### Search Log Schema Fields

Searches field names or display names from existing log schemas using keywords for reference when creating log schemas. This operation requires administrator role.

#### HTTP Request

GET /api/sonar/log-schema-assist/fields

cURL Example

curl -H "Authorization: Bearer <API\_KEY>" \ "https://HOSTNAME/api/sonar/log-schema-assist/fields?locale=en&type=field\_name&keywords=ip"

Request Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key | Required | Type | Description | Note |
| locale | Y | String | Language | One of en, ko, zh |
| limit | N | 32-bit integer | Maximum count | Default: 10, Minimum: 0 |
| type | Y | String | Search type | One of field\_name, display\_name |
| keywords | Y | String | Search keyword | Search string for the field specified by type |

#### Successful Response

{ "total\_count": 2, "schema\_names": { "session": { "en": "Session", "ko": "세션" } }, "schema\_fields": [ { "schema\_code": "session", "name": "src\_ip", "display\_name": "Source IP", "type": "IP", "ordinal": 1 }, { "schema\_code": "session", "name": "src\_port", "display\_name": "Source Port", "type": "PORT", "ordinal": 2 } ]}

Response Fields

|  |  |  |
| --- | --- | --- |
| Field | Type | Description |
| total\_count | 32-bit integer | Total count matching search criteria |
| schema\_names | Object | Multilingual name mapping by schema code |
| schema\_fields | Array | Log schema field list |
| schema\_fields.schema\_code | String | Schema code |
| schema\_fields.name | String | Field name |
| schema\_fields.display\_name | String | Display name |
| schema\_fields.type | String | Field type. One of DATE, STRING, IP, PORT, SHORT, INT, LONG, FLOAT, DOUBLE, BOOL, COUNTRY, MD5, SHA1, SHA256, URL, DOMAIN |
| schema\_fields.ordinal | 32-bit integer | Field display order (optional) |

#### Error Responses

locale: unsupported locale

HTTP status code 400 response

{ "error\_code": "invalid-argument", "error\_msg": "unsupported-locale: ru"}

type: invalid value

HTTP status code 400 response

{ "error\_code": "invalid-argument", "error\_msg": "invalid-type: wrong-parameter"}

keywords: parameter missing

HTTP status code 400 response

{ "error\_code": "invalid-argument", "error\_msg": "keywords-required"}

limit: wrong type

HTTP status code 400 response

{ "error\_code": "invalid-argument", "error\_msg": "'limit' parameter should be int type"}

limit: negative value

HTTP status code 400 response

{ "error\_code": "invalid-argument", "error\_msg": "'limit' must be greater than or equal to 0."}

permission: no schema viewing permission

HTTP status code 500 response

{ "error\_code": "illegal-state", "error\_msg": "no-permission"}