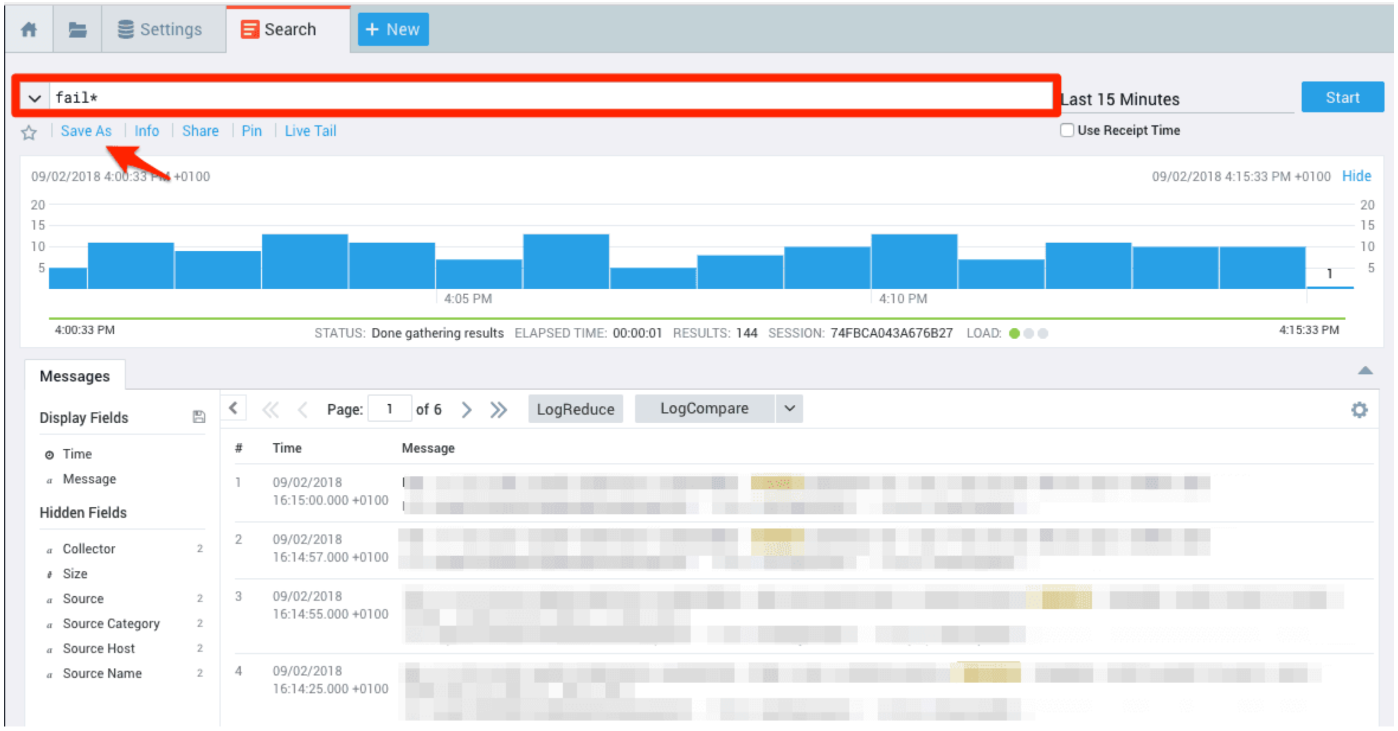
1.1.1 About Search Basics.md

github.com/aniket0609/Sumo_Logic_basic/blob/main/1.1.1 About Search Basics.md



STATUS: Done gathering results ELAPSED TIME: 00:00:01 RESULTS: 144 SESSION: 74FBCA043A676B27

About Search Basics

Our Search Syntax is based on a funnel or "pipeline" concept. The wide mouth of the funnel begins with all your current Sumo Logic data, and you narrow the funnel by entering keywords and operators separated by pipes (`|`). Each operator acts on the results from the previous operator so that you can progressively filter and pinpoint your search until you find exactly what you're looking for.

Let See How to search data using the Basic Search Mode in Sumo Logic.

In the Search tab, a search query is typically formatted something like this:

```
keyword search | parse | where | group-by | sort | limit
```

Start with a basic search:

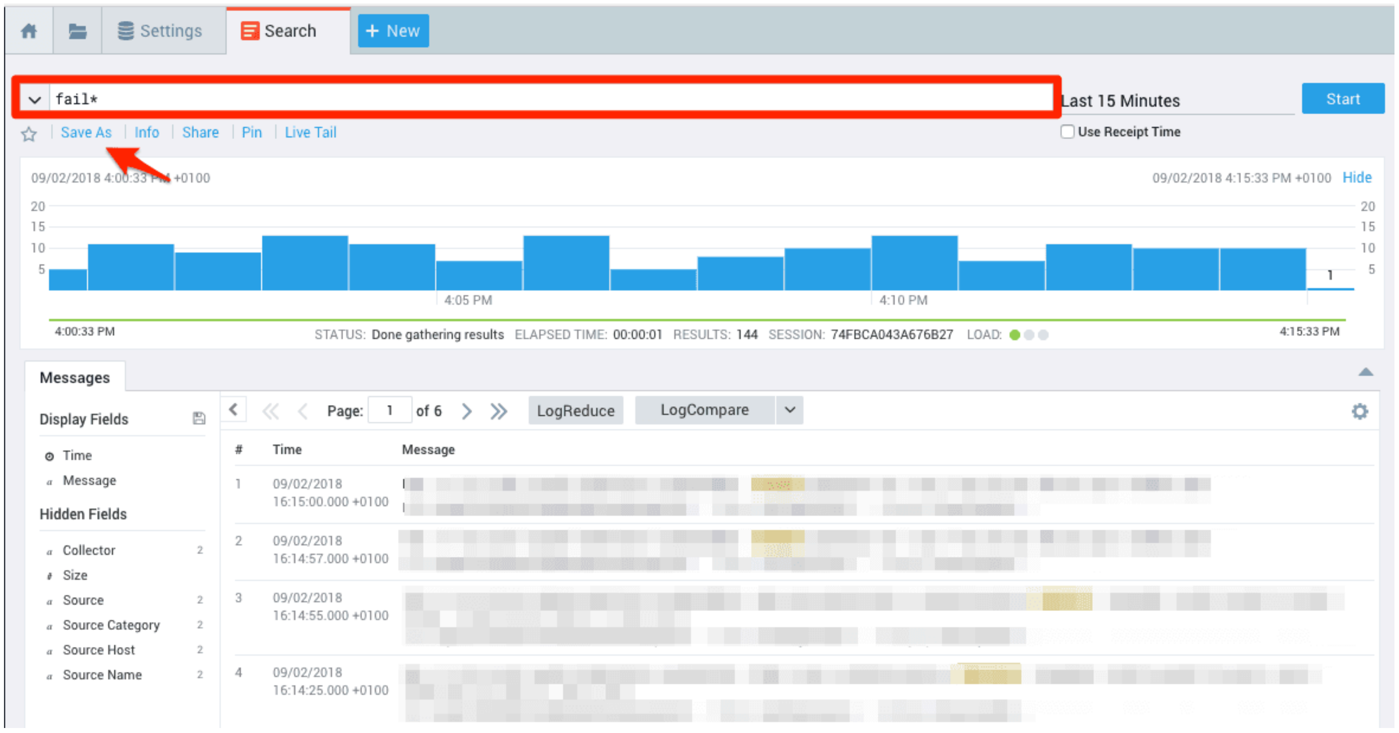
1.Click `+ New` button in the tab bar and select `Log Search`.

2.Enter a simple key term like `"error"` in the search field, or type an asterisk wildcard `(*)` to find all messages. Hit `Enter` or `click Start`.

3.Sumo Logic returns all the **log entries** containing the search term in the Messages tab below the **histogram**. Review a slightly more complex search query to see how queries are formed.

Step 1: Click `+ New` button

Step 2: write in Log Search field and click `Start`



```
fail*
```

09/02/2018 4:00:33 PM +0100 — 09/02/2018 4:15:33 PM +0100

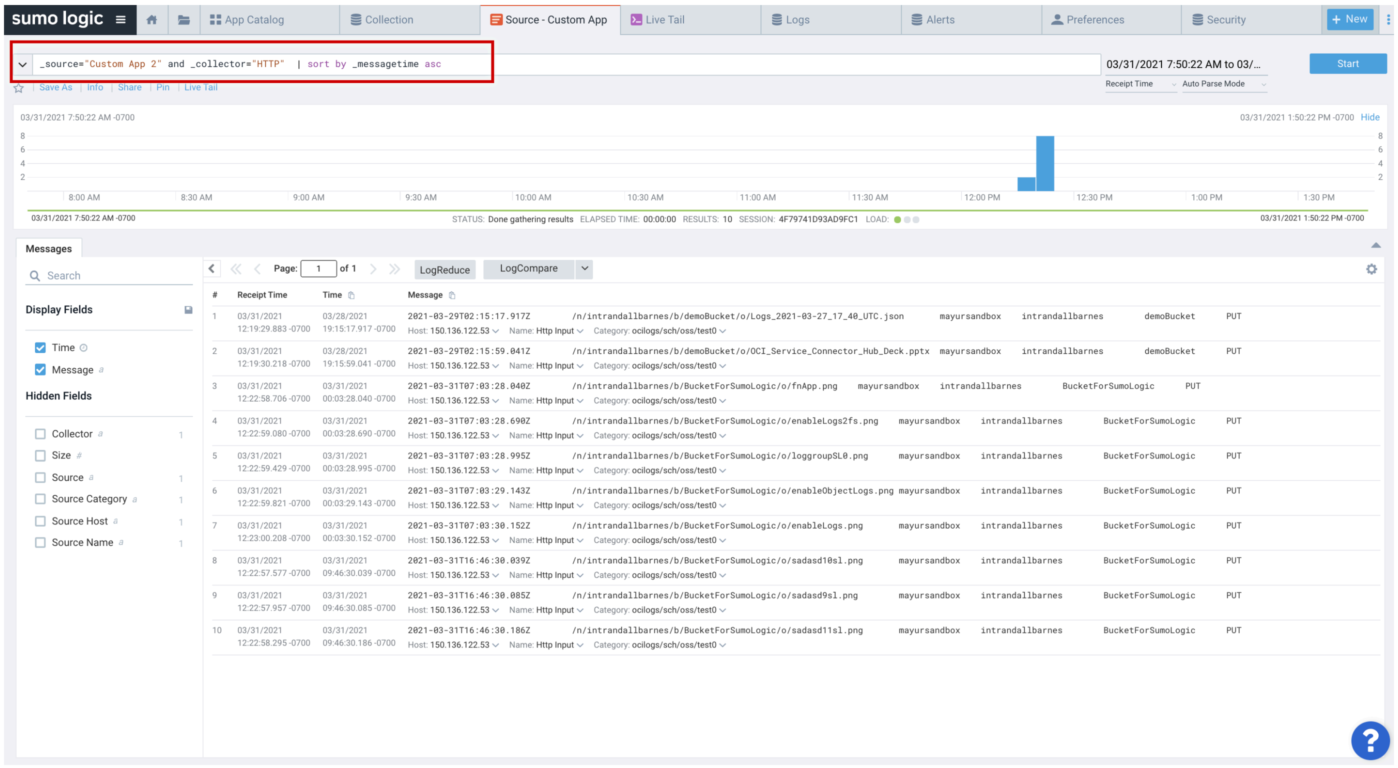
4:00:33 PM STATUS: Done gathering results ELAPSED TIME: 00:00:01 RESULTS: 144 SESSION: 74FBCA043A676B27 LOAD: 4:15:33 PM

Page: 1 of 6

Display Fields	
Time	
Message	
Hidden Fields	
Collector	2
Size	
Source	2
Source Category	2
Source Host	2
Source Name	2

#	Time	Message
1	09/02/2018 16:15:00.000 +0100	
2	09/02/2018 16:14:57.000 +0100	
3	09/02/2018 16:14:55.000 +0100	
4	09/02/2018 16:14:25.000 +0100	

Step 3: return all `Log entries` in Histogram



```
_source="Custom App 2" and _collector="HTTP"  | sort by _messagetime asc
```

03/31/2021 7:50:22 AM to 03/…

STATUS: Done gathering results ELAPSED TIME: 00:00:00 RESULTS: 10 SESSION: 4F79741D93AD9FC1

03/31/2021 7:50:22 AM -0700 — 03/31/2021 1:50:22 PM -0700

Page: 1 of 1

#	Receipt Time	Time	Message
1	03/31/2021 12:19:29.883 -0700	03/28/2021 19:15:17.917 -0700	`2021-03-29T02:15:17.917Z /n/intrandallbarnes/b/demoBucket/o/Logs_2021-03-27_17_40_UTC.json mayursandbox intrandallbarnes demoBucket PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
2	03/31/2021 12:19:30.218 -0700	03/28/2021 19:15:59.041 -0700	`2021-03-29T02:15:59.041Z /n/intrandallbarnes/b/demoBucket/o/OCI_Service_Connector_Hub_Deck.pptx mayursandbox intrandallbarnes demoBucket PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
3	03/31/2021 12:22:58.706 -0700	03/31/2021 00:03:28.040 -0700	`2021-03-31T07:03:28.040Z /n/intrandallbarnes/b/BucketForSumoLogic/o/fnApp.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
4	03/31/2021 12:22:59.080 -0700	03/31/2021 00:03:28.690 -0700	`2021-03-31T07:03:28.690Z /n/intrandallbarnes/b/BucketForSumoLogic/o/enableLogs2fs.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
5	03/31/2021 12:22:59.429 -0700	03/31/2021 00:03:28.995 -0700	`2021-03-31T07:03:28.995Z /n/intrandallbarnes/b/BucketForSumoLogic/o/loggroupSL0.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
6	03/31/2021 12:22:59.821 -0700	03/31/2021 00:03:29.143 -0700	`2021-03-31T07:03:29.143Z /n/intrandallbarnes/b/BucketForSumoLogic/o/enableObjectLogs.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
7	03/31/2021 12:23:00.208 -0700	03/31/2021 00:03:30.152 -0700	`2021-03-31T07:03:30.152Z /n/intrandallbarnes/b/BucketForSumoLogic/o/enableLogs.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
8	03/31/2021 12:22:57.577 -0700	03/31/2021 09:46:30.039 -0700	`2021-03-31T16:46:30.039Z /n/intrandallbarnes/b/BucketForSumoLogic/o/sadasd10sl.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
9	03/31/2021 12:22:57.957 -0700	03/31/2021 09:46:30.085 -0700	`2021-03-31T16:46:30.085Z /n/intrandallbarnes/b/BucketForSumoLogic/o/sadasd9sl.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0
10	03/31/2021 12:22:58.295 -0700	03/31/2021 09:46:30.186 -0700	`2021-03-31T16:46:30.186Z /n/intrandallbarnes/b/BucketForSumoLogic/o/sadasd11sl.png mayursandbox intrandallbarnes BucketForSumoLogic PUT` Host: 150.136.122.53 Name: Http Input Category: ocilogs/sch/oss/test0

advantages on basic mode over Advance mode

Important

Easy to use, structured query builder.

guide users through query builder based on how the data has been set up in your sumo Logic Sysytem.

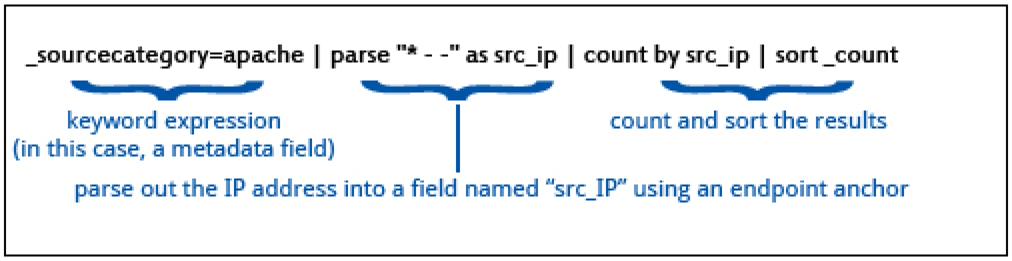
help perform log search queries with minima training.

All queries begin with a **keyword** or **string search**. Wildcards are allowed including an asterisk `(*)` for zero or more characters and a question mark `(?)` for a single character. Strings can be parsed based on start and stop anchor points in messages, and then aliased as user-created fields. **All operators** are separated by the pipe symbol `(|)`.

Here's an example:

```
_sourceCategory=apache | parse "* --" as src_ip | count by src_ip | sort _count
```

This query means:


_sourcecategory=apache | parse "* - -" as src_ip | count by src_ip | sort _count
keyword expression
(in this case, a metadata field)
count and sort the results
parse out the IP address into a field named "src_IP" using an endpoint anchor

As queries get longer and more complex, it is a best practice to format your queries by using a soft return before the pipes, such as:

```
_sourceCategory=apache

| parse "* --" as src_ip

| count by src_ip | sort _count
```

This method lines up the pipes and makes your query much easier to read.

Note

Searches can be long and complex, but they are limited to a maximum of 15,000 characters.