Verifying Data Quality for a Data Warehouse

Exercise 1 - Getting the environment ready

Step 1: Start the postgresql server.

- Start Postgres

```
theia@theiadocker-craigtrupp8:/home/project$ start_postgres
Starting your Postgres database....
This process can take up to a minute.

Postgres database started, waiting for all services to be ready....

Your Postgres database is now ready to use and available with username:
postgres password: MjE1NDQtY3JhaWd0

You can access your Postgres database via:
 • The Browser with pgadmin
   • URL:
https://craigtrupp8-5050.theiadocker-3-labs-prod-theiak8s-4-tor01.proxy.cog
nitiveclass.ai/browser/
   • Database Password: MjE1NDQtY3JhaWd0
 • CommandLine: psql --username=postgres --host=localhost
theia@theiadocker-craigtrupp8:/home/project$
```

Step 2: Download the staging area setup script.

```
wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
ouse/setup_staging_area.sh
```

```
theia@theiadocker-craigtrupp8:/home/project$ wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
```

```
ouse/setup_staging_area.sh
--2023-10-05 15:33:28--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
ouse/setup_staging_area.sh
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 816 [text/x-sh]
Saving to: 'setup_staging_area.sh'

setup_staging_area.sh
100%[=============================================>]     816  --.-KB/s
in 0s

2023-10-05 15:33:28 (176 MB/s) - 'setup_staging_area.sh' saved [816/816]

theia@theiadocker-craigtrupp8:/home/project$ cat setup_staging_area.sh | wc
-l
29
theia@theiadocker-craigtrupp8:/home/project$ chmod +x setup_staging_area.sh
theia@theiadocker-craigtrupp8:/home/project$ ls -l
total 8
drwxr-sr-x 3 theia users 4096 Oct  5 14:03 postgres
-rwxr-xr-x 1 theia users  816 Aug  3  2022 setup_staging_area.sh
```

Step 3: Run the setup script.

- Need to drop in memory database instance prior to running bash script

```
theia@theiadocker-craigtrupp8:/home/project$ psql --username=postgres
--host=localhost
psql (15.2 (Ubuntu 15.2-1.pgdg18.04+1), server 13.2)
Type "help" for help.

postgres=# \l
```

```
                                                 List of databases
   Name    |  Owner   | Encoding |  Collate   |   Ctype    | ICU Locale |
Locale Provider |   Access privileges
-----------+----------+----------+------------+------------+------------+--
---------------+-----------------------
 billingDW | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            |
 postgres  | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            |
 template0 | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            | =c/postgres          +
           |          |          |            |            |            |
| postgres=CTc/postgres
 template1 | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            | =c/postgres          +
           |          |          |            |            |            |
| postgres=CTc/postgres
(4 rows)

postgres=# DROP DATABSE IF EXISTS "billingDW";
ERROR:  syntax error at or near "DATABSE"
LINE 1: DROP DATABSE IF EXISTS "billingDW";
             ^
postgres=# DROP DATABASE IF EXISTS billingDW;
NOTICE:  database "billingdw" does not exist, skipping
DROP DATABASE
postgres=# DROP DATABASE IF EXISTS "billingDW";
DROP DATABASE
postgres=# \l
                                                 List of databases
   Name    |  Owner   | Encoding |  Collate   |   Ctype    | ICU Locale |
Locale Provider |   Access privileges
-----------+----------+----------+------------+------------+------------+--
---------------+-----------------------
 postgres  | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            |
 template0 | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            | =c/postgres          +
           |          |          |            |            |            |
| postgres=CTc/postgres
 template1 | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            | =c/postgres          +
           |          |          |            |            |            |
```

```
| postgres=CTc/postgres
(3 rows)

postgres=#
```

- Now that we've dropped we can run the script

```
theia@theiadocker-craigtrupp8:/home/project$ bash setup_staging_area.sh
Creating the database
Downloading the data files
--2023-10-05 15:40:29--
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Setting%20up%20a%20staging%20area/billing-datawareh
ouse.tgz
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)...
169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
(cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud)|169.63.118.104
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 944578 (922K) [application/x-tar]
Saving to: 'billing-datawarehouse.tgz'

billing-datawarehouse.tgz
100%[=====================================================================>]
922.44K  --.-KB/s    in 0.008s

2023-10-05 15:40:29 (108 MB/s) - 'billing-datawarehouse.tgz' saved
[944578/944578]

Extracting files
DimCustomer.sql
DimMonth.sql
FactBilling.sql
star-schema.sql
verify.sql
Creating schema
BEGIN
CREATE TABLE
CREATE TABLE
CREATE TABLE
ALTER TABLE
```

```
ALTER TABLE
COMMIT
Loading data
INSERT 0 1000
INSERT 0 132
INSERT 0 132000
Finished loading data
Verifying data
"Checking row in DimMonth Table"
 count
-------
   132
(1 row)

"Checking row in DimCustomer Table"
 count
-------
  1000
(1 row)

"Checking row in FactBilling Table"
 count
--------
 132000
(1 row)

Successfully setup the staging area
```

- See the set staging data for the sql scripts which were also in the tar file … not including here again
    - https://github.com/craigtrupp/ibm-deng/tree/main/DataWarehousing_BIAnalytics/Labs/Set_Staging_Area

Exercise 2 - Getting the testing framework ready

You can perform most of the data quality checks by manually running sql queries on the data warehouse.

It is a good idea to automate these checks using custom programs or tools. Automation helps you to easily

- create new tests,
- run tests,
- and schedule tests.

We will be using a python based framework to run the data quality tests.

Step 1 : Download Framework

```
wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
ouse/dataqualitychecks.py

wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
ouse/dbconnect.py

wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
ouse/mytests.py

wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Verifying%20Data%20Quality%20for%20a%20Data%20Wareh
ouse/generate-data-quality-report.py

ls
```

- This brings in scripts I'll push to github - have fixed any import errors for importing functions from files at same directory level
- https://medium.com/geekculture/how-to-import-another-file-in-python-4f833ea462b1

Step 2: Install the python driver for Postgresql.

Run the command below to install the python driver for Postgresql database

```
python3 -m pip install psycopg2
```

```
theia@theiadocker-craigtrupp8:/home/project$ python3 -m pip install
psycopg2
Collecting psycopg2
  Cache entry deserialization failed, entry ignored
  Downloading
https://files.pythonhosted.org/packages/9e/78/3b15ee8bbbf36f8bace9b0e6fe8a7
481372650c76bcf1a7de1ed723cce96/psycopg2-2.9.8.tar.gz (383kB)
    100% |████████████████████████████████| 389kB 3.6MB/s
Building wheels for collected packages: psycopg2
  Running setup.py bdist_wheel for psycopg2 ... done
  Stored in directory:
/home/theia/.cache/pip/wheels/d6/c2/f5/7cac48f8833bc6a2b7327dd5f3d69e65174f
fe78ddd1d4d10e
Successfully built psycopg2
Installing collected packages: psycopg2
Successfully installed psycopg2-2.9.8
```

Step 3: Test database connectivity.

Now we need to check

- if the Postgresql python driver is installed properly.
- if Postgresql server is up and running.
- if our micro framework can connect to the database.

dbconnect.py script

```
import os
import psycopg2
pgpassword = os.environ.get('POSTGRES_PASSWORD')
conn = None
try:
      conn = psycopg2.connect(
            user = "postgres",
          password = pgpassword,
          host = "localhost",
          port = "5432",
```

```
          database = "postgres")
except Exception as e:
      print("Error connecting to data warehouse")
      print(e)
else:
      print("Successfully connected to warehouse")
finally:
      if conn:
            conn.close()
            print("Connection closed")
```

Run Script from right file directory

```
theia@theiadocker-craigtrupp8:/home/project$ python3 dbconnect.py
Successfully connected to warehouse
Connection closed
theia@theiadocker-craigtrupp8:/home/project$
```

Exercise 3 - Create a sample data quality report

Run the command below to install pandas.

```
python3 -m pip install pandas tabulate
```

Run the command below to generate a sample data quality report.

```
python3 generate-data-quality-report.py
```

- You should see a list of tests that were run and their status.

```
theia@theiadocker-craigtrupp8:/home/project$ python3 -m pip install pandas
tabulate
Collecting pandas
```

```
  Downloading
https://files.pythonhosted.org/packages/c3/e2/00cacecafbab071c787019f00ad84
ca3185952f6bb9bca9550ed83870d4d/pandas-1.1.5-cp36-cp36m-manylinux1_x86_64.w
hl (9.5MB)
    100% |████████████████████████████████| 9.5MB 169kB/s
Collecting tabulate
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading
https://files.pythonhosted.org/packages/92/4e/e5a13fdb3e6f81ce11893523ff289
870c87c8f1f289a7369fb0e9840c3bb/tabulate-0.8.10-py3-none-any.whl
Collecting numpy>=1.15.4 (from pandas)
  Downloading
https://files.pythonhosted.org/packages/45/b2/6c7545bb7a38754d63048c7696804
a0d947328125d81bf12beaa692c3ae3/numpy-1.19.5-cp36-cp36m-manylinux1_x86_64.w
hl (13.4MB)
    100% |████████████████████████████████| 13.4MB 123kB/s
Collecting pytz>=2017.2 (from pandas)
  Cache entry deserialization failed, entry ignored
  Downloading
https://files.pythonhosted.org/packages/32/4d/aaf7eff5deb402fd9a24a1449a811
9f00d74ae9c2efa79f8ef9994261fc2/pytz-2023.3.post1-py2.py3-none-any.whl
(502kB)
    100% |████████████████████████████████| 512kB 3.4MB/s
Collecting python-dateutil>=2.7.3 (from pandas)
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading
https://files.pythonhosted.org/packages/36/7a/87837f39d0296e723bb9b62bbb257
d0355c7f6128853c78955f57342a56d/python_dateutil-2.8.2-py2.py3-none-any.whl
(247kB)
    100% |████████████████████████████████| 256kB 6.3MB/s
Collecting six>=1.5 (from python-dateutil>=2.7.3->pandas)
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading
https://files.pythonhosted.org/packages/d9/5a/e7c31adbe875f2abbb91bd84cf2dc
52d792b5a01506781dbcf25c91daf11/six-1.16.0-py2.py3-none-any.whl
Installing collected packages: numpy, pytz, six, python-dateutil, pandas,
tabulate
Successfully installed numpy-1.19.5 pandas-1.1.5 python-dateutil-2.8.2
pytz-2023.3.post1 six-1.16.0 tabulate-0.8.10
theia@theiadocker-craigtrupp8:/home/project$ python3
```

```
generate-data-quality-report.py
Connected to data warehouse
************************************************
Thu Oct  5 15:57:06 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.0030155181884765625
Thu Oct  5 15:57:06 2023
************************************************
************************************************
Thu Oct  5 15:57:06 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = month
table = DimMonth
minimum = 1
maximum = 12

Duration :  0.001041412353515625
Thu Oct  5 15:57:06 2023
************************************************
************************************************
Thu Oct  5 15:57:06 2023
Starting test Check for valid values
{'Company', 'Individual'}
Finished test Check for valid values
Test Passed True
Test Parameters
column = category
table = DimCustomer
valid_values = {'Company', 'Individual'}

Duration :  0.0023987293243408203
Thu Oct  5 15:57:06 2023
************************************************
************************************************
```

```
Thu Oct  5 15:57:06 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.0013823509216308594
Thu Oct  5 15:57:06 2023
************************************************
+----+------------------------+-------------+----------+---------------+
|    | Test Name              | Table       | Column   | Test Passed   |
|----+------------------------+-------------+----------+---------------|
|  1 | Check for nulls        | DimMonth    | monthid  | True          |
|  2 | Check for min and max  | DimMonth    | month    | True          |
|  3 | Check for valid values | DimCustomer | category | True          |
|  4 | Check for duplicates   | DimMonth    | monthid  | True          |
+----+------------------------+-------------+----------+---------------+
Disconnected from data warehouse
```

Script details

```
import os
import psycopg2
import pandas as pd
from tabulate import tabulate

import mytests
# import the data quality checks
from dataqualitychecks import check_for_nulls
from dataqualitychecks import check_for_min_max
from dataqualitychecks import check_for_valid_values
from dataqualitychecks import check_for_duplicates
from dataqualitychecks import run_data_quality_check

# connect to database
pgpassword = os.environ.get('POSTGRES_PASSWORD')
conn = psycopg2.connect(
            user = "postgres",
          password = pgpassword,
```

```
          host = "localhost",
          port = "5432",
          database = "billingDW")

print("Connected to data warehouse")

#Start of data quality checks
results = []
tests = {key:value for key,value in mytests.__dict__.items() if
key.startswith('test')}
for testname,test in tests.items():
    test['conn'] = conn
    results.append(run_data_quality_check(**test))

#print(results)
df=pd.DataFrame(results)
df.index+=1
df.columns = ['Test Name', 'Table','Column','Test Passed']
print(tabulate(df,headers='keys',tablefmt='psql'))
#End of data quality checks
conn.close()
print("Disconnected from data warehouse")
```

Exercise 4 - Explore Data Quality Tests

The file mytests.py contains all the data quality tests.

It provides a quick and easy way to author and run new data quality tests.

The testing framework provides the following tests:

- check_for_nulls - this test will check for nulls in a column
- check_for_min_max - this test will check if the values in a column are with a range of min and max values
- check_for_valid_values - this test will check for any invalid values in a column
- check_for_duplicates - this test will check for duplicates in a column

Each test can be authored by mentioning a minimum of 4 parameters.

testname - The human readable name of the test for reporting purposes
test - The actual test name that the testing micro framework provides
table - The table name on which the test is to be performed
column - The table name on which the test is to be performed

```
from dataqualitychecks import check_for_nulls
from dataqualitychecks import check_for_min_max
from dataqualitychecks import check_for_valid_values
from dataqualitychecks import check_for_duplicates


test1={
      "testname":"Check for nulls",
      "test":check_for_nulls,
      "column": "monthid",
      "table": "DimMonth"
}


test2={
      "testname":"Check for min and max",
      "test":check_for_min_max,
      "column": "month",
      "table": "DimMonth",
      "minimum":1,
      "maximum":12
}


test3={
      "testname":"Check for valid values",
      "test":check_for_valid_values,
      "column": "category",
      "table": "DimCustomer",
      "valid_values":{'Individual','Company'}
}


test4={
      "testname":"Check for duplicates",
```

```
    "test":check_for_duplicates,
    "column": "monthid",
    "table": "DimMonth"
}
```

Exercise 5 - Check for nulls

Let us now see what a `check_for_nulls` test looks like.

Here is a sample `check_for_nulls` test:

```
test1={
    "testname":"Check for nulls",
    "test":check_for_nulls,
    "column": "monthid",
    "table": "DimMonth"
}
```

All tests must be named as `test` following by a unique number to identify the test.

- Give an easy to understand description for `testname`
- mention `check_for_nulls` for `test`
- mention the column name on which you wish to check for nulls
- mention the table name where this column exists

Let us now create a new `check_for_nulls` test and run it.

The test below checks if there are any null values in the column `year` in the table `DimMonth` .

The test fails if nulls exist.

- Append to end of mytests.py file

```
test5={
    "testname":"Check for nulls",
    "test":check_for_nulls,
    "column": "year",
    "table": "DimMonth"
}
```

```
theia@theiadocker-craigtrupp8:/home/project$ python3
generate-data-quality-report.py
Connected to data warehouse
**************************************************
Thu Oct  5 16:02:17 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.0023932456970214844
Thu Oct  5 16:02:17 2023
**************************************************
**************************************************
Thu Oct  5 16:02:17 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = month
table = DimMonth
minimum = 1
maximum = 12

Duration :  0.0007867813110351562
Thu Oct  5 16:02:17 2023
**************************************************
**************************************************
Thu Oct  5 16:02:17 2023
Starting test Check for valid values
{'Individual', 'Company'}
Finished test Check for valid values
Test Passed True
```

```
Test Parameters
column = category
table = DimCustomer
valid_values = {'Individual', 'Company'}

Duration :  0.002331972122192383
Thu Oct  5 16:02:17 2023
**************************************************
**************************************************
Thu Oct  5 16:02:17 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.001264333724975586
Thu Oct  5 16:02:17 2023
**************************************************
**************************************************
Thu Oct  5 16:02:17 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = year
table = DimMonth

Duration :  0.0005178451538085938
Thu Oct  5 16:02:17 2023
**************************************************
+----+------------------------+-------------+----------+---------------+
|    | Test Name              | Table       | Column   | Test Passed   |
|----+------------------------+-------------+----------+---------------|
|  1 | Check for nulls        | DimMonth    | monthid  | True          |
|  2 | Check for min and max  | DimMonth    | month    | True          |
|  3 | Check for valid values | DimCustomer | category | True          |
|  4 | Check for duplicates   | DimMonth    | monthid  | True          |
|  5 | Check for nulls        | DimMonth    | year     | True          |
+----+------------------------+-------------+----------+---------------+
Disconnected from data warehouse
theia@theiadocker-craigtrupp8:/home/project$
```

Exercise 6 - Check for min max range

Let us now see what a `check_for_min_max` test looks like.

Here is a sample `check_for_min_max` test

```
test2={
    "testname":"Check for min and max",
    "test":check_for_min_max,
    "column": "monthid",
    "table": "DimMonth",
    "minimum":1,
    "maximum":12
}
```

In addition to the usual fields, you have two more fields here.

- minimum is the lowest valid value for this column. (Example 1 in case of month number)
- maximum is the highest valid value for this column. (Example 12 in case of month number)

Let us now create a new `check_for_min_max` test and run it.

The test below checks for minimum of 1 and maximum of 4 in the column `quarter` in the table `DimMonth` .

The test fails if there any values less than minimum or more than maximum.

- Append to mytest.py file for test6

```
test6={
    "testname":"Check for min and max",
    "test":check_for_min_max,
    "column": "quarter",
    "table": "DimMonth",
```

```
    "minimum":1,
    "maximum":4
}
```

```
theia@theiadocker-craigtrupp8:/home/project$ python3
generate-data-quality-report.py
Connected to data warehouse
**************************************************
Thu Oct  5 16:06:03 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.004159212112426758
Thu Oct  5 16:06:03 2023
**************************************************
**************************************************
Thu Oct  5 16:06:03 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = month
table = DimMonth
minimum = 1
maximum = 12

Duration :  0.0009131431579589844
Thu Oct  5 16:06:03 2023
**************************************************
**************************************************
Thu Oct  5 16:06:03 2023
Starting test Check for valid values
{'Individual', 'Company'}
Finished test Check for valid values
Test Passed True
Test Parameters
```

```
column = category
table = DimCustomer
valid_values = {'Individual', 'Company'}

Duration :  0.003124237060546875
Thu Oct  5 16:06:03 2023
************************************************
************************************************
Thu Oct  5 16:06:03 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.001790761947631836
Thu Oct  5 16:06:03 2023
************************************************
************************************************
Thu Oct  5 16:06:03 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = year
table = DimMonth

Duration :  0.000985860824584961
Thu Oct  5 16:06:03 2023
************************************************
************************************************
Thu Oct  5 16:06:03 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = quarter
table = DimMonth
minimum = 1
maximum = 4

Duration :  0.0007839202880859375
```

```
Thu Oct  5 16:06:03 2023
************************************************
+----+------------------------+-------------+----------+---------------+
|    | Test Name              | Table       | Column   | Test Passed   |
|----+------------------------+-------------+----------+---------------|
|  1 | Check for nulls        | DimMonth    | monthid  | True          |
|  2 | Check for min and max  | DimMonth    | month    | True          |
|  3 | Check for valid values | DimCustomer | category | True          |
|  4 | Check for duplicates   | DimMonth    | monthid  | True          |
|  5 | Check for nulls        | DimMonth    | year     | True          |
|  6 | Check for min and max  | DimMonth    | quarter  | True          |
+----+------------------------+-------------+----------+---------------+
Disconnected from data warehouse
```

Exercise 7 - Check for any invalid entries

Let us now see what a `check_for_valid_values` test looks like.

Here is a sample `check_for_valid_values` test:

```
test3={
    "testname":"Check for valid values",
    "test":check_for_valid_values,
    "column": "category",
    "table": "DimCustomer",
    "valid_values":{'Individual','Company'}
}
```

In addition to the usual fields, you have an additional field here.

- use the field `valid_values` to mention what are the valid values for this column.

Let us now create a new `check_for_valid_values` test and run it.

The test below checks for valid values in the column `quartername` in the table `DimMonth` .

The valid values are Q1,Q2,Q3,Q4

The test fails if there any values less than minimum or more than maximum.

- Append to mytest.py for new test object

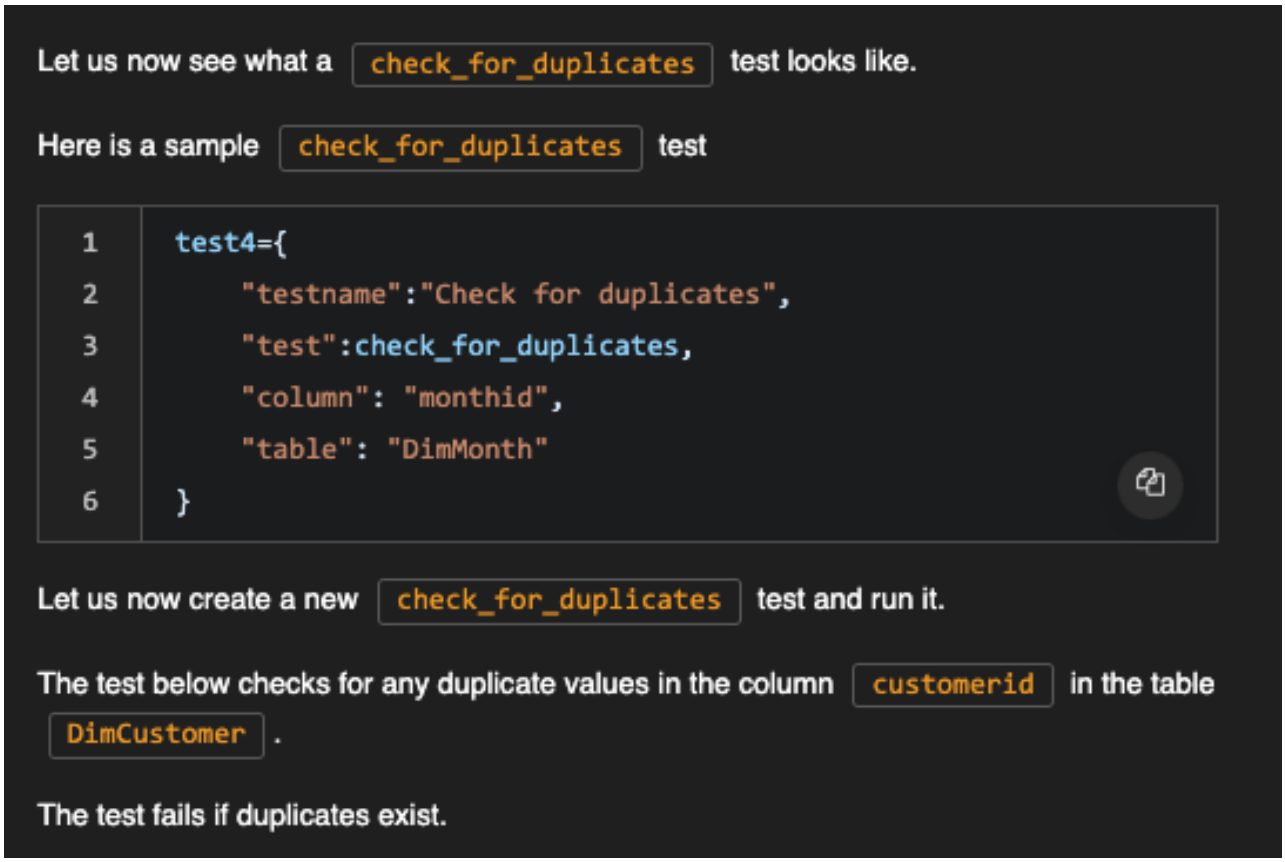
```
test7={
    "testname":"Check for valid values",
    "test":check_for_valid_values,
    "column": "quartername",
    "table": "DimMonth",
    "valid_values":{'Q1','Q2','Q3','Q4'}
}
```

```
theia@theiadocker-craigtrupp8:/home/project$ python3
generate-data-quality-report.py
Connected to data warehouse
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.0029528141021728516
Thu Oct  5 16:10:49 2023
************************************************
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = month
table = DimMonth
minimum = 1
maximum = 12

Duration :  0.0010571479797363281
Thu Oct  5 16:10:49 2023
************************************************
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for valid values
{'Individual', 'Company'}
Finished test Check for valid values
Test Passed True
Test Parameters
column = category
table = DimCustomer
valid_values = {'Company', 'Individual'}

Duration :  0.0024480819702148438
Thu Oct  5 16:10:49 2023
************************************************
```

```
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = monthid
table = DimMonth

Duration :  0.0013585090637207031
Thu Oct  5 16:10:49 2023
************************************************
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = year
table = DimMonth

Duration :  0.0006077289581298828
Thu Oct  5 16:10:49 2023
************************************************
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = quarter
table = DimMonth
minimum = 1
maximum = 4

Duration :  0.000640869140625
Thu Oct  5 16:10:49 2023
************************************************
************************************************
Thu Oct  5 16:10:49 2023
Starting test Check for valid values
{'Q3', 'Q1', 'Q4', 'Q2'}
Finished test Check for valid values
```

```
Test Passed True
Test Parameters
column = quartername
table = DimMonth
valid_values = {'Q3', 'Q1', 'Q4', 'Q2'}

Duration :  0.0006723403930664062
Thu Oct  5 16:10:49 2023
************************************************
+----+------------------------+-------------+-------------+---------------+
|    | Test Name              | Table       | Column      | Test Passed   |
|----+------------------------+-------------+-------------+---------------|
|  1 | Check for nulls        | DimMonth    | monthid     | True          |
|  2 | Check for min and max  | DimMonth    | month       | True          |
|  3 | Check for valid values | DimCustomer | category    | True          |
|  4 | Check for duplicates   | DimMonth    | monthid     | True          |
|  5 | Check for nulls        | DimMonth    | year        | True          |
|  6 | Check for min and max  | DimMonth    | quarter     | True          |
|  7 | Check for valid values | DimMonth    | quartername | True          |
+----+------------------------+-------------+-------------+---------------+
Disconnected from data warehouse
```

Exercise 8 - Check for duplicate entries




```
test8={
    "testname":"Check for duplicates",
    "test":check_for_duplicates,
    "column": "customerid",
    "table": "DimCustomer"
}
```

```
theia@theiadocker-craigtrupp8:/home/project$ python3
generate-data-quality-report.py
Connected to data warehouse
**************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
```

```
column = monthid
table = DimMonth

Duration :  0.0025262832641601562
Thu Oct  5 16:12:42 2023
************************************************
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = month
table = DimMonth
minimum = 1
maximum = 12

Duration :  0.0011775493621826172
Thu Oct  5 16:12:42 2023
************************************************
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for valid values
{'Company', 'Individual'}
Finished test Check for valid values
Test Passed True
Test Parameters
column = category
table = DimCustomer
valid_values = {'Company', 'Individual'}

Duration :  0.0026252269744873047
Thu Oct  5 16:12:42 2023
************************************************
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = monthid
table = DimMonth
```

```
Duration :  0.0013098716735839844
Thu Oct  5 16:12:42 2023
************************************************
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = year
table = DimMonth

Duration :  0.0004940032958984375
Thu Oct  5 16:12:42 2023
************************************************
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = quarter
table = DimMonth
minimum = 1
maximum = 4

Duration :  0.0005514621734619141
Thu Oct  5 16:12:42 2023
************************************************
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for valid values
{'Q2', 'Q4', 'Q3', 'Q1'}
Finished test Check for valid values
Test Passed True
Test Parameters
column = quartername
table = DimMonth
valid_values = {'Q2', 'Q4', 'Q3', 'Q1'}

Duration :  0.0005536079406738281
Thu Oct  5 16:12:42 2023
************************************************
```

```
************************************************
Thu Oct  5 16:12:42 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = customerid
table = DimCustomer

Duration :  0.0011513233184814453
Thu Oct  5 16:12:42 2023
************************************************
+----+------------------------+-------------+-------------+---------------+
|    | Test Name              | Table       | Column      | Test Passed   |
|----+------------------------+-------------+-------------+---------------|
|  1 | Check for nulls        | DimMonth    | monthid     | True          |
|  2 | Check for min and max  | DimMonth    | month       | True          |
|  3 | Check for valid values | DimCustomer | category    | True          |
|  4 | Check for duplicates   | DimMonth    | monthid     | True          |
|  5 | Check for nulls        | DimMonth    | year        | True          |
|  6 | Check for min and max  | DimMonth    | quarter     | True          |
|  7 | Check for valid values | DimMonth    | quartername | True          |
|  8 | Check for duplicates   | DimCustomer | customerid  | True          |
+----+------------------------+-------------+-------------+---------------+
Disconnected from data warehouse
```

Practice exercises

- You can check the methods in the **dataqualitychecks.py** file to see how the arguments are being passed and how to satisfy the objects in the **mytests.py** file for the practice tests.
- The **generate-data-quality-report.py** has all the requisite imports as well as a dictionary comprehension from the mytests file to pass all the tests to

```
#Start of data quality checks
results = []
tests = {key:value for key,value in mytests.__dict__.items() if
key.startswith('test')}
for testname,test in tests.items():
    test['conn'] = conn
```

```
    results.append(run_data_quality_check(**test))
```

1. Create a **check_for_nulls** test on column **billedamount** in the table **FactBilling**
2. Create a **check_for_duplicates** test on column **billid** in the table **FactBilling**
3. Create a **check_for_valid_values** test on column **quarter** in the table **DimMonth**.
    a. The valid values are 1, 2, 3, 4

- New tests

```
test9={
    "testname":"Check for nulls",
    "test":check_for_nulls,
    "column":"billedamount",
    "table":"FactBilling"
}

test10={
    "testname":"Check for duplicates",
    "test":check_for_duplicates,
    "column":"billid",
    "table":"FactBilling"
}

test11={
    "testname":"Check for valid values",
    "test":check_for_valid_values,
    "column":"quarter",
    "table":"DimMonth",
    "valid_values":{1,2,3,4}
}
```

```
theia@theiadocker-craigtrupp8:/home/project$ python3
generate-data-quality-report.py
Connected to data warehouse
**********************************************
Thu Oct  5 16:30:36 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
```

```
Test Parameters
column = monthid
table = DimMonth

Duration :  0.0023508071899414062
Thu Oct  5 16:30:36 2023
**************************************************
**************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = month
table = DimMonth
minimum = 1
maximum = 12

Duration :  0.0008423328399658203
Thu Oct  5 16:30:36 2023
**************************************************
**************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for valid values
{'Company', 'Individual'}
Finished test Check for valid values
Test Passed True
Test Parameters
column = category
table = DimCustomer
valid_values = {'Company', 'Individual'}

Duration :  0.0024466514587402344
Thu Oct  5 16:30:36 2023
**************************************************
**************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = monthid
table = DimMonth
```

```
Duration :  0.0013043880462646484
Thu Oct  5 16:30:36 2023
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = year
table = DimMonth

Duration :  0.0006489753723144531
Thu Oct  5 16:30:36 2023
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for min and max
Finished test Check for min and max
Test Passed True
Test Parameters
column = quarter
table = DimMonth
minimum = 1
maximum = 4

Duration :  0.0005788803100585938
Thu Oct  5 16:30:36 2023
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for valid values
{'Q3', 'Q4', 'Q2', 'Q1'}
Finished test Check for valid values
Test Passed True
Test Parameters
column = quartername
table = DimMonth
valid_values = {'Q3', 'Q4', 'Q2', 'Q1'}

Duration :  0.0006229877471923828
Thu Oct  5 16:30:36 2023
```

```
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = customerid
table = DimCustomer

Duration :  0.0013325214385986328
Thu Oct  5 16:30:36 2023
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for nulls
Finished test Check for nulls
Test Passed True
Test Parameters
column = billedamount
table = FactBilling

Duration :  0.015103816986083984
Thu Oct  5 16:30:36 2023
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for duplicates
Finished test Check for duplicates
Test Passed True
Test Parameters
column = billid
table = FactBilling

Duration :  0.031942129135131836
Thu Oct  5 16:30:36 2023
************************************************
************************************************
Thu Oct  5 16:30:36 2023
Starting test Check for valid values
{1, 2, 3, 4}
Finished test Check for valid values
Test Passed True
```

```
Test Parameters
column = quarter
table = DimMonth
valid_values = {1, 2, 3, 4}

Duration :  0.0006504058837890625
Thu Oct  5 16:30:36 2023
**********************************************
+----+-----------------------+------------+-------------+---------------
+
|    | Test Name             | Table      | Column      | Test Passed
|
|----+-----------------------+------------+-------------+---------------
|
|  1 | Check for nulls       | DimMonth   | monthid     | True
|
|  2 | Check for min and max | DimMonth   | month       | True
|
|  3 | Check for valid values | DimCustomer | category    | True
|
|  4 | Check for duplicates  | DimMonth   | monthid     | True
|
|  5 | Check for nulls       | DimMonth   | year        | True
|
|  6 | Check for min and max | DimMonth   | quarter     | True
|
|  7 | Check for valid values | DimMonth   | quartername | True
|
|  8 | Check for duplicates  | DimCustomer | customerid  | True
|
|  9 | Check for nulls       | FactBilling | billedamount | True
|
| 10 | Check for duplicates  | FactBilling | billid      | True
|
| 11 | Check for valid values | DimMonth   | quarter     | True
|
+----+-----------------------+------------+-------------+---------------
+
Disconnected from data warehouse
```