



Demeter

Automated QA for Athena

APRIL 2020

CRISTOBAL ORTEGA ARTERO

ortega.cortega@gmail.com

<https://www.linkedin.com/in/cristobalortegaartero/>

+34 647473971

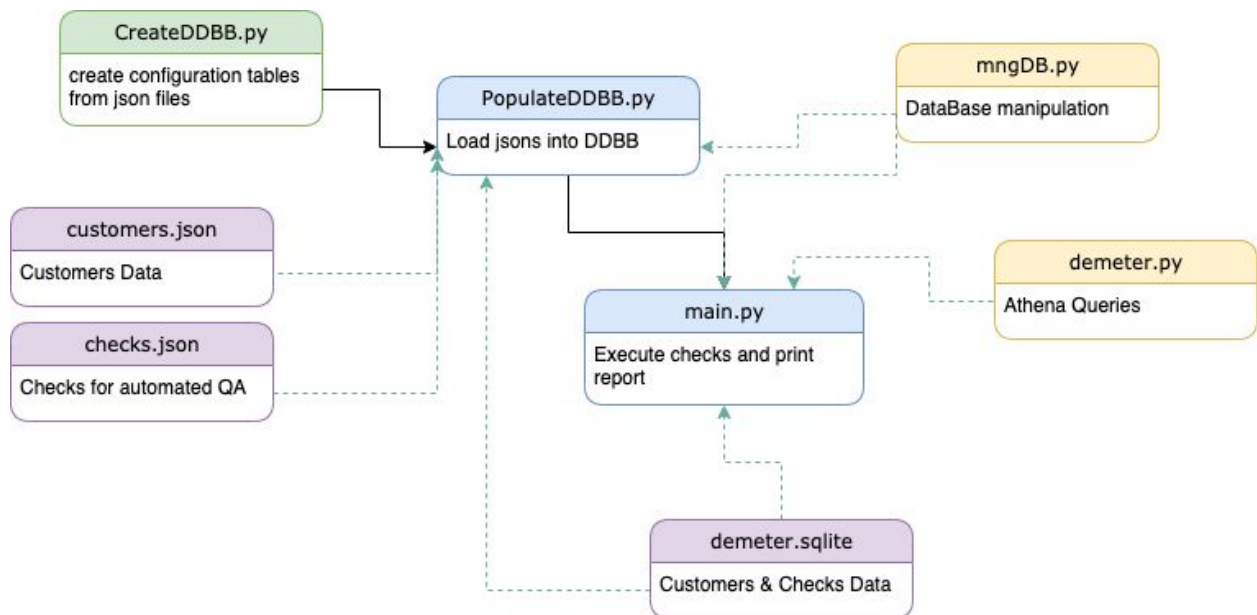
Introduction & Background

General

A small python suite to connect Athena in order to execute queries.
So, far it only supports the use of AWS_CLI credentials.

General Schema

Modules



CreateDDBB.py

Dependencies:

```
import sqlite3
```

The module creates **demeter.sqlite** tables.

It needs to be used only when the database needs to be created.

> python CreateDDBB.py

PopulateDDBB.py

Dependencies:

```
import mngDB
```

```
import json
```

The module loads **checks.json** and **customers.json** into **demeter.sqlite** tables.

It needs to be used when the **checks.json** or **customers.json** are modified in order to refresh the database **demeter.sqlite**.

> python PopulateDDBB.py

demeter.py

Dependencies:

```
import boto3
```

```
import time
```

This module contains the functions to run the connection and queries against **Athena**.

A possible improvement is to convert it into a class.

```
session(region_name,profile_name)
```

This function creates and returns the client athena object. Needs only the profile name from AWS credentials file and the region.

```
athena_query_result(client, params,max_execution = 5)
```

This function executes the query given in params.

Params dict:

```

params = {
    'region':
    'database':
    'bucket':
    'path':
    'query':
}

```

main.py

Dependencies:

```

import demeter
import mngDB

```

This module executes an example of a report. This is the result with the jsons in the example:

```

-----
procesando cliente: M7_HDAus
-----

Check REG_001:
    WARNING!!!!!!:20200412 > 20200406
    Registrados:129949 . Last computed daydate: 20200406 (Goal: 20200412)

Check SUBS_001:
    Subscripciones. Last computed daydate: 20200406 (Goal: 20200412)
        suscripcion                : count
        -----
        Kombi                      : 65599
        HD Austria                 : 32122
        HD Austria 24h Ero         : 5759
        Plus Austria 24h Ero       : 3513
        HD Austria Smart           : 3230
        Plus Austria 8h Ero        : 743
        Plus Austria No Ero        : 175

```

Check SUBS_002:

Cancelaciones. Month: 202003

mes	: suscripcion	: count
-----	---------------	---------

202003	: HD Austria	: 303
202003	: HD Austria 24h Ero	: 613
202003	: HD Austria Smart	: 214
202003	: Kombi	: 8
202003	: Plus Austria 24h Ero	: 422
202003	: Plus Austria 8h Ero	: 24
202003	: Plus Austria No Ero	: 46

Check PLAY_001:

WARNING!!!!!!:20200412 > 20200405

Playbacks. Last computed date: 20200405 (Goal: 20200412)

daydate	: count
---------	---------

20200405	: 216946
20200404	: 216323
20200403	: 192822
20200402	: 176459
20200401	: 188065
20200331	: 189815
20200330	: 202155

procesando cliente: PUREFLIX

Check PLAY_001:

Athena query with the string "select daydate,date_format(now(),'%Y%m%d'),
count(*) from pro.fact_playbackactivity group by daydate order by daydate desc
limit 7;" failed or was cancelled

Process finished with exit code 0

The last error is due the table fact_playbackactivity is not yet created.



Json files

customers.json

Example:

```
{
  "M7_HDAus": {
    "region": "eu-west-1",
    "database": "stg",
    "bucket": "jptvs-aws-stg-af6f35ba-5f82-11ea-b6ba-0354e9fb4247",
    "path": "athena",
    "profile_name": "M7_stg",
    "brand": "291bfcf4-5f83-11ea-b6ba-8fc2f567b0a6"
  },
  "PUREFLIX": {
    "region": "us-east-1",
    "database": "pro",
    "bucket": "jptvs-aws-pro-21e452e6-684e-11ea-8438-976e67e72f9c",
    "path": "athena",
    "profile_name": "pureflix",
    "brand": ""
  }
}
```

- **brand** is the internal database parameter of jump.

checks.json

Example:

```
{
  "REG_001": {
```

```

    "query": "select daydate,date_format(now(),'%Y%m%d'),count(*) from #d.fact_registereduser where type =
'Registration' and brandid='#b' group by daydate order by daydate desc limit 1;",
    "warning": "#2 > #1",
    "header": "Registros:#3 . Last computed daydate: #1 (Goal: #2)",
    "legend": "",
    "message": "",
    "applyto": "",
    "notapplyto": "PUREFLIX"
  },
  "SUBS_001": {
    "query": "select daydate,date_format(now(),'%Y%m%d'),subscriptionid, count(*) from
#d.fact_registereduser where type = 'Subscription' and brandid='#b' and daydate = (select max(daydate)
from #d.fact_registereduser) group by daydate,subscriptionid order by count(*) desc limit 10;",
    "warning": "",
    "header": "Suscripciones. Last computed daydate: #1 (Goal: #2)",
    "legend": "suscripcion : count",
    "message": "#3 : #4",
    "applyto": "",
    "notapplyto": "PUREFLIX"
  },
  "SUBS_002": {
    "query": "select substr(daydate,1,6), subscriptionid, count(*) from #d.fact_userdeactivation where event =
'Subscription' and eventtype='End' and brandid='#b' and substr(daydate,1,6) = (select
max(substr(daydate,1,6)) from #d.fact_userdeactivation) group by substr(daydate,1,6), subscriptionid order
by substr(daydate,1,6) desc,subscriptionid;",
    "warning": "",
    "header": "Cancelaciones. Month: #1",
    "legend": "mes : suscripcion : count",
    "message": "#1 : #2 : #3",
    "applyto": "",
    "notapplyto": "PUREFLIX"
  },
  "PLAY_001": {
    "query": "select daydate,date_format(now(),'%Y%m%d'), count(*) from #d.fact_playbackactivity group by
daydate order by daydate desc limit 7;",
    "warning": "#2 > #1",
    "header": "Playbacks. Last computed date: #1 (Goal: #2)",
    "legend": "daydate : count",
    "message": "#1 : #3",
    "applyto": "",
    "notapplyto": ""
  }
}

```

```
}  
}
```

- **query** must use **#d** for the database and **#b** for the brand in order to make the test usable for several brands/customers. It must be in a single line.
- **warning** is the condition to give a warning. **#1** to **#n** are the result fields of the query.
- **header** is the header in the report
- **legend** is the header of the query results if multiple
- **message** is the presented results if multiple
- **applyto** are the customers to be checked, * is wildcard
- **notapplyto** are the customers to exclude, * is wildcard