AWS Cloud Project submission document

Harish Kumar Chikkamaranahalli Ashokkumar

ID : 1001100559

The file contains .py files which have the programming code written for the project.

from boto.s3.connection import S3Connection

from boto.s3.key import Key

import time

conn = S3Connection('ACCESS KEY','SECRET KEY')

bucket = conn.create\_bucket('harishkumarca')

k=Key(bucket)

k.key = 'myfile'

start = time.time()

k.set\_contents\_from\_filename('all\_month.csv')

k.get\_contents\_to\_filename('downlaoded.csv')

end = time.time()

print end-start

conn.close()

Time taken to upload and download file from S3 is 4.2647600174

For the above to run we need to install boto interface into the system.

install mysql:

1: sudo apt-get install mysql-server

2: sudo apt-get install python-dev libmysqlclient-dev

3: sudo apt-get install python-pip

4: sudo apt-get install python-mysqldb

5: sudo apt-get awscli

6: sudo pip install python-memcached

Installing mysql is another process that has to be completed before running the below code. I have used UBUNTU 14.04 on EC2 instance hence the above procedure.

The below code is used to upload data from AWS S3 to AWS RDS instance.

import csv

import MySQLdb

import time

import os

import memcache

os.system('aws s3 cp s3://harishkumarca/myfile earthquake\_details.csv')

con = MySQLdb.connect(host="hostname",user="username",passwd="password",db="datbasename",local\_infile=1)

cursor = con.cursor()

cursor.execute("DROP TABLE IF EXISTS PROJECT2")

cursor.execute("CREATE TABLE PROJECT2(time timestamp,latitude float,longitude float,depth float,mag float,magtype varchar(255),nst int,gap float,dmin float,rms float,net varchar(255),id varchar(255),updated timestamp,place varchar(255),type varchar(255))")

cursor.execute("""load data local infile 'earthquake\_details.csv' into table PROJECT2 fields terminated by ',' enclosed by '\"' lines terminated by '\n' ignore 1 lines;""")

cursor.execute("commit")

query = """select week, count(mag2) as mag2, count(mag3) as mag3, count(mag4) as mag4, count(mag5) as mag5

from

((select

case

when date(time) between cast('2015-01-20' as date) and cast('2015-01-26' as date) then 1

when date(time) between cast('2015-01-27' as date) and cast('2015-02-02' as date) then 2

when date(time) between cast('2015-02-03' as date) and cast('2015-02-09' as date) then 3

when date(time) between cast('2015-02-10' as date) and cast('2015-02-16' as date) then 4

when date(time) between cast('2015-02-17' as date) and cast('2015-02-23' as date) then 5

when date(time) between cast('2015-02-24' as date) and cast('2015-02-29' as date) then 6

end week,

PROJECT2.id

from

PROJECT2) as week, (select

case

when mag between 2 and 2.99 then mag

end mag2,

PROJECT2.id

from

PROJECT2) as mag2, (select

case

when mag between 3 and 3.99 then mag

end mag3,

PROJECT2.id

from

PROJECT2) as mag3, (select

case

when mag between 4 and 4.99 then mag

end mag4,

PROJECT2.id

from

PROJECT2) as mag4, (select

case

when mag >= 5 then mag

end mag5,

PROJECT2.id

from

PROJECT2) as mag5)

where

week.id = mag2.id and

week.id = mag3.id and

week.id = mag4.id and

week.id = mag5.id

group by week;"""

start=time.time()

cursor.execute(query)

end=time.time()

mag\_time = end - start

print "Time for calculating time magnitude is",mag\_time

print cursor.fetchall();

print('2000 Random queries')

query = 'select \* from PROJECT2 group by rand() limit 10'

start=time.time()

for i in range(1,2000):

cursor.execute(query)

end=time.time()

rand\_time = end-start

print "Time for executing 2000 random queries is",rand\_time

start=time.time()

restricted\_query = 'select \* from (select \* from PROJECT2 limit 2000) A group by rand() limit 10'

for i in range(1,2000):

cursor.execute(restricted\_query)

end=time.time()

limit\_time = end-start

print "Limited query time is ",limit\_time

con.close()

def memcache\_processing():

memc = memcache.Client(['127.0.0.1:11211'], debug=1);

conn = MySQLdb.connect(host="harishkumarca.c52ndx7xtxxn.us-east-1.rds.amazonaws.com",user="harishkumarca",passwd="ahamBrahman$1",db="harishkumarca",local\_infile=1)

start=time.time()

cursor = conn.cursor()

for i in range(1,2000):

PROJECT2 = memc.get('PROJECT2')

if not PROJECT2:

cursor.execute('select \* from PROJECT2 order by time limit 5')

rows = cursor.fetchall()

memc.set('PROJECT2',rows,60)

print "Updated memcached with MySQL data"

else:

print "Loaded data from memcached", i

end=time.time()

start=time.time()

restricted\_query = 'select \* from (select \* from PROJECT2 limit 2000) A group by rand() limit 10'

for i in range(1,2000):

cursor.execute(restricted\_query)

end=time.time()

limit\_time = end-start

memcache\_time=end-start

print "Time taken for memcache processing", memcache\_time

print "Time taken for limited query in memcache",limit\_time

conn.close()

print "deleted file";

os.system('rm earthquake\_details.csv');

memcache\_processing()

print "done";

The below are the result time taken for many of the process to get completed.

Time taken for memcache processing 0.261202096939

Time taken for limited query in memcache 15.0676710606

Time for executing 2000 random queries is 42.0944821835

Limited query time is 14.9025080204

Time for calculating time magnitude is 0.0911300182343

References:

<http://boto.readthedocs.org/en/latest/>

<http://mysql-python.sourceforge.net/MySQLdb.html>