



Tipps & Tricks

Dokumentation der SEMCO API



Inhaltsverzeichnis

Ausgangspunkt.....	3
Grundlagen.....	4
Service-URL	4
Datenformat.....	4
Authentifizierung.....	5
Metadaten	5
Paginierung	6
Sortierung.....	6
Filterung.....	6
Status-Codes und Fehlermeldungen	7
Caching.....	8
Swagger UI.....	9
Zugriff auf Swagger UI	9
Überblick.....	9
Aufrufe der öffentlichen Ressourcen (API-Key Authentifizierung).....	10
Aufrufe der nicht öffentlichen Ressourcen (Authentifizierung mit SEMCO Benutzer)	13
Erstellen von Online-Buchungen	14
Privatbuchung.....	15
Firmenbuchung durch einen Teilnehmer	16
Firmenbuchung durch einen Ansprechpartner mit zwei Teilnehmern	17
Anpassbarkeit.....	19

Ausgangspunkt

SEMCO verfügt über eine RESTful API, die es Ihnen ermöglicht Daten aus SEMCO auszulesen und auch zurückzuschreiben.

Meistens wird die **SEMCO API** von Webagenturen verwendet, um Kursdaten auszulesen, auf einer Webseite in einem beliebigen Design darzustellen und Online-Buchungen anzubieten.

Damit Sie und Ihre IT sich mit den Möglichkeiten unserer **SEMCO API** vertraut machen können, wurde in diesem Dokument alles Wissenswerte zusammengefasst.

Grundlagen

Die **SEMCO** API ist ein sogenannter RESTful Webservice. Jede Ressource wird über eine eindeutige URI angesprochen.

Welche Operationen mit Ressourcen durchgeführt werden, wird mittels der entsprechenden HTTP-Methoden bestimmt.

Folgende Operationen sind grundsätzlich möglich:

GET	Ressourcen lesen
POST	Ressourcen erstellen
PUT	Ressourcen bearbeiten
PATCH	Ressourcen spezifisch/teilweise bearbeiten

Service-URL

Die Service URL hat den folgenden Aufbau:

```
https://{SEMCO-URL}/api/v1/{resource}/{id}
```

{SEMCO-URL}	Die URL der SEMCO-Instanz, z.B. https://mustermann-trainings.semcosoft.com
{RESOURCE}	Ressource, die angesprochen wird
{ID}	ID der konkreten Ressource

Datenformat

Die **SEMCO** API unterstützt das JSON-Format. Das Format wird im HTTP-Header folgendermaßen angegeben:

```
Accept: application/json
```



Grundsätzlich gilt: die per HTTP-Request übergebenen Daten müssen UTF-8 encodiert sein. Sämtliche Responses werden auch in UTF-8 geliefert.

Authentifizierung

Die Authentifizierung in **SEMCO** kann in zwei Stufen erfolgen. Um auf die API grundsätzlich zugreifen zu können, wird ein API-Benutzer benötigt. Mit diesem Benutzer wird auf die öffentlichen Daten wie veröffentlichte Kurse oder Dictionary-Daten wie beispielsweise Kurskategorien oder Kursarten zugegriffen.

Zusätzlich zum API-Benutzer ist es möglich, sich mit einem beliebigen **SEMCO** Benutzer zu authentifizieren. Diese Art der Authentifizierung ermöglicht den Zugriff auf weitere nicht öffentliche Daten wie Buchungen und Dokumente. Dabei werden die Berechtigungen des Benutzers sowie die Einschränkungen entsprechend des Benutzertyps – wie Teilnehmer oder Firmenansprechpartner – berücksichtigt.

AUTHENTIFIZIERUNG MIT API-KEY

In **SEMCO** wird der API-Schlüssel bei einem Benutzer vom Typ API hinterlegt. Dieser API-Schlüssel wird im HTTP-Header folgendermaßen angegeben:

```
X-APIKey: {API-Schlüssel}
```

AUTHENTIFIZIERUNG MIT SEMCO BENUTZER

Für diese Authentifizierungsart wird die sogenannte HTTP Basic Authentication verwendet. Dabei wird der Benutzername und das Passwort des entsprechenden Benutzers base64-encodiert im HTTP-Header angegeben.

BEISPIEL:

Der Benutzer *mmustermann* mit dem Passwort *topsecret* wird folgendermaßen authentifiziert:

```
Authorization: Basic bW11c3Rlcm1hbm46dG9wc2VjcmV0
```

Dabei ist *bW11c3Rlcm1hbm46dG9wc2VjcmV0* die Zusammensetzung des Benutzernamen und Passworts getrennt mit einem Doppelpunkt (*mmustermann:topsecret*) und encodiert mit base64.

Metadaten

Mit Hilfe der Ressource *metadata* ist es möglich, Metadaten einer Ressource in Erfahrung zu bringen:

- ▶ *description* – Beschreibung einer Ressource
- ▶ *fields* – Auflistung der in der Ressource enthaltenen Felder mitsamt den folgenden Informationen:
 - *type* integer/number/string/boolean/object
 - *isWritable* true/false. Gibt an, ob das Feld bei einem schreibenden Zugriff gesetzt sein darf.
 - Weitere Eigenschaften abhängig vom Typ wie *format*, *maxLength*, *properties* (bei Object)
- ▶ *idField* – Gibt an, welches Feld das Schlüsselfeld ist.

► *filters* – Auflistung der zur Verfügung stehenden Filter mitsamt den folgenden Informationen:

- *schema* – hat das gleiche Format wie die Einträge im Bereich *fields*
- *label* – die Bezeichnung des Filters
- *options* – eine Liste der möglichen Werte/Bezeichnungen

Paginierung

Die Anzahl der Datensätze per Request (bei einer Ressourcen-Liste) kann über den Parameter *per_page* angegeben werden, durch die einzelnen Seiten kann mit dem Parameter *page* navigiert werden.

BEISPIEL:

```
https://mustermann-trainings.semcosoft.com/api/v1/classes?per_page=30&page=2
```

Um die Anzahl der möglichen Seiten zu ermitteln, ist in der Response die Gesamtanzahl der Einträge enthalten: *total*.

Sortierung

Mittels des URL-Parameters *order_by* kann eine Sortierreihenfolge angegeben werden. Die Angabe der Sortierung besteht aus dem Feldnamen und der Richtung der Sortierung: ASC für aufsteigend und DESC für absteigend. Sortierreihenfolgen können kombiniert werden, indem mehrere Felder kommasetrennt angegeben werden.

BEISPIEL:

```
https://mustermann-trainings.semcosoft.com/api/v1/classes?order_by=class_startdate+ASC,class_token+DESC
```

Filterung

Bei GET Zugriffen können Filter angewendet werden. Die für eine Ressource zur Verfügung stehenden Filter können mittels der *metadata* Ressource ermittelt werden. Es ist möglich, nach jedem Feld einer Ressource zu filtern. Außerdem stehen meistens zusätzliche Filter zur Verfügung, die eine spezielle Filterlogik anwenden. Meistens fängt die Bezeichnung von einem speziellen Filter mit einem Ausrufezeichen an. Beispiel: *!class*.

Filter werden mittels URL-Parameter angegeben:

```
https://mustermann-trainings.semcosoft.com/api/v1/classes?class_startdate%5B0%5D%5Bv%5D=2021-02-1&class_startdate%5B0%5D%5Bo%5D=%3E%3D
```

Filter-Parameter URL decodiert:

```
?class_startdate[0][v]=2021-02-01&class_startdate[0][o]=>=
```

Des Weiteren ist es möglich, einen gleichen Filter mehrmals anzuwenden:

```
https://mustermann-trainings.semcosoft.com/api/v1/classes?class_startdate%5B0%5D%5Bv%5D=2021-02-01&class_startdate%5B0%5D%5Bo%5D=%3E%3D&class_startdate%5B1%5D%5Bv%5D=2021-03-01&class_startdate%5B1%5D%5Bo%5D=%3C%3D
```

Filter-Parameter Url decodiert:

```
?class_startdate[0][v]=2021-02-01&class_startdate[0][o]=>=&class_startdate[1][v]=2021-03-01&class_startdate[1][o]=<=
```

Status-Codes und Fehlermeldungen

Die folgenden allgemeinen Status Codes können potentiell bei allen Aufrufen auftreten und müssen entsprechend behandelt werden.

STATUS-CODE	BESCHREIBUNG
200 OK	Der Request wurde erfolgreich ausgeführt.
400 Bad request	Der Request wurde nicht angenommen. Details stehen im Response Body. Meistens kommt dieser Fall bei POST Requests vor, wenn die Validierung der Daten fehlschlägt.
401 Not authorized	Es wurde versucht ohne Angabe des API-Keys auf eine Ressource zuzugreifen.
403 Access denied	Der Zugriff auf diese Ressource ist mit der verwendeten HTTP-Methode nicht erlaubt.

STATUS-CODE	BESCHREIBUNG
404 Nothing found	Unter dieser URL kennt der Server keine Ressource. Der Eintrag mit der angegebenen ID wird nicht gefunden. Die Abfrage liefert keine Einträge zurück.
405 Action not allowed	Die verwendete HTTP-Methode existiert bei der verwendeten Ressource nicht.
406 Invalid content type requested	Der im Accept-Header angefragte Content-Type wird nicht unterstützt.
412 Precondition failed	Ein Pflichtfeld ist bei einem PATCH Zugriff leer.
500 Server error	Es ist ein serverseitiger Fehler aufgetreten. Im Response Body ist ein Fehler-Code angegeben. Wenn das Problem nicht nachvollziehbar ist, kontaktieren Sie den Support und geben Sie dabei den Fehler-Code an.

Wenn ein Fehler auftritt, werden neben dem HTTP-Status Code Fehlermeldungen mit Bezug zu den Ressource-Feldern in einem Array angegeben. Jeder Index des Arrays entspricht einer Feldbezeichnung. Fehlermeldungen, die keinen direkten Bezug zu einem Feld haben, werden mit dem Index `__global` angegeben.

BEISPIELE:

```
{"errors":{"__global":["Bad API key"]}}
{"errors":{"salutation_id":["Das Feld ist ein Pflichtfeld"],"firstname":["Das Feld ist ein Pflichtfeld"],"lastname":["Das Feld ist ein Pflichtfeld"]}}
```

Caching

Es wird empfohlen die Ergebnisse von GET Zugriffen zu cachen. Jede Ressource gibt vor, wie lange die Daten im Cache bleiben können. Diese Angabe steht im HTTP-Header des Responses:

```
Cache-Control: max-age=60, public
```

Die Angabe `max-age=60` bedeutet, dass die Daten für 60 Sekunden gecacht werden können.

Swagger UI

SEMCO erzeugt on-the-fly ein Open-API Dokument, das die Spezifikation der API beinhaltet.

Ab Version 5.4 ist in SEMCO das Tool Swagger UI eingebaut. Das Tool generiert eine interaktive Dokumentation der API gemäß der Open-API Spezifikation.

Swagger UI ermöglicht unter anderem einen Request an die API abzuschicken und zeigt die Response an.

Zugriff auf Swagger UI

Swagger UI befindet sich in Ihrem System unter dem Pfad `/api/swagger`. Zum Beispiel:

```
https://mustermann-trainings.semcosoft.com/api/swagger
```

Beim Zugriff darauf, werden Sie aufgefordert einen Benutzernamen und Passwort einzugeben. Die Anmeldung erfolgt mit folgenden Zugangsdaten:

Benutzername: api

Passwort ist der API-Schlüssel Ihres API-Benutzers

Überblick

Swagger UI listet alle zur Verfügung stehenden Ressourcen auf. Zu jeder Ressource werden Zugriffsarten angegeben und unter anderem welche HTTP-Methoden bei der entsprechenden Ressource möglich sind.

Beim Klick auf die entsprechende Zugriffsart einer Ressource werden die möglichen Parameter aufgelistet, die bei einem Request verwendet werden können. Dabei werden neben den Standardparametern für Paginierung und Sortierung auch alle möglichen Filterparameter aufgelistet.



Einige Ressourcen benötigen Pflichtparameter. Diese werden mit einem roten Stern markiert. Zum Beispiel `class_id` bei der Ressource `class-schedules`.

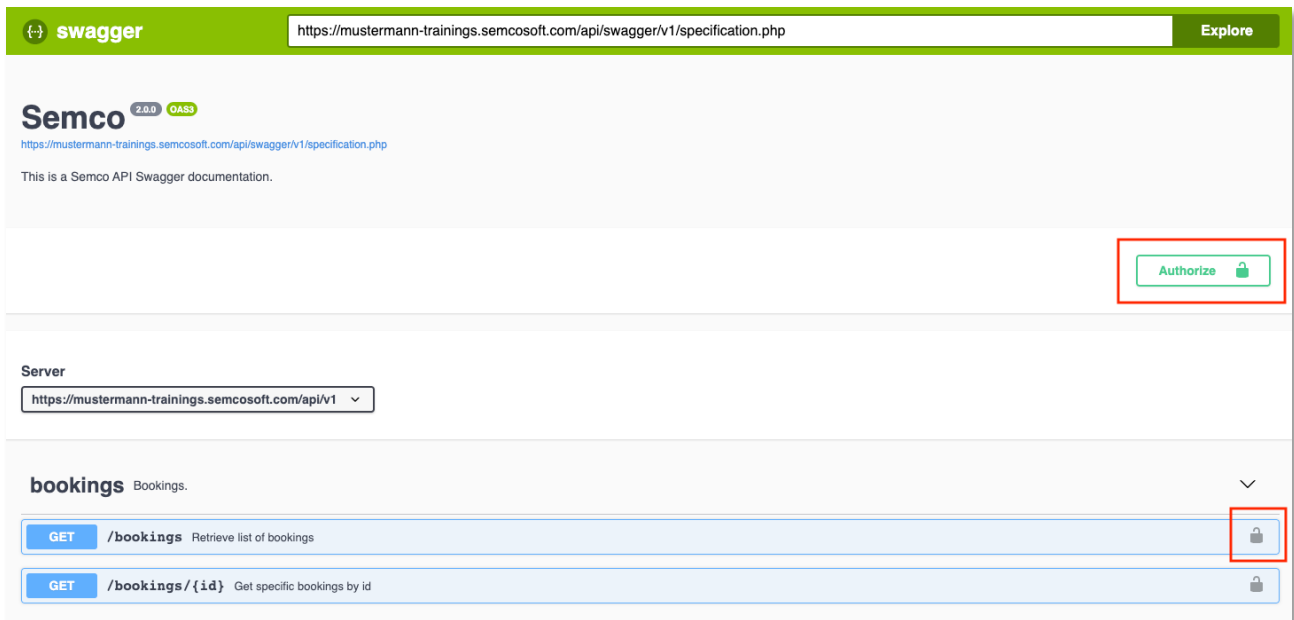
Unterhalb der Parameterliste werden Beispiele für Responses für jeden Status-Code angezeigt. Der Status-Code 200 beinhaltet dabei ein Beispiel für die positive Rückmeldung der Ressource.

Nach der Ressourcenliste werden die sogenannten Models aufgelistet. Beim Klick auf das entsprechende Model klappt die Spezifikation der Datenfelder auf, die im Model enthalten sind.

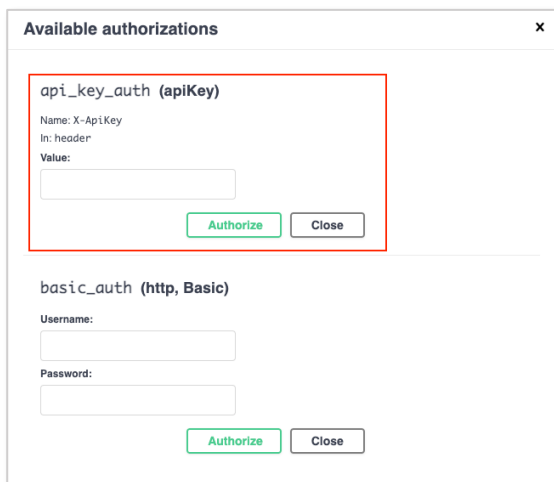
Aufrufe der öffentlichen Ressourcen (API-Key Authentifizierung)

Um eine Ressource mit Swagger UI aufzurufen, müssen die entsprechenden Zugangsdaten eingegeben werden. Diese werden anschließend beim Aufruf verwendet.

Klicken Sie dazu auf die Schaltfläche [Authorize](#) beziehungsweise auf das Schloss-Icon bei einer Ressource.



Im eingblendeten Fenster geben Sie den API-Schlüssel ein und klicken Sie auf [Authorize](#) und anschließend auf [Close](#).



Nun können Sie mit dem hinterlegten API-Schlüssel auf die öffentlichen Ressourcen zugreifen.

Klicken Sie dazu zum Beispiel auf die GET Zugriffsart der Ressource [classes](#) und anschließend auf die Schaltfläche [Try it out](#).

classes Classes. ⌵

GET /classes Retrieve list of classes 🔒

Parameters Try it out

Name	Description
order_by string (query)	Order by clause (e.g. 'booking_id' or '_created DESC, _modified DESC')
per_page integer (query)	Number of classes per page



Nach dem Klick auf *Try it out* werden Eingabefelder zu den Parametern eingeblendet. Wenn Sie Angaben in diesen Feldern machen, werden Sie in den Request miteinbezogen.

Scrollen Sie zu der Schaltfläche *Execute* und klicken Sie darauf. Die API wird aufgerufen und Sie sehen folgende Informationen:

Execute

Clear

1. Es wird ein curl-Befehl angezeigt, der auf der Kommandozeile unter Linux ausgeführt werden kann. Der Befehl beinhaltet alle angegebenen Parameter inklusive des API-Keys:

Responses

Curl

```
curl -X GET "https://mustermann-trainings.semco-software.com/api/v1/classes" -H "accept: application/json" -H "X-APIKey: TOPSECRET"
```

2. Es wird der Response Body angezeigt, der die zurückgelieferten Daten beinhaltet:

Request URL

```
https://mustermann-trainings.semco-software.com/api/v1/classes
```

Server response

Code	Details
200	Response body <pre>{ "total": "19", "data": [{ "class_id": "1", "class_token": "10001", "class_short_desc": "Erste Hilfe Grundkurs (Webinar)", "course_long_desc": null, "course_title": "Erste Hilfe Grundkurs (Webinar)", "class_desc": null, "class_startdate": "2021-05-10", "class_enddate": "2021-05-11", "class_min_participants": "8", "class_max_participants": "20", "level_short_desc": "Online-Training", "level_custom01": null, "hour_base": "1", "quota": "2.00", "course_price": "699.00", "currency_symbol": "EUR", "max_participants": "20", "min_participants": "8", "free_places": "15", "branch_id": "1", "branch_token": "SEMCO AKADEMIE", "branch_company_name": "SEMCO Akademie", "branch_short_desc": null, "department_id": "1" }] }</pre> Response headers <pre>cache-control: max-age=60, public connection: Keep-Alive content-type: application/json date: Mon, 08 Feb 2021 12:52:55 GMT keep-alive: timeout=5, max=100 server: Apache/2.4.25 (Debian) transfer-encoding: chunked</pre> Download

Aufrufe der nicht öffentlichen Ressourcen (Authentifizierung mit SEMCO Benutzer)

Um auf eine Ressource mit einem **SEMCO** Benutzer zuzugreifen, müssen die Zugangsdaten des Benutzers zusätzlich zum API-Key angegeben werden.

Klicken Sie dazu auf die Schaltfläche **Authorize** beziehungsweise auf das Schloss-Icon bei einer Ressource.

Im eingeblendeten Fenster geben Sie den API-Schlüssel im Bereich **api_key_auth** ein und klicken Sie auf **Authorize**. Im Bereich **basic_auth** geben Sie die Zugangsdaten des Benutzers ein und klicken Sie auf **Authorize**. Abschließend klicken Sie auf **Close**.

Nun können Sie mit den hinterlegten Authentifizierungsdaten auf die nicht öffentlichen Ressourcen zugreifen.

Klicken Sie dazu zum Beispiel auf die GET Zugriffsart der Ressource **bookings** und anschließend auf die Schaltfläche **Try it out**.

Scrollen Sie zu der Schaltfläche **Execute** und klicken Sie darauf. Die API wird aufgerufen und Sie sehen folgende Informationen:

The screenshot shows the Swagger UI interface with the 'Execute' tab selected. The 'Responses' section displays the following information:

Execute **Clear**

Responses

Curl

```
curl -X GET "https://mustermann-trainings.semcosoft.com/api/v1/bookings" -H "accept: application/json" -H "X-APIKey: TOPSECRET" -H "Authorization: Basic dGltby5nrW5+qENlbWVvc29mdC5jb206awSpddfdGlbhbaX"
```

Request URL

```
https://mustermann-trainings.semcosoft.com/api/v1/bookings
```

Server response

Code	Details
200	Response body <pre>{ "total": "1", "data": [{ "enrollment_id": "1", "class_id": "1", "class_token": "10001", "class_short_desc": "Erste Hilfe Grundkurs (Webinar)", "class_desc": null, "e_course_start": "2021-05-10", "e_course_end": "2021-05-11", "booking_id": "1", "booking_state_id": "CONFIRMED", "booking_state_desc": "Bestätigte Buchung", "documents": [{ "entity_id": "1", "document_type_id": "99", "document_title": "Rechnung RE21-00001", "document_identifier": "", "document_title_full": "Rechnung RE21-00001", "access_token": "78b62004726d83182a9a33c1737b5a081fb1fdcc30dc41b395e42652969dbc" }, { "entity_id": "1", "document_type_id": "2", "document_title": "Anmeldebestätigung", "document_identifier": "" }] }] }</pre> Response headers <pre>cache-control: max-age=60, private connection: Keep-Alive content-type: application/json date: Mon, 08 Feb 2021 13:02:12 GMT keep-alive: timeout=5, max=100 server: Apache/2.4.25 (Debian) transfer-encoding: chunked</pre> Download

Erstellen von Online-Buchungen

Die Daten zu einer Online-Buchung werden in einem flachen Array übermittelt. In **SEMCO** werden einzelne Felder den entsprechenden Entitäten wie Online-Buchung, Buchung, Teilnehmer, Firma, Firmenansprechpartner zugeordnet.

SEMCO unterstützt drei Arten der Online-Buchungen:

- ▶ Privatbuchung
- ▶ Firmenbuchung durch einen Teilnehmer
- ▶ Firmenbuchung durch einen Ansprechpartner mit mehreren Teilnehmern

Abhängig davon, welche Art der Online-Buchung gewünscht ist, müssen unterschiedliche Felder gesetzt werden. Während bei einer Privatbuchung nur Teilnehmerdaten notwendig sind, müssen bei einer Firmenbuchung neben den Teilnehmerdaten auch Firmendaten übergeben werden.

Folgende Tabelle gibt einen Überblick über die besonderen Felder einer Online-Buchung:

FELD	ERKLÄRUNG
online_order_course_id	ID des gebuchten Kurses. Ressource zum Nachschlagen: <i>classes</i> .
online_order_type_id	Art der Online-Buchung. Folgende Ausprägungen sind möglich: <i>PERSONAL</i> , <i>COMPANY</i> , <i>COMPANY_MULTI</i> .
online_order_ip	IP-Adresse des Buchenden. Kann mit PHP folgendermaßen ermittelt werden: <i>getenv('HTTP_CLIENT_IP') ?:</i> <i>getenv('HTTP_X_FORWARDED_FOR') ?:</i> <i>getenv('HTTP_X_FORWARDED') ?:</i> <i>getenv('HTTP_FORWARDED_FOR') ?:</i> <i>getenv('HTTP_FORWARDED') ?:</i> <i>getenv('REMOTE_ADDR')</i> .
online_order_payment_state_id	Muss den Wert <i>INVOICE</i> haben.
online_order_payment_option_id	Muss den Wert <i>INVOICE</i> haben.
Anrede	Felder wie <i>salutation_id</i> oder <i>contact_person_salutation_id</i> sind Verweise und können in der Ressource <i>salutations</i> nachgeschlagen werden.
Land	Felder wie <i>country_of_residence_id</i> oder <i>company_country</i> sind Verweise und können in der Ressource <i>countries</i> nachgeschlagen werden.

Privatbuchung

Nachfolgend finden Sie die Daten einer Beispiel-Online-Buchung als JSON-Array:

```
{
  "online_order_course_id": "1",
  "online_order_type_id": "PERSONAL",
  "online_order_ip": "123.123.123.123",
  "online_order_payment_state_id": "INVOICE",
  "online_order_payment_option_id": "INVOICE",
  "salutation_id": "1",
  "academic_title": "Dr.",
  "firstname": "Max",
  "lastname": "Mustermann",
  "work_email": "mmustermann@semcosoft.com",
  "work_phone": "01234568",
  "work_mobile": "01601234567",
  "street": "Berliner Alle 100",
  "zip": "86123",
  "city": "Augsburg",
  "country_of_residence_id": "DE",
  "create_account": "1",
  "booking_comment": ""
}
```

BEISPIEL-AUFRUF MIT CURL:

```
curl -X POST "https://mustermann-trainings.semcosoft.com/api/v1/online-orders" -H
"accept: application/json" -H "X-APIKey: SECRET" -H "Content-Type:
application/json" -d
"{\"online_order_course_id\": \"1\", \"online_order_type_id\": \"PERSONAL\", \"online_
order_ip\": \"123.123.123.123\", \"online_order_payment_state_id\": \"INVOICE\", \"onl
ine_order_payment_option_id\": \"INVOICE\", \"salutation_id\": \"1\", \"academic_title
\": \"Dr.\", \"firstname\": \"Max\", \"lastname\": \"Mustermann\", \"work_email\": \"mmus
termann@semcosoft.com\", \"work_phone\": \"01234568\", \"work_mobile\": \"01601234567\
\", \"street\": \"Berliner Alle
100\", \"zip\": \"86123\", \"city\": \"Augsburg\", \"country_of_residence_id\": \"DE\", \
\"create_account\": \"1\", \"booking_comment\": \"\"}"
```

Firmenbuchung durch einen Teilnehmer

Nachfolgend finden Sie die Daten einer Beispiel-Online-Buchung als JSON-Array:

```
{
  "online_order_course_id": "1",
  "online_order_type_id": "COMPANY",
  "online_order_ip": "123.123.123.123",
  "online_order_payment_state_id": "INVOICE",
  "online_order_payment_option_id": "INVOICE",
  "salutation_id": "1",
  "academic_title": "Dr.",
  "firstname": "Max",
  "lastname": "Mustermann",
  "work_email": "mmustermann@semcosoft.com",
  "work_phone": "01234568",
  "work_mobile": "01601234567",
  "company_name": "Beispielfirma AG",
  "company_address_line1": "Maximilianstra\u00dfe 200",
  "company_address_line2": "81456",
  "company_address_line3": "M\u00fcnchen",
  "company_country": "DE",
  "booking_comment": ""
}
```

BEISPIEL-AUFRUF MIT CURL:

```
curl -X POST "https://mustermann-training.com/api/v1/online-orders" -H "accept: application/json" -H "X-ApiKey: TOPSECRET" -H "Content-Type: application/json" -d '{"online_order_course_id":"1","online_order_type_id":"COMPANY","online_order_ip":"123.123.123.123","online_order_payment_state_id":"INVOICE","online_order_payment_option_id":"INVOICE","salutation_id":"1","academic_title":"Dr.","firstname":"Max","lastname":"Mustermann","work_email":"mmustermann@semcosoft.com","work_phone":"01234568","work_mobile":"01601234567","company_name":"Beispielfirma AG","company_address_line1":"Maximilianstra\u00dfe 200","company_address_line2":"81456","company_address_line3":"M\u00fcnchen","company_country":"DE","booking_comment":""}'
```


Firmenbuchung durch einen Ansprechpartner mit zwei Teilnehmern

Nachfolgend finden Sie die Daten einer Beispiel-Online-Buchung als JSON-Array:

```
{
  "online_order_course_id": "1",
  "online_order_type_id": "COMPANY_MULTI",
  "online_order_ip": "123.123.123.123",
  "online_order_payment_state_id": "INVOICE",
  "online_order_payment_option_id": "INVOICE",
  "company_name": "Beispielfirma AG",
  "company_address_line1": "Maximilianstra\u00dfe 200",
  "company_address_line2": "81456",
  "company_address_line3": "M\u00fcnchen",
  "company_country": "DE",
  "contact_person_salutation_id": "1",
  "contact_person_firstname": "Susanne",
  "contact_person_lastname": "Gro\u00df",
  "contact_person_academic_title": "",
  "contact_person_position": "Personalleiterin",
  "contact_person_phone": "0123456789",
  "contact_person_email": "sgross@semcosoft.com",
  "company_alternative_invoice_recipient": "",
  "booking_comment": "",
  "entries": [
    {
      "participants": {
        "salutation_id": "1",
        "firstname": "Max",
        "lastname": "Mustermann",
        "work_email": "mmustermann@semcosoft.com"
      },
      "bookings": [
      ]
    },
    {
      "participants": {
        "salutation_id": "2",
        "firstname": "Erika",
        "lastname": "Musterfrau",
        "work_email": "emusterfrau@semcosoft.com"
      },
      "bookings": [
      ]
    }
  ]
}
```

Die Besonderheit dieser Art der Online-Buchung besteht darin, dass neben den Daten zur Firmenbuchung auch Teilnehmer- und Buchungsdaten der Teilnehmerbuchungen im *entries*-Array enthalten sind.

BEISPIEL-AUFRUF MIT CURL:

```
curl -X POST "https://mustermann-trainings.semcosoft.com/api/v1/online-orders" -H
"accept: application/json" -H "X-APIKey: SECRET" -H "Content-Type:
application/json" -d
"{\"online_order_course_id\":\"1\", \"online_order_type_id\":\"COMPANY_MULTI\", \"on
line_order_ip\":\"123.123.123.123\", \"online_order_payment_state_id\":\"INVOICE\",
\"online_order_payment_option_id\":\"INVOICE\", \"company_name\":\"Beispielfirma
AG\", \"company_address_line1\":\"Maximilianstraße
200\", \"company_address_line2\":\"81456\", \"company_address_line3\":\"München\", \"
company_country\":\"DE\", \"contact_person_salutation_id\":\"1\", \"contact_person_f
irstname\":\"Susanne\", \"contact_person_lastname\":\"Groß\", \"contact_person_acade
mic_title\":\"\", \"contact_person_position\":\"Personalleiterin\", \"contact_person
_phone\":\"0123456789\", \"contact_person_email\":\"sgross@semcosoft.com\", \"compan
y_alternative_invoice_recipient\":\"\", \"booking_comment\":\"\", \"entries\": [{\"pa
rticipants\": {\"salutation_id\":\"1\", \"firstname\":\"Max\", \"lastname\":\"Musterm
ann\", \"work_email\":\"mmustermann@semcosoft.com\"}, \"bookings\": [], {\"participan
ts\": {\"salutation_id\":\"2\", \"firstname\":\"Erika\", \"lastname\":\"Musterfrau\",
\"work_email\":\"emusterfrau@semcosoft.com\"}, \"bookings\": []}]}"
```

Anpassbarkeit

Der [SEMCO](#) Kurskalender und das [SEMCO](#) Kundenportal nutzen die API für den Datenaustausch. In unserem Best-Practice-System ist die API für den Einsatz mit dem [SEMCO](#) Kurskalender und dem [SEMCO](#) Kundenportal vorkonfiguriert.

Es besteht jedoch die Möglichkeit die API kundenindividuell anzupassen. Bei einzelnen Ressourcen können Felder aufgenommen werden. Bei bestimmten Ressourcen können schreibende Zugriffe aktiviert werden. Es ist auch möglich, kundenindividuelle Ressourcen hinzuzufügen, die zum Beispiel Daten eines Berichts zurückliefern.

Die Anpassung der API ist nur durch einen [SEMCO](#) Berater möglich.



SEMCO Software Engineering GmbH

Hauptsitz

Ellimahdstr. 40

89420 Höchstädt

Tel.: +49 9074 799 2903-0

Fax: +49 9074 799 2903-9

E-Mail: info@semcosoft.com

www.semcosoft.com