

pg_example_queries

May 19, 2019

1 Project 03: Data Warehouse

```
[1]: import configparser
import psycopg2
%reload_ext sql

[2]: config = configparser.ConfigParser()
config.read('pg_dwh.cfg')

db_host = config['CLUSTER']['HOST']
db_port = config['CLUSTER']['DB_PORT']
db_name = config['CLUSTER']['DB_NAME']
db_user = config['CLUSTER']['DB_USER']
db_pass = config['CLUSTER']['DB_PASSWORD']

connection_string = "postgresql://{user}:{password}@{host}:{port}/{name}".
    ↳format(user=db_user, password=db_pass, host=db_host, port=db_port,
    ↳name=db_name)
print('connection_string=%s' % connection_string)

%sql $connection_string
```

```
connection_string=postgresql://sparkify:sparkify@127.0.0.1:5432/sparkify
```

```
[2]: 'Connected: sparkify@sparkify'
```

```
[4]: %%sql
-- what are the free and paid user counts by location?
select f.location, f.level, count(level)
from f_songplay f
group by f.location, f.level
order by f.location, f.level;
```

```
* postgresql://sparkify:***@127.0.0.1:5432/sparkify
76 rows affected.
```

[4]: [('Atlanta-Sandy Springs-Roswell, GA', 'free', 28),
('Atlanta-Sandy Springs-Roswell, GA', 'paid', 428),
('Augusta-Richmond County, GA-SC', 'paid', 140),
('Birmingham-Hoover, AL', 'free', 15),
('Birmingham-Hoover, AL', 'paid', 208),
('Cedar Rapids, IA', 'free', 10),
('Chicago-Naperville-Elgin, IL-IN-WI', 'free', 13),
('Chicago-Naperville-Elgin, IL-IN-WI', 'paid', 462),
('Columbia, SC', 'free', 25),
('Dallas-Fort Worth-Arlington, TX', 'free', 21),
('Detroit-Warren-Dearborn, MI', 'free', 4),
('Detroit-Warren-Dearborn, MI', 'paid', 72),
('Elkhart-Goshen, IN', 'free', 2),
('Eugene, OR', 'free', 20),
('Eureka-Arcata-Fortuna, CA', 'free', 16),
('Harrisburg-Carlisle, PA', 'free', 37),
('Houston-The Woodlands-Sugar Land, TX', 'free', 66),
('Indianapolis-Carmel-Anderson, IN', 'free', 14),
('Janesville-Beloit, WI', 'free', 7),
('Janesville-Beloit, WI', 'paid', 241),
('Klamath Falls, OR', 'free', 20),
('La Crosse-Onalaska, WI-MN', 'free', 45),
('Lake Havasu City-Kingman, AZ', 'paid', 321),
('Lansing-East Lansing, MI', 'paid', 557),
('Las Cruces, NM', 'free', 6),
('London, KY', 'free', 5),
('Longview, TX', 'paid', 17),
('Los Angeles-Long Beach-Anaheim, CA', 'free', 6),
('Lubbock, TX', 'free', 27),
('Marinette, WI-MI', 'paid', 169),
('Miami-Fort Lauderdale-West Palm Beach, FL', 'free', 2),
('Milwaukee-Waukesha-West Allis, WI', 'free', 9),
('Minneapolis-St. Paul-Bloomington, MN-WI', 'free', 15),
('Myrtle Beach-Conway-North Myrtle Beach, SC-NC', 'free', 1),
('Nashville-Davidson--Murfreesboro--Franklin, TN', 'free', 25),
('New Haven-Milford, CT', 'free', 48),
('New Orleans-Metairie, LA', 'free', 55),
('New York-Newark-Jersey City, NY-NJ-PA', 'free', 113),
('New York-Newark-Jersey City, NY-NJ-PA', 'paid', 149),
('Ogden-Clearfield, UT', 'free', 7),
('Oxnard-Thousand Oaks-Ventura, CA', 'free', 7),
('Palestine, TX', 'free', 27),
('Parkersburg-Vienna, WV', 'free', 5),
('Pensacola-Ferry Pass-Brent, FL', 'free', 3),
('Philadelphia-Camden-Wilmington, PA-NJ-DE-MD', 'free', 35),
('Phoenix-Mesa-Scottsdale, AZ', 'free', 27),
('Plymouth, IN', 'free', 10),

```
(('Portland-South Portland, ME', 'free', 17),
('Portland-South Portland, ME', 'paid', 648),
('Portland-Vancouver-Hillsboro, OR-WA', 'free', 7),
('Raleigh, NC', 'free', 4),
('Red Bluff, CA', 'free', 23),
('Red Bluff, CA', 'paid', 178),
('Richmond, VA', 'free', 13),
('Sacramento--Roseville--Arden-Arcade, CA', 'free', 29),
('Sacramento--Roseville--Arden-Arcade, CA', 'paid', 241),
('Saginaw, MI', 'free', 3),
('Salinas, CA', 'free', 37),
('Salt Lake City, UT', 'free', 4),
('San Antonio-New Braunfels, TX', 'free', 19),
('San Antonio-New Braunfels, TX', 'paid', 33),
('San Diego-Carlsbad, CA', 'free', 5),
('San Francisco-Oakland-Hayward, CA', 'free', 41),
('San Francisco-Oakland-Hayward, CA', 'paid', 650),
('San Jose-Sunnyvale-Santa Clara, CA', 'free', 114),
('San Jose-Sunnyvale-Santa Clara, CA', 'paid', 178),
('Santa Rosa, CA', 'free', 20),
('Seattle-Tacoma-Bellevue, WA', 'free', 14),
('St. Louis, MO-IL', 'free', 16),
('Tampa-St. Petersburg-Clearwater, FL', 'free', 18),
('Tampa-St. Petersburg-Clearwater, FL', 'paid', 289),
('Washington-Arlington-Alexandria, DC-VA-MD-WV', 'free', 34),
('Waterloo-Cedar Falls, IA', 'paid', 397),
('Winston-Salem, NC', 'paid', 213),
('Youngstown-Warren-Boardman, OH-PA', 'free', 8),
('Yuba City, CA', 'free', 27)]
```

```
[5]: %%sql
-- what are the top 10 most played artists
select
    row_number() over (order by count(f.artist_id) desc) as rank,
    d1.name as artist_name
from f_songplay f
join d_artist d1 on d1.artist_id = f.artist_id
where f.artist_id is not null
group by f.artist_id, d1.name
order by count(f.artist_id) desc
limit 10;
```

```
* postgresql://sparkify:***@127.0.0.1:5432/sparkify
10 rows affected.
```

```
[5]: [(1, 'Coldplay'),
(2, 'Kings Of Leon'),
(3, 'Dwight Yoakam'),
```

```
(4, 'The Black Keys'),
(5, 'Muse'),
(6, 'Jack Johnson'),
(7, 'The Killers'),
(8, 'John Mayer'),
(9, 'Radiohead'),
(10, 'Alliance Ethnik')]
```

```
[6]: %sql
-- what are the top 10 most played artists by gender
with
female_top10_artists as (
    select
        row_number() over (order by count(f.artist_id) desc, d1.gender, d2.
→name) as rank,
        d1.gender,
        d2.name
    from f_songplay f
    join d_user d1 on d1.user_id = f.user_id
    join d_artist d2 on d2.artist_id = f.artist_id
    where f.artist_id is not null
    and d1.gender = 'F'
    group by d1.gender, d2.name
    order by count(f.artist_id) desc, d1.gender, d2.name
    limit 10
),
male_top10_artists as (
    select
        row_number() over (order by count(f.artist_id) desc, d1.gender, d2.
→name) as rank,
        d1.gender,
        d2.name
    from f_songplay f
    join d_user d1 on d1.user_id = f.user_id
    join d_artist d2 on d2.artist_id = f.artist_id
    where f.artist_id is not null
    and d1.gender = 'M'
    group by d1.gender, d2.name
    order by count(f.artist_id) desc, d1.gender, d2.name
    limit 10
)
select F.rank, F.name as female, M.name as male
from female_top10_artists as F
join male_top10_artists as M on F.rank = M.rank;
```

```
* postgresql://sparkify:***@127.0.0.1:5432/sparkify
10 rows affected.
```

```
[6]: [(1, 'Coldplay', 'Coldplay'),
      (2, 'Kings Of Leon', 'Kings Of Leon'),
      (3, 'Dwight Yoakam', 'The Black Keys'),
      (4, 'Alliance Ethnik', 'Metallica'),
      (5, 'Jack Johnson', 'Muse'),
      (6, 'Linkin Park', 'Jack Johnson'),
      (7, 'Muse', 'John Mayer'),
      (8, 'Kanye West', 'Dwight Yoakam'),
      (9, 'OneRepublic', 'Radiohead'),
      (10, 'The Killers', 'The Killers')]
```

```
[7]: %%sql
-- where are most Colplay song plays occuring?
select count(f.location), f.location
from f_songplay f
join d_artist d1 on d1.artist_id = f.artist_id
where d1.name = 'Coldplay'
group by f.location
order by count(f.location) desc
limit 10;
```

```
* postgresql://sparkify:***@127.0.0.1:5432/sparkify
10 rows affected.
```

```
[7]: [(10, 'Lansing-East Lansing, MI'),
      (7, 'San Francisco-Oakland-Hayward, CA'),
      (5, 'Lake Havasu City-Kingman, AZ'),
      (4, 'Augusta-Richmond County, GA-SC'),
      (4, 'Portland-South Portland, ME'),
      (3, 'Chicago-Naperville-Elgin, IL-IN-WI'),
      (3, 'Janesville-Beloit, WI'),
      (3, 'Nashville-Davidson--Murfreesboro--Franklin, TN'),
      (2, 'Red Bluff, CA'),
      (2, 'Atlanta-Sandy Springs-Roswell, GA')]
```

```
[8]: %%sql
-- where are most Kings of Leon song plays occuring?
select count(f.location), f.location
from f_songplay f
join d_artist d1 on d1.artist_id = f.artist_id
where d1.name = 'Kings Of Leon'
group by f.location
order by count(f.location) desc
limit 10;
```

```
* postgresql://sparkify:***@127.0.0.1:5432/sparkify
10 rows affected.
```

```
[8]: [(9, 'San Francisco-Oakland-Hayward, CA'),  
      (9, 'Atlanta-Sandy Springs-Roswell, GA'),  
      (7, 'Lansing-East Lansing, MI'),  
      (5, 'Chicago-Naperville-Elgin, IL-IN-WI'),  
      (4, 'Waterloo-Cedar Falls, IA'),  
      (3, 'New York-Newark-Jersey City, NY-NJ-PA'),  
      (3, 'Portland-South Portland, ME'),  
      (3, 'Lake Havasu City-Kingman, AZ'),  
      (2, 'Dallas-Fort Worth-Arlington, TX'),  
      (2, 'Janesville-Beloit, WI')]
```

```
[ ]:
```