Quotas Sub-Object v5

The following API calls are currently available for the Quotas object:

- GET LIST
- GET QUOTAS
- CREATE QUOTAS

- UPDATE QUOTAS
- DELETE QUOTAS
- EXAMPLES

The Quota Sub Object is intended for use with the Segmented Quotas feature in Alchemer. This object does *not* function with Overall Quotas. If one is looking to manipulate an Overall Quota, refer to the Survey Object help page, using

use the CREATE or UPDATE calls with the parameter options[overall_quota].

GET LIST - Get a list of all of your survey quotas.

https://api.alchemer.com/v5/survey/123456/quotas

Parameters	Example	Required
Authentication Credentials	api_token=abcd12345&api_token_secret=abcd12345	True

GET QUOTAS - Get information about a specified quota.

https://api.alchemer.com/v5/survey/123456/quotas/1

Parameters	Example	Required
Authentication Credentials	api_token=abcd12345&api_token_secret=abcd12345	True

CREATE QUOTAS - Create a new quotas.

https://api.alchemer.com/v5/survey/123456/quotas?_method=PUT

Parameters	Example	Required
Authentication Credentials	api_token=abcd12345&api_token_secret=abcd12345	True
name	This is my quota name	True
description	This is my quota description	False
limit	50	True
distributed	true	False

UPDATE QUOTAS - Update a specified quota.

https://api.alchemer.com/v5/survey/123456/quotas/12345?_method=POST

Parameters	Example	Required
Authentication Credentials	api_token=abcd12345&api_token_secret=abcd12345	True
name	This is my quota name	False
description	This is my quota description	False
limit	50	False
distributed	false	False
groups	URL encoded JSON file - **see examples**	False

Groups JSON Examples and Reference

In order to specify groups you will need to send a JSON file* with your rules.

*Depending on your implementation, you many need to convert the JSON to string format and URL encode it.

Each rule has the following parameters:

input_value

This is the type of field you are referencing. *Most often, this will be the Question ID.*

If you wish to specify groups based on something other than the a question this will be equivalent to the left dropdown menu in the logic builder in the UI. Field types can be found by referencing the below list or by inspecting element within the logic builder in the UI.

Possible Values for input_value

- Survey Question: {Question ID}
- Time Taken On Survey: 15_timetaken
- Current Date: 6_date
- Is Mobile (true/false): 21_mobile
- IP Address: 5_ip
- Country: 8_country
- Region: 9_region
- Postal Code: 19_postal
- City: 10_city
- Contact ID: 13_id

- Email Address: 13_emailaddress
- First Name: 13_firstname
- Last Name: 13_lastname
- Organization: 13_organization
- Department: 13_department
- Division: 13_division

- Team: 13_team
- Group: 13_group
- Role: 13_role
- Is Manager: 13_isManager
- Home Phone: 13_homephone
- Fax Number: 13_fax
- Business Phone: 13_workphone
- Mailing Address: 13_street
- State/Region: 13_region
- Postal Code: 13_postal
- Country: 13_country
- Title: 13_jobtitle
- Custom 1-10: 13_custom1-10
- Mailing Address 2: 13_suite
- City: 13_city

---

operator

This is the operator used in the logic rule.

Possible Values for operator

- is exactly equal to: 4
- is not exactly equal to: 22
- is one of the following answers: 12
- us not one of the following answers: 5
- greater than: 6
- greater than or equal to: 7
- less than: 9
- less than or equal to: 8
- Date after or equal to: 23

- Date before or equal to: 24
- is answered: 20
- is not answered: 21
- matches regex pattern: 14
- does not match regex pattern: 15
- is true: 26
- is false: 27
- contains: 10
- is always true: 25

---

answers_type

This is the type of data that is contained in the field specified in input_type.

Possible Values for answers_type

- Open-text Field: 2
- Input Field (Radio, Checkboxes, Drop-down Menus, etc.): 17

---

answers_values

These are the actual values you wish to filter by.

Possible Values for answers_values

- QID-OPTIONID - i.e. 27-10021 for question with QID 27 and answer option with OptionID 10021
- String - open-text questions and system data such as city, region, country, IP, etc.

---

next_operator

This is the next operator you wish to use to group your conditions.

Possible Values for next_operator **(distributed=false)**

- "and"
- "or"

Example JSON file for the groups parameter with distributed=false (_method=PUT):

Depending on your implementation, you many need to convert the JSON to string format and URL encode it.

```json
[
  {
    "rules": [
      {
        "input_value": "26",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "26-10019"
        ],
        "next_operator": "and"
      },
      {
        "input_value": "27",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "27-10021"
        ],
        "next_operator": ""
      }
    ],
    "next_operator": "or"
  },
  {
    "rules": [
      {
        "input_value": "19_postal",
        "operator": "4",
        "answers_type": "2",
        "answers_values": "82334",
        "next_operator": ""
      }
    ],
    "next_operator": "and"
  },
  {
    "rules": [
      {
        "input_value": "10_city",
        "operator": "4",
        "answers_type": "2",
        "answers_values": "Sinclair",
        "next_operator": "or"
      },
      {
        "input_value": "8_country",
        "operator": "4",
        "answers_type": "2",
        "answers_values": "USA",
        "next_operator": "or"
      },
      {
        "input_value": "5_ip",
        "operator": "4",
        "answers_type": "2",
        "answers_values": "192.169.0.1",
        "next_operator": ""
      }
    ],
    "next_operator": "and"
  },
  {
    "rules": [
```

```
    rules : [
      {
        "input_value": "13_region",
        "operator": "4",
        "answers_type": "2",
        "answers_values": "WY",
        "next_operator": ""
      }
    ],
    "next_operator": ""
  }
]
```

Example JSON file for the groups parameter with distributed=true (_method=PUT):

Depending on your implementation, you many need to convert the JSON to string format and URL encode it.

```
[
  {
    "rules": [
      {
        "input_value": "26",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "26-10019"
        ],
        "rule_percentage": "80"
      },
      {
        "input_value": "26",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "26-10020"
        ],
        "rule_percentage": "20"
      }
    ]
  },
  {
    "rules": [
      {
        "input_value": "27",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "27-10021"
        ],
        "rule_percentage": "90"
      },
      {
        "input_value": "27",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "27-10022"
        ],
        "rule_percentage": "10"
      }
    ]
  }
]
```

Example JSON file for the groups parameter with distributed=false (_method=POST):

Depending on your implementation, you many need to convert the JSON to string format and URL encode it.

```
[
  {
    "id": "58891b4ea7b14",
    "rules": [
      {
        "id": "58891b4ea7b63",
        "input_value": "20",
        "operator": "12",
        "answers_type": "17",
        "answers_values": [
          "20-10021"
        ]
      }
    ]
  }
]
```

Note: id value (highlighted above) is required to update quota rules using the POST method.

DELETE QUOTA - Delete a specified quota.

https://api.alchemer.com/v5/survey/123456/quotas/12345?_method=DELETE

Parameters	Example	Required
Authentication Credentials	api_token=abcd12345&api_token_secret=abcd12345	True

Back to the Top