BIG DATA PROJECT – NASA ACCESS LOGS 1995

- Diljyot Singh , Aditya Kankane

Project Overview

Brief Overview of The Project

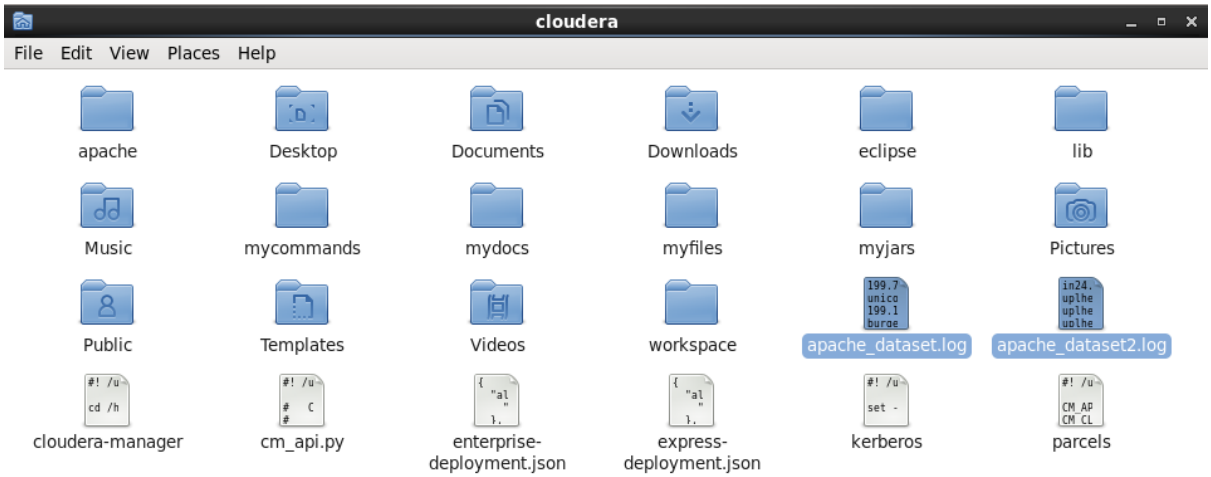
Our project mainly focused on working with HDFS, PIG, HIVE, and SQOOP to retrieve, preprocess, and find key insights from the data provided using Hadoop. Firstly we tried to clean and preprocess the data using apache log loader and piggybanks.jar files in PIG. Then we tried to consolidate the data which PIG has provided us in parts. We merged these part files and exported a consolidated file into our machine using the HDFS commands to provide the input to the HIVE as a CSV file. We then imported the file into HIVE and wrote HIVE queries in order to analyze the data as per the requirement. To more efficiently analyze and create more visually appealing data we used SQOOP in order to transfer the data from HIVE to MySQL.

Learning Objective

- Using PIG we were able to learn how to process and clean large files.
- Using HIVE we were able to learn how to analyse the dataset.
- Using SQOOP we were able to learn how to import data from hive to relational databases (MySQL)

Commands / Code Section

Copying log file to Linux



HDFS Commands

su root

hdfs dfs -mkdir /Project

hdfs dfs -mkdir /Project/input

hdfs dfs -put /home/cloudera/apache_dataset.log /Project/input

hdfs dfs -put /home/cloudera/apache_dataset2.log /Project/input

PIG Commands

REGISTER /home/cloudera/piggybank.jar;

DEFINE ApacheCommonLogLoader org.apache.pig.piggybank.storage.apachelog.CommonLogLoader();

logs = load '/Project/input' USING ApacheCommonLogLoader as (HostIP, hyphen, user, timestamp, Protocol, URL, HttpVers, Status, Bytes);

sortedlogs = FOREACH logs GENERATE HostIP, (chararray)SUBSTRING(timestamp,0,9) as date, (chararray)SUBSTRING(timestamp,12,20) as time, Protocol, URL, HttpVers, Status, Bytes;

dump sortedlogs;

```
cloudera@quickstart:/home/cloudera
File  Edit  View  Search  Terminal  Help
f,HTTP/1.0,200,199746)
(www-c8.proxy.aol.com,31/Aug/19,23:59:42,GET,/icons/menu.xbm,HTTP/1.0,200,527)
(www-c8.proxy.aol.com,31/Aug/19,23:59:42,GET,/icons/blank.xbm,HTTP/1.0,304,0)
(www-c8.proxy.aol.com,31/Aug/19,23:59:42,GET,/icons/image.xbm,HTTP/1.0,304,0)
(198.146.93.224,31/Aug/19,23:59:45,GET,/shuttle/countdown/lps/back.gif,HTTP/1.0,
200,1289)
(cindy.yamato.ibm.co.jp,31/Aug/19,23:59:47,GET,/facilities/tour.html,HTTP/1.0,20
0,3826)
(gatekeeper.uccu.com,31/Aug/19,23:59:47,GET,/facilities/lc39a.html,HTTP/1.0,304,
0)
(gatekeeper.uccu.com,31/Aug/19,23:59:49,GET,/images/kscmap-tiny.gif,HTTP/1.0,304
,0)
(gatekeeper.uccu.com,31/Aug/19,23:59:49,GET,/images/ksclogosmall.gif,HTTP/1.0,30
4,0)
(gatekeeper.uccu.com,31/Aug/19,23:59:49,GET,/images/lc39a-logo.gif,HTTP/1.0,304,
0)
(cys-cap-9.wyoming.com,31/Aug/19,23:59:52,GET,/shuttle/missions/sts-71/movies/st
s-71-launch-3.mpg,HTTP/1.0,200,57344)
(www-c8.proxy.aol.com,31/Aug/19,23:59:52,GET,/icons/unknown.xbm,HTTP/1.0,200,515
)
(cindy.yamato.ibm.co.jp,31/Aug/19,23:59:53,GET,/images/kscmap-small.gif,HTTP/1.0
,200,39017)
grunt>
```

store sortedlogs into '/Project/apache_http' USING PigStorage (',');

Hadoop Overview Datanodes Snapshot Startup Progress Utilities

Browse Directory

/Project/apache_http Go!

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	root	supergroup	0 B	Mon Nov 23 09:04:16 -0800 2020	1	128 MB	_SUCCESS
-rw-r--r--	root	supergroup	108.97 MB	Mon Nov 23 09:04:15 -0800 2020	1	128 MB	part-m-00000
-rw-r--r--	root	supergroup	108.72 MB	Mon Nov 23 09:04:16 -0800 2020	1	128 MB	part-m-00001
-rw-r--r--	root	supergroup	84.72 MB	Mon Nov 23 09:04:03	1	128 MB	part-m-00002

HIVE Commands

create database project;

use project;

create table projectlogs (HostIP string, date string, time string, Protocol string, URL string, HttpVers string, Status int, bytes double) row format delimited fields terminated by ',' lines terminated by '\n';

load data local inpath '/home/cloudera/hiveinput.csv' overwrite into table projectlogs;

create table highcount_host (HostIP string, hostcount int) row format delimited fields terminated by ',';

create table highcount_URL (URL string, URLcount int) row format delimited fields terminated by ',';

create table highdata_host (HostIP string, Hostdata int) row format delimited fields terminated by ',';

create table highdata_URL (URL string, URLdata int) row format delimited fields terminated by ',';

insert into highcount_host select HostIP, count(*) as Hostcount from projectlogs group by HostIP order by Hostcount desc;

insert into highcount_URL select URL, count(*) as URLcount from projectlogs group by URL order by URLcount desc;

insert into highdata_host select HostIP, sum(bytes) as Hostdata from projectlogs group by HostIP order by Hostdata desc;

insert into highdata_URL select URL, sum(bytes) as URLdata from projectlogs group by URL order by URLdata desc;

MySQL Commands

create database project;

use project;

create table highcount_host (HostIP VARCHAR(255), Hostcount INT);

create table highcount_URL (URL VARCHAR(255), URLcount INT);

create table highdata_host (HostIP VARCHAR(255), Hostdata DOUBLE);

create table highdata_URL (URL VARCHAR(255), URLdata DOUBLE);

Sqoop Commands

sqoop export --connect jdbc:mysql://localhost/project --username root --password cloudera --table highcount_host --export-dir /user/hive/warehouse/project.db/highcount_host --input-fields-terminated-by ',' --lines-terminated-by '\n'

sqoop export --connect jdbc:mysql://localhost/project --username root --password cloudera --table highcount_URL --export-dir /user/hive/warehouse/project.db/highcount_url --input-fields-terminated-by ',' --lines-terminated-by '\n'

sqoop export --connect jdbc:mysql://localhost/project --username root --password cloudera --table highdata_host --export-dir /user/hive/warehouse/project.db/highdata_host --input-fields-terminated-by ',' --lines-terminated-by '\n'

sqoop export --connect jdbc:mysql://localhost/project --username root --password cloudera --table highdata_URL --export-dir /user/hive/warehouse/project.db/highdata_url --input-fields-terminated-by ',' --lines-terminated-by '\n'

Analytics Section (As HIVE TABLE)

How many times each individual host has connected to our server? Store data sorted by highest count first.

HIVE Output:

Command: -
select * from highcount_host limit 10;

```
hive> select * from highcount_host limit 10;
OK
piweba3y.prodigy.com    21988
piweba4y.prodigy.com    16437
piweba1y.prodigy.com    12825
edams.ksc.nasa.gov      11962
163.206.89.4    9697
news.ti.com     8161
www-d1.proxy.aol.com    8047
alyssa.prodigy.com      8037
siltb10.orl.mmc.com     7573
www-a2.proxy.aol.com    7516
Time taken: 0.062 seconds, Fetched: 10 row(s)
```

MySQL Output:

Command: -
select * from highcount_host order by Hostcount desc limit 10;

```
mysql> select* from highcount_host order by Hostcount desc limit 10;
+----------------------+-----------+
| HostIP               | Hostcount |
+----------------------+-----------+
| piweba3y.prodigy.com |     21988 |
| piweba4y.prodigy.com |     16437 |
| piweba1y.prodigy.com |     12825 |
| edams.ksc.nasa.gov   |     11962 |
| 163.206.89.4         |      9697 |
| news.ti.com          |      8161 |
| www-d1.proxy.aol.com |      8047 |
| alyssa.prodigy.com   |      8037 |
| siltb10.orl.mmc.com  |      7573 |
| www-a2.proxy.aol.com |      7516 |
+----------------------+-----------+
10 rows in set (0.04 sec)
```

How many times each individual page has been requested from our server? Store data sorted by highest count first.

HIVE Output:

Command: -
Select * from highcount_URL limit 10;

```
hive> select * from highcount_URL limit 10;
OK
/images/NASA-logosmall.gif      208425
/images/KSC-logosmall.gif       164804
/images/MOSAIC-logosmall.gif    127647
/images/USA-logosmall.gif       126811
/images/WORLD-logosmall.gif     125667
/images/ksclogo-medium.gif      121277
/ksc.html       83684
/images/launch-logo.gif 75955
/history/apollo/images/apollo-logo1.gif 68854
/shuttle/countdown/     64691
Time taken: 0.051 seconds, Fetched: 10 row(s)
```

MySQL Output:

Command: -
select * from highcount_URL order by URLcount desc limit 10;

```
mysql> select * from highcount_URL order by URLcount desc limit 10;
+----------------------------------------+----------+
| URL                                    | URLcount |
+----------------------------------------+----------+
| /images/NASA-logosmall.gif             |   208425 |
| /images/KSC-logosmall.gif              |   164804 |
| /images/MOSAIC-logosmall.gif           |   127647 |
| /images/USA-logosmall.gif              |   126811 |
| /images/WORLD-logosmall.gif            |   125667 |
| /images/ksclogo-medium.gif             |   121277 |
| /ksc.html                              |    83684 |
| /images/launch-logo.gif                |    75955 |
| /history/apollo/images/apollo-logo1.gif |    68854 |
| /shuttle/countdown/                    |    64691 |
+----------------------------------------+----------+
10 rows in set (0.01 sec)
```

How much data has been downloaded by each individual host that has connected to our server? Store data sorted by highest count first.

HIVE Output:

Command: -
select * from highdata_host limit 10;

```
hive> select * from highdata_host limit 10;
OK
piweba3y.prodigy.com    524051073
piweba1y.prodigy.com    328707273
piweba4y.prodigy.com    327210469
news.ti.com     272165569
alyssa.prodigy.com      214506290
e659229.boeing.com      209036877
piweba2y.prodigy.com    189623731
webgate1.mot.com        177891198
163.206.89.4    175160386
poppy.hensa.ac.uk       173895618
Time taken: 0.069 seconds, Fetched: 10 row(s)
```

MySQL Output:

Command: -
Select * from highdata_host order by Hostdata desc limit 10;

```
mysql> select * from highdata_host order by Hostdata desc limit 10;
+----------------------+-----------+
| HostIP               | Hostdata  |
+----------------------+-----------+
| piweba3y.prodigy.com | 524051073 |
| piweba1y.prodigy.com | 328707273 |
| piweba4y.prodigy.com | 327210469 |
| news.ti.com          | 272165569 |
| alyssa.prodigy.com   | 214506290 |
| e659229.boeing.com   | 209036877 |
| piweba2y.prodigy.com | 189623731 |
| webgate1.mot.com     | 177891198 |
| 163.206.89.4         | 175160386 |
| poppy.hensa.ac.uk    | 173895618 |
+----------------------+-----------+
10 rows in set (0.07 sec)
```

How much data was sent out as each individual page was downloaded from our server? Store data sorted by highest count first.

HIVE Output:

Command: -
select * from highdata_URL limit 10;

```
hive> select * from highdata_URL limit 10;
OK
/shuttle/missions/sts-71/movies/sts-71-launch.mpg       2147483647
/shuttle/missions/sts-71/movies/sts-71-mir-dock.mpg     1405249895
/shuttle/missions/sts-71/movies/sts-71-tcdt-crew-walkout.mpg    1136737784
/shuttle/missions/sts-70/movies/sts-70-launch.mpg       1098272261
/shuttle/technology/sts-newsref/stsref-toc.html 1058787140
/shuttle/missions/sts-53/movies/sts-53-launch.mpg       1034715432
/shuttle/missions/sts-69/count69.gif    1004960681
/shuttle/countdown/video/livevideo2.gif 980732435
/shuttle/countdown/count70.gif  918852591
/shuttle/countdown/count.gif    828821710
Time taken: 0.051 seconds, Fetched: 10 row(s)
```

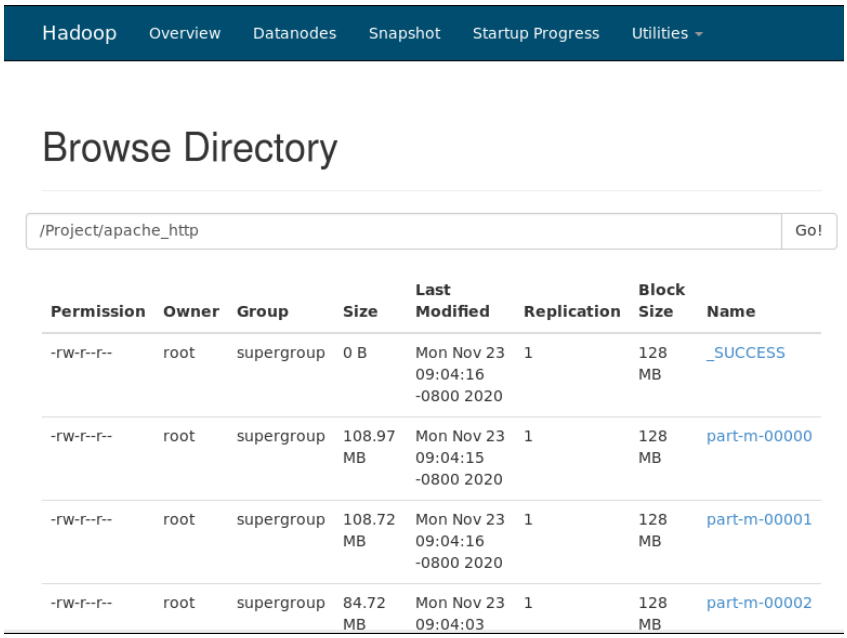
MySQL Output:

Command: -
select * from highdata_URL order by URLdata desc limit 10;

```
mysql> select * from highdata_URL order by URLdata desc limit 10;
+-------------------------------------------------------------+------------+
| URL                                                         | URLdata    |
+-------------------------------------------------------------+------------+
| /shuttle/missions/sts-71/movies/sts-71-launch.mpg           | 2147483647 |
| /shuttle/missions/sts-71/movies/sts-71-mir-dock.mpg         | 1405249895 |
| /shuttle/missions/sts-71/movies/sts-71-tcdt-crew-walkout.mpg | 1136737784 |
| /shuttle/missions/sts-70/movies/sts-70-launch.mpg           | 1098272261 |
| /shuttle/technology/sts-newsref/stsref-toc.html             | 1058787140 |
| /shuttle/missions/sts-53/movies/sts-53-launch.mpg           | 1034715432 |
| /shuttle/countdown/sts-69/count69.gif                       | 1004960681 |
| /shuttle/missions/video/livevideo2.gif                      |  980732435 |
| /shuttle/countdown/count70.gif                              |  918852591 |
| /shuttle/countdown/count.gif                                |  828821710 |
+-------------------------------------------------------------+------------+
10 rows in set (0.09 sec)
```

Concatenating Files (As HDFS FILE)

Merging Pig part files and Exporting into CSV for HIVE input:

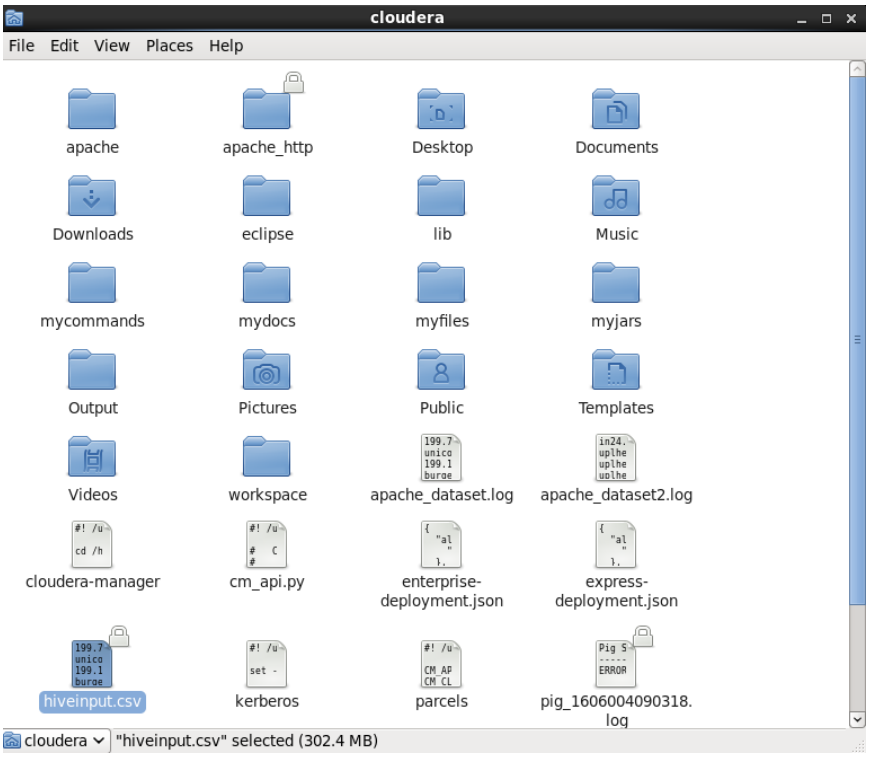


Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	root	supergroup	0 B	Mon Nov 23 09:04:16 -0800 2020	1	128 MB	_SUCCESS
-rw-r--r--	root	supergroup	108.97 MB	Mon Nov 23 09:04:15 -0800 2020	1	128 MB	part-m-00000
-rw-r--r--	root	supergroup	108.72 MB	Mon Nov 23 09:04:16 -0800 2020	1	128 MB	part-m-00001
-rw-r--r--	root	supergroup	84.72 MB	Mon Nov 23 09:04:03	1	128 MB	part-m-00002

HDFS Commands:

hdfs dfs -get /Project/apache_http /home/cloudera

hadoop fs -getmerge /Project/apache_http /home/cloudera/hiveinput.csv



Using the above results and also carrying out any other analysis as required, providing answers to the following questions.

Which host has connected the maximum number of times to our server? Give the host name & count of connections from that host.

Ans: - Host: **piweba3y.prodigy.com**

Count of connections: **21988**

Explanation:

We processed the data from the highcount_host table which we have created and limited the data to 1 for the maximum count of the host.

```
hive> select *  from highcount_host limit 1;
OK
piweba3y.prodigy.com    21988
Time taken: 0.04 seconds, Fetched: 1 row(s)
```

Which page that has been requested the maximum number of times from our server? Give the page name & count of the times the page was requested.

Ans: - Page: **/images/NASA-logosmall.gif**

Count: **208425**

Explanation:

We processed the data from the highcount_url table which we have created and limited the data to 1 for the maximum count of the page.

```
hive> select * from highcount_url limit 1;
OK
/images/NASA-logosmall.gif      208425
Time taken: 0.058 seconds, Fetched: 1 row(s)
```

How many unique hosts have connected to our server? Give counts.

Ans: - Count: **137842**

Explanation:

We processed the data from the highcount_host table which we have created and limited the data to 1 for the count of the unique host.

```
OK
137842
Time taken: 24.493 seconds, Fetched: 1 row(s)
hive> select count(*) as uniquehosts from highcount_host limit 1;
```

How many unique pages have been requested from our server? Give counts.
Ans: - Count: **15809**
Explanation:
We processed the data from the highcount_url table which we have created and limited the data to 1 for the count of the unique pages.

```
OK
15809
Time taken: 23.723 seconds, Fetched: 1 row(s)
hive> select count(*) as uniquepages from highcount_url;
```

Which host has caused maximum data transfer from our server? Give host name & the data transfer for the host.
Ans: - Host: **piweba3y.prodigy.com**
Data Transfer: **5240510738 bytes**
Explanation:
We processed the data from the projectlogs table which we have created and limited the data to 1 for the maximum data transfer by the host.

```
OK
piweba3y.prodigy.com    5.24051073E8
Time taken: 69.649 seconds, Fetched: 1 row(s)
hive> select HostIP, sum(bytes) as Hostdata from projectlogs group by HostIP order by Hostdata desc limit 1;
```

Which page has caused maximum data transfer from our server? Give page name & the data transfer for the page.
Ans: - Page: **/shuttle/missions/sts-71/movies/sts-71-launch.mpg**
Data Transfer: **31929459 bytes**
Explanation:
We processed the data from the projectlogs table which we have created and limited the data to 1 for the maximum data transfer by the page.

```
OK
/shuttle/missions/sts-71/movies/sts-71-launch.mpg       3.192945E9
Time taken: 65.804 seconds, Fetched: 1 row(s)
hive> select URL, sum(bytes) as URLdata from projectlogs group by URL order by URLdata desc limit 1;
```

Which page has maximum download size from our server? Give page name & the size for the page.
Ans: - Page: **/shuttle/countdown/video/livevideo.jpeg**
Size: **6823936 bytes**
Explanation:
We processed the data from the projectlogs table which we have created and limited the data to 1 for the maximum download size by the page. We have also added the (where protocol like 'GET') clause to exclude the upload data ('POST' and 'HEAD') for the HIVE query.

```
OK
/shuttle/countdown/video/livevideo.jpeg 6823936.0
Time taken: 158.751 seconds, Fetched: 1 row(s)
hive> select URL, bytes from projectlogs where Protocol like 'GET' order by bytes desc limit 1;
```

What is the download count of the page that has maximum download size from our server? Give page name & download count.

Ans: - Page: **/shuttle/countdown/video/livevideo.jpeg**

Count: **11070**

Explanation:

We processed the data from the projectlogs table which we have created and limited the data to 1 for the download count of the maximum download size by the page. We have also added the (where protocol like 'GET') clause to exclude the upload data ('POST' and 'HEAD') for the HIVE query. URLcount was used as an output variable for the count.

```
OK
/shuttle/countdown/video/livevideo.jpeg 11070
Time taken: 96.449 seconds, Fetched: 1 row(s)
hive> select URL, count(*) as URLcount from projectlogs where (select URL from projectlogs where Protocol like 'GET' and bytes>0 order by by
tes desc limit 1);
```

Which page has minimum download size from our server? Give page name & the size for the page.

Ans: - Page: **/cgi-bin/imagemap/countdown70?396**

Size: **1 byte**

Explanation:

We processed the data from the projectlogs table which we have created and limited the data to 1 for the minimum download size by the page. We have added the where clause to exclude the NULL and zero values in the dataset.

```
OK
/cgi-bin/imagemap/countdown70?396       1.0
Time taken: 187.163 seconds, Fetched: 1 row(s)
hive> select URL, bytes from projectlogs where Protocol like 'GET' and bytes>0 order by bytes limit 1;
```

What is the download count of the page that minimum download size from our server? Give page name & the size for the page.

Ans: - Page: **/cgi-bin/imagemap/countdown70?396**

Count: **144**

Explanation:

We processed the data from the projectlogs table which we have created and limited the data to 1 for the minimum download size by the page. We have added the where clause to exclude the NULL and zero values in the dataset. URLcount was used as an output variable.

```
OK
/cgi-bin/imagemap/countdown70?396       144     1.0
Time taken: 143.141 seconds, Fetched: 1 row(s)
hive> select URL, count(*) as URLcount, min(bytes) as URLdata from projectlogs where Protocol like 'GET' and bytes>0 group by URL order by URLdata lim
it 1;
```

Summary

Hadoop is an open-source software framework that provides for processing of large data sets using simple programming models such as PIG, HIVE, SQOOP, etc.
Using PIG we were able to learn how to process and clean large files. Using HIVE we were able to learn how to analyse the dataset. Using SQOOP we were able to learn how to import data from hive to relational databases (MySQL).
Hadoop helps us analyze and work on big data with ease where traditional methods fails.
It helps us exploring data with large scale datasets and provides an environment for exploratory data analysis
It helps us easy the data pre-processing tasks by providing tools like MapReduce, PIG, and Hive for efficiently handling large scale data.
Data mining techniques also got easy to use as HIVE uses similar SQL queries which we have learnt in the past.