

[Alt+S] Mumbai Yash Chaudhary

Instances (1/4) Info

Find Instance by attribute or tag (case-sensitive) Any state

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/> 202318022-main	i-00f495b730e7f1ffb	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a
<input type="checkbox"/> 202318022-dataCluster1	i-0cf8f1d16aa838fb9	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a
<input type="checkbox"/> 202318022-dataCluster2	i-01c5a6d7fee54fa8	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a
<input type="checkbox"/> 202318022-SNN	i-0152fad87776ca69f	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a

Instance: i-00f495b730e7f1ffb (202318022-main)

Details Status and alarms New Monitoring Security Networking Storage Tags

▼ Instance summary Info

Instance ID
i-00f495b730e7f1ffb (202318022-main)

IPv6 address
-

Hostname type
IP name: ip-172-31-34-199.ap-south-1.compute.internal

Public IPv4 address
13.234.118.49 [open address](#)

Instance state
Running

Private IP DNS name (IPv4 only)
ip-172-31-34-199.ap-south-1.compute.internal

Private IPv4 addresses
ec2-13-234-118-49.ap-south-1.compute.amazonaws.com [open address](#)

Public IPv4 DNS copied

Mapper:

```
#!/usr/bin/python3 -0

import sys

# Loop through each line in the input
for line in sys.stdin:
    # Remove leading and trailing whitespace
    line = line.strip()
    # Split the line into words
    words = line.split()
    # Emit key-value pairs of word and count of 1
    for word in words:
        print(word+"\t",1)
```

Reducer:

```
#!/usr/bin/python3 -0

import sys

# Initialize variables to keep track of current word and its count
current_word = None
current_count = 0

# Loop through each line in the input
```

```

for line in sys.stdin:
    # Split the line into word and count, separated by tab
    word, count = line.strip().split('\t', 1)

    # Convert count to integer
    count = int(count)

    # If the word is the same as the current word, increment its count
    if word == current_word:
        current_count += count
    else:
        # If the word is different, print the current word and its count
        if current_word:
            print(current_word, "\t", current_count)
        # Update current word and its count
        current_word = word
        current_count = count

# Print the last word and its count
if current_word:
    print(current_word, "\t", current_count)

```

```

ubuntu@ip-172-31-34-199: ~
ubuntu@ip-172-31-34-199:~$ hadoop/sbin/start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
Starting namenodes on [ec2-13-234-118-49.ap-south-1.compute.amazonaws.com]
ec2-13-234-118-49.ap-south-1.compute.amazonaws.com: starting namenode, logging to /home/ubuntu/hadoop/logs/hadoop-ubuntu-
namenode-ip-172-31-34-199.out
ec2-35-154-77-212.ap-south-1.compute.amazonaws.com: starting datanode, logging to /home/ubuntu/hadoop/logs/hadoop-ubuntu-
datanode-ip-172-31-32-80.out
ec2-65-1-86-124.ap-south-1.compute.amazonaws.com: starting datanode, logging to /home/ubuntu/hadoop/logs/hadoop-ubuntu-d
atanode-ip-172-31-46-91.out
Starting secondary namenodes [ec2-13-235-245-182.ap-south-1.compute.amazonaws.com]
ec2-13-235-245-182.ap-south-1.compute.amazonaws.com: bash: line 0: cd: /home/ubuntu/hadoop: No such file or directory
ec2-13-235-245-182.ap-south-1.compute.amazonaws.com: bash: /home/ubuntu/hadoop/sbin/hadoop-daemon.sh: No such file or di
rectory
starting yarn daemons
starting resourcemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubuntu-resourcemanager-ip-172-31-34-199.out
ec2-35-154-77-212.ap-south-1.compute.amazonaws.com: starting nodemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubunt
u-nodemanager-ip-172-31-32-80.out
ec2-65-1-86-124.ap-south-1.compute.amazonaws.com: starting nodemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubuntu-
nodemanager-ip-172-31-46-91.out
ubuntu@ip-172-31-34-199:~$ jps
23729 NameNode
24201 Jps
23961 ResourceManager

```

```

ubuntu@ip-172-31-34-199: ~
ubuntu@ip-172-31-34-199:~$ hadoop jar /home/ubuntu/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.7.3.jar -file /home/
ubuntu/mapper.py -mapper mapper.py -file /home/ubuntu/reducer.py -reducer reducer.py -input /tmp.txt -output /1
24/02/28 12:50:24 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [/home/ubuntu/mapper.py, /home/ubuntu/reducer.py, /tmp/hadoop-unjar6744328353870054625/] [] /tmp/streamjo
b4882233691055212999.jar tmpDir=null
24/02/28 12:50:26 INFO client.RMProxy: Connecting to ResourceManager at ec2-13-234-118-49.ap-south-1.compute.amazonaws.c
om/172.31.34.199:8032
24/02/28 12:50:26 INFO client.RMProxy: Connecting to ResourceManager at ec2-13-234-118-49.ap-south-1.compute.amazonaws.c
om/172.31.34.199:8032
24/02/28 12:50:27 WARN hdfs.DFSClient: Caught exception

```

```
ubuntu@ip-172-31-34-199: ~  
Merged Map outputs=2  
GC time elapsed (ms)=370  
CPU time spent (ms)=1610  
Physical memory (bytes) snapshot=510308352  
Virtual memory (bytes) snapshot=5515010048  
Total committed heap usage (bytes)=259874816  
Shuffle Errors  
BAD_ID=0  
CONNECTION=0  
IO_ERROR=0  
WRONG_LENGTH=0  
WRONG_MAP=0  
WRONG_REDUCE=0  
File Input Format Counters  
Bytes Read=56  
File Output Format Counters  
Bytes Written=59  
24/02/28 12:51:01 INFO streaming.StreamJob: Output directory: /1  
ubuntu@ip-172-31-34-199:~$ hdfs dfs -cat /1/part-00000  
are      2  
doing    1  
hello    1  
how      1  
what     1  
you      2  
ubuntu@ip-172-31-34-199:~$ cat temp.txt  
hello  
how are you  
what are you doing  
ubuntu@ip-172-31-34-199:~$
```