

Odoo REST API

Model Configuration in Odoo

By default, none of the models available in the API, and they need to be enabled one by one by administrator.

Go to **Settings > Technical > Models** > search for model (e.g. **sale.order**) > open model.

Then go to the **API Settings** tab:

The screenshot shows the Odoo Model Configuration screen for the 'sale.order' model. The 'API Settings' tab is selected. Several fields have annotations:

- REST API**: A checked checkbox with a yellow highlight. A red annotation text: "Need to tick this box for models required in API If not ticked, API will respond with \"model not allowed\" error".
- Delete Domain**: A field containing "[('state', '=', 'draft')]" with a yellow highlight. A red annotation text: "Delete Domain is to specify conditions when records CAN be deleted".
- Include Computed Fields**: A section with a red arrow pointing to it. A red annotation text: "Some fields are computed, if need to include, can do it here".
- Include IDs in POST response**: A field containing "partner_id". A red annotation text: "These are fields to be included in GET responses By default, all fields included, but it's recommended to select required fields for better API performance and less info noise".
- API Fields**: A table with a red border. It lists various fields with their labels, types, and settings. The columns are: Field Name, Field Label, Field Type, Required, Readonly, and Type. The table includes rows for client_order_ref, date_order, id, margin, name, order_line, and origin.

Type to enter a caption.

1. Tick **Rest API** to enable the model (minimum requirement)
2. Specify **Delete Domain** if you need. Format is [('field_name', '=', value)]

3. Specify **Include Computed Fields** if required (comma-separated words). By default, no computed fields included as they're not stored in the database; but you can still get them in the API responses by adding their labels here .

4. Specify fields that you want **Include IDs in POST responses**. By default, POST response is minimal, but you can include there fields you might need (comma-separated words).

5. Select **fields** you want to have in the API. By default, everything is included, but it can look pretty noisy as there're a lot of technical fields that rarely needed. If you specify some fields here, API will work faster and responses will look cleaner.

Note, that for some "child" fields you might need to enable API too. For example, if you want to update sale order lines, then you need to include **sale.order.line** model too.

User Configuration in Odoo

Auth in the API is via access tokens that can be generated for each user in **Settings > Users&Companies > Users** > search for user > open.

Then navigate to the **API** tab:

The screenshot shows the Odoo user configuration page for a user named 'solnet'. The top navigation bar includes 'Related Partner' and 'Administrator'. Below the user name, there are tabs for 'Access Rights', 'Preferences', and 'API'. The 'API' tab is highlighted with a yellow background. A red arrow points to the 'GENERATE NEW API AUTH TOKEN' button, with the text 'Click to generate a new login token' above it. Another red arrow points to the 'fa1982ba-f10b-4a5c-aace-3eb9ce385632' token value, with the text 'Use this token for /login endpoint' below it. The text 'New tab under user config' is overlaid in red at the top right of the screenshot area.

Type to enter a caption.

When you want to revoke a user's access, just click on **Generate New API Auth Token**, and the old one will be revoked. Use the token generated for `/login` endpoint, then use **access_token** from the response as **Access-Token** to all other endpoints.

If no token or token's invalid -- API returns 401 Unauthorised error with a message.

API usage

Example GET endpoints for Sale Order:

- GET /api/v1/login
- GET /api/v1/model/sale.order
- GET api/v1/model/sale.order/<Record ID> (for specific order)
- GET /api/v1/model/sale.order/?limit=10&fields=name&order=amount_untaxed
- GET /api/v1/model/sale.order/?domain=name,=,SO27807

Optional params for GET "search" endpoints:

- **limit** to limit a number of records (default is 20)
- **fields** if some specific fields required (overwrites model config)
- **order** if some specific ordering required
- **domain** if specific domain has to be applied

Also accepts PUT, DELETE and POST calls – see Postman Collection for more examples.