert)	String	Nein

---

# ? Hinweise

## Wichtige Implementierungsdetails:

- **ordertoken:** Muss **eindeutig** sein – idealerweise eine Kombination aus Zeitstempel, Shop-ID oder externer Ordnernummer.
- **trigger\_id:** Wenn du diese nicht kennst, frage im Frontend deine Kampagnenkonfiguration ab.
- **Bulk-Aktionen:** Du kannst auch mehrere Orders in einem Array auf einmal anlegen.
- **Updates:** Wird eine Order mit gleichem `ordertoken` bereits im System gefunden, kann ein `PUT-Request` genutzt werden, um sie zu aktualisieren.

# Update / easy.affiliate REST-API (Orders)

## ? Einleitung

Um Orders automatisiert im **easy.affiliate-System** bearbeiten zu können, stellt die easy Marketing GmbH eine Webservice-API zur Verfügung. Diese ermöglicht es, bestehende Orders im Nachgang zu bearbeiten, sobald sie den Validierungsprozess durchlaufen.

Jedem User stehen hierfür ein **Authentifizierungs-Token** und eine **Login\_id** zur Verfügung, die über das Frontend abgerufen werden können.

- **User-ID:** Wo ist die User-ID hinterlegt? (Siehe Frontend-Profil)
- **Access-Token:** Wo ist der Access-Token hinterlegt? (Siehe API-Einstellungen)

## Endpunkte

Zielgruppe	URL
Admin	<code>https://SUBDOMAIN.de/ws/V6/admin/JSON/Orders</code>
Advertiser	<code>https://SUBDOMAIN.de/ws/V6/advertiser/JSON/Orders</code>

## Beispiele

### Beispiel mit cURL

```
curl -X PUT \  
  -H "Content-Type: application/json" \  
  -H "X-Network-ID: -1" \  
  -H "X-Auth-Token: ADMIN_APIUSER_TOKEN" \  
  -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" \  
  -d '{ "campaign_id": 1, "id": 12345, "status": 1, "turnover":100.00, "commission":12.34 },{'
```

```
"campaign_id": 1, "ordertoken":"123a456b", "status": 2, "turnover": 0.00 }]' \  
https://DOMAIN/ws/V6/admin/JSON/Orders
```

# Codebeispiel

Mit der folgenden Methode können mehrere Orders geupdatet werden. Ein Beispielaufruf sieht folgendermaßen aus:

## Headers

Variable	Wert / Bedeutung
Content-Type	application/json
X-Network-ID	-1
X-Auth-Token	ADMIN_APIUSER_TOKEN
X-Auth-ID	ADMIN_APIUSER_LOGIN_ID

## Body

```
[  
  {  
    "id": "12345",  
    "campaign_id": 1,  
    "status": 1,  
    "turnover": 10.00,  
    "commission": 0.99  
  },  
  {  
    "ordertoken": "123a456b",  
    "campaign_id": 1,  
    "trigger_id": 1,  
    "status": 2,  
    "turnover": 9.99,  
    "cancel_reason": "Storno Grund"  
  }  
]
```

# Variablenerläuterung

## Headers

Variable	Bedeutung	Datentyp
<b>Content-Type</b>	Der Content-Type des Requests	String
<b>X-NETWORKID</b>	Hier wird die ID des Mandanten eingetragen. Wenn nur ein Mandant vorhanden ist oder mandantenübergreifend gearbeitet wird, muss der Wert "-1" eingetragen werden.	Integer
<b>X-AUTH-TOKEN</b>	Hier wird der API-Authentifizierungs-Token des Admin Nutzers hinterlegt.	String
<b>X-AUTH-ID</b>	Hier wird die ID des Admin Nutzers hinterlegt.	Integer

## Body

“ Im nachfolgenden werden die Standard-Parameter aufgelistet. Es ist aber über die API möglich, alle zur Verfügung stehenden Parameter zu updaten.

Parameter	Erklärung	Datentyp	Pflichtfeld
<b>id</b>	Hier wird die Transaktionsid hinterlegt. Aufbau: <code>"id": "ID"</code>	String	Ja (oder <code>ordertoken</code> )
<b>ordertoken</b>	Hier wird der Ordertoken hinterlegt. Aufbau: <code>"ordertoken": "ORDERTOKEN"</code>	String	Ja (oder <code>id</code> ).*
<b>campaign_id</b>	Hier wird die Campaign_id hinterlegt. Aufbau: <code>"campaign_id": CAMPAIGN_ID</code>	Integer	Ja
<b>trigger_id</b>	Hier wird die Trigger ID hinterlegt. Aufbau: <code>"trigger_id": TRIGGER_ID</code>	Integer	Nein, nur falls <code>ordertoken</code> genutzt wird.

Parameter	Erklärung	Datentyp	Pflichtfeld
<b>status</b>	Hier wird der Status hinterlegt. <b>Werte:</b> <code>-5</code> (ausstehende Nachbuchung), <code>0</code> (offen), <code>1</code> (bestätigt), <code>2</code> (storniert)	Integer	Nein
<b>turnover</b>	Hier wird der Bestellwert der Transaktion in Kampagnenwährung hinterlegt. Aufbau: <code>"turnover":TURNOVER</code>	Float	Nein
<b>original_turnover</b>	Hier wird der Bestellwert der Transaktion in Fremdwährung hinterlegt. Aufbau: <code>"original_turnover":TURNOVER_IN_FREMDWÄHRUNG</code>	Float	Nein
<b>commission</b>	Hier wird die Provisionshöhe in Kampagnenwährung hinterlegt, falls Sie die Commission selbst berechnen möchten.	Float	Nein
<b>cancel_reason</b>	Übergeben Sie den Stornogrund. Aufbau: <code>"cancel_reason":"Stornogrund"</code>	String	Nein

\*) **Hinweis:** Wird der `ordertoken` genutzt, ist neben der `campaign_id` auch die `trigger_id` ein Pflichtfeld.