### Get Log Schema Fields

Retrieves the field list for a specified log schema.

#### HTTP Request

GET /api/sonar/log-schemas/:code/fields

Request using cURL

curl -H "Authorization: Bearer <API\_KEY>" \ "https://HOSTNAME/api/sonar/log-schemas/session/fields?locale=en"

Request Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key | Required | Type | Description | Note |
| code | O | String | Code | Schema identifier |
| field | X | String | Field name search pattern | Search by string contained in field\_name |
| locale | X | String | Language | One of en, ko, zh. Defaults to session locale |
| limit | X | 32-bit integer | Maximum count | Load all fields if field not specified, default: 20 if specified |
| offset | X | 32-bit integer | Number of records to skip | Default: 0 |
| keywords | X | String | Comprehensive search keywords | Search against field\_name, display\_name, and display\_name\_trans (localized) |

**Note: Difference between `field` and `keywords` parameters**

* field: Searches only field names (e.g., field=ip → src\_ip, dst\_ip, etc.)
* keywords: Searches field name + display name + localized display name (e.g., keywords=IP → src\_ip, Source IP, 출발지IP, etc.)

#### Success Response

Without `field` parameter

Retrieves all fields in the schema.

{ "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 } ]}

With `field` parameter

Searches for fields that contain a specific string in the field name. Example: field=ip

curl -H "Authorization: Bearer <API\_KEY>" \ "https://HOSTNAME/api/sonar/log-schemas/session/fields?field=ip"

{ "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": "dst\_ip", "display\_name": "Destination IP", "type": "IP", "ordinal": 3 } ]}

* **total\_count** (32-bit integer): Total number of records matching search criteria (returned only when field parameter is present)
* **schema\_names** (Map): Localized names by schema code (returned only when field parameter is present)
* **schema\_fields** (Array): Log schema field list
* **schema\_code** (String): Schema code
* **name** (String): Field name
* **display\_name** (String): Display name
* **type** (String): Type. One of DATE, STRING, IP, PORT, SHORT, INT, LONG, FLOAT, DOUBLE, BOOL, COUNTRY, MD5, SHA1, SHA256, URL, DOMAIN
* **ordinal** (32-bit integer): Field display order

#### Error Responses

No privilege to view log schema

HTTP status code 500.

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

Log schema not found

HTTP status code 500.

{ "error\_code": "illegal-state", "error\_msg": "cannot get log schema {code}"}

Unsupported locale specified

HTTP status code 400.

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

offset, limit values are not integers

HTTP status code 400.

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

offset, limit values are negative

HTTP status code 400.

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