sumologic-documentation/docs/search/search-query-language/group-aggregate-operators/values.md at 7b53abb04b0e84a5189b3ef4320a95c27c2a5a2f · SumoLogic/sumologic-documentation

github.com/SumoLogic/sumologic-documentation/blob/7b53abb04b0e84a5189b3ef4320a95c27c2a5a2f/docs/search/search-query-language/group-aggregate-operators/values.md

---

id: values

title: values Grouping Operator

sidebar_label: values

---

import useBaseUrl from '@docusaurus/useBaseUrl';

The `values` operator provides all the distinct values of a field. This allows you to quickly identify and understand all the values a field has in your data. Additionally, you have the option to group by other fields of interest.

Syntax

```sql
```

values(<field>) [by <group_by_fields>] [as <field_name>]

```

### Response Field

The response field separates each value with a new line character and places them in lexicographical order as follows:

* Numbers before letters

* Numbers sorted in ascending based on the value of the first digit

* Letters sorted in alphabetical order

* Uppercase before lowercase letters

This is an example of a response field with IP addresses:

<img src={useBaseUrl('img/search/searchquerylanguage/group-aggregate-operators/values-operator-response-field-example.png')} alt="Example of a response field with IP addresses" style={{border: '1px solid gray'}} width="150

" />

### Limitation

* The first 100 distinct values are returned for a field.

## Examples

### Operational Analytics

To identify all IP addresses by region:

```

_sourceCategory=Labs/*

| parse regex "(?<ip_address>\b\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})"

| values(ip_address) by region

```

To identify all IP addresses and namespaces by region:

```

_sourceCategory=Labs/*

| parse regex "(?<ip_address>\b\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})"

| values(ip_address) as val_ip, values(namespace) as val_namespace by region

To identify all sources by error type in my stack that logged an error in the last 24 hours:

```sql

_sourceCategory=prod01*

| parse regex "(?i)(?<log_level>WARN|CRITICAL|ERROR|FATAL)"

| toUppercase(log_level)
| _sourceCategory as sc

| count as errors, values(sc) by log_level

```

To identify users that logged in from more than one country in the last 24 hours with a list of countries logged in from:

```sql

_sc=org-service “login”

| parse username

| geolookup country on ip=login_ip

| count_distinct(country), values(country) by username

| where count_distinct > 1

```

Security Analytics

To know if my services have interacted with any known IOC threats.

```sql

...| values(IOC) by src_ip
```

To understand what ports were scanned or communicated over by one

`src_ip`.

```sql
_source="PatchingInfo" and _collector="AWS SecurityHub Non Prod"
| json field=_raw "port_name" as ports
| json field=_raw "src_ip" as src_ip
| values(ports) by ami
```