Provider Table

provider_id	provider_name	Created
1	John Smith	3/4/2009 12:45:00
2	Michael Rowe	5/3/2010 15:34:34
3	Gregory Adams	4/5/2011 23:44:59

Profession Table

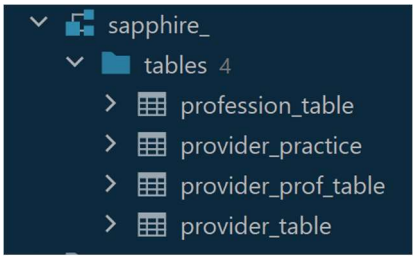
profession_id	Profession_degree
1	DO
2	MD

provider_profession table

provider_id	profession_id
1	1
1	2
2	2
3	1
3	2

Provider_practice table

Provider_practice_id	Provider_id	City
1	1	Morristown
2	2	Passaic
3	2	Jersey City
4	1	Trenton
5	2	East Rutherford
6	3	Rutherford
7	1	Lyndhurst



Queries

Please write a SQL query to satisfy the following:.

1. List all the providers, ordered by provider name. While outputting the names of all the providers, if the name of the provider contains the word 'John', output 'Johnathan' instead of 'John' within the same output field of provider_name.

```
a. select provider_id, replace(provider_name, 'John', 'Johnathan')
   provider_name, created
   from sapphire_.provider_table
   order by provider_name
```

2. List out the degrees and the counts that are associated to the degree. If the degree is listed more than twice, output 'Popular degree'.

```
a. select profession_degree, count(profession_degree) degree_count,
          case when count(profession_degree) >= 2 then 'Popular degree' else ''
   end degree_popularity
   from sapphire_.provider_prof_table prvprof
   left join sapphire_.profession_table prof
   on prvprof.profession_id=prof.profession_id
   group by profession_degree
```

3. How many providers have a profession of DO and is associated with a city that contains 'er' in its name?

```
a. select *
   from sapphire_.profession_table pft
   left join sapphire_.provider_prof_table ppt
   on pft.profession_id=ppt.profession_id
   left join sapphire_.provider_practice pp
   on pp.provider_id=ppt.provider_id
   where profession_degree='DO' and city like '%er%'
```

4. List all providers with their profession code created after April of 2010 and is associated with a city that has more than one word in the name.

```
a. select * from sapphire_.provider_practice pp
   left join (select provider_id, created::date
       from sapphire_.provider_table) dt
   on dt.provider_id=pp.provider_id
   where (created > '2010-04-01') and ((city) like '% %')
```

5. Roger has a list of doctors on a csv file. The file contains a column called “provider_id”. He would like to compare that file against the provider table and see what provider_id does not exist. How would you bring the file in the database and what query would you use on the comparison?
    a. **How to bring it into the database:** parse with Python and then load into the database using pyodbc, psycopg2, sqlite3, or any other database driver tool within Python. Below are two options to load. The first uses AWS’s boto3 to move csvs into S3 and the second uses Google Cloud functions to delivery into a Cloud Storage.

```
# OPTION 1 - using the aws boto3 library to load data into a S3 bucket
import boto3
from botocore.exceptions import NoCredentialsError
import os

AWSAccessKeyId=''
AWSSecretKey=''


def upload_to_aws(local_file, bucket, s3_file=None):
    '''
    local_file = name of local file on your origin drive
    bucket = name of s3 bucket
    s3_file = what to name it in the s3 bucket
    '''

    s3 = boto3.client('s3', aws_access_key_id = AWSAccessKeyId,
aws_secret_access_key=AWSSecretKey)
    try:
        s3.upload_file(local_file, bucket, s3_file)
        print('Upload Sucessful')
    except FileNotFoundError:
        print('File was not found.')
        return False
    except NoCredentialsError:
        print('Credentials not available')
        return False

for file in os.listdir('C:\\Users\eugen\PycharmProjects\pythonProject\datatoload'):
    s3_file_name = file.replace('rt_hrl_lmps_2021_from', '')

upload_to_aws(f'C:\\Users\eugen\PycharmProjects\pythonProject\datatoload\\{file}',
'plotlydashtest', s3_file=f'2021_da/{s3_file_name}')

--------------------------------------------------------------------------------
# OPTION 2 - using google cloud’s cloud functions
def day_ahead():
  import pandas as pd
  from random import randint
  import time
  import json, csv
  import os
  from datetime import date
  import requests
  from google.cloud import storage
  from google.cloud import bigquery

  day = int(date.today().day) - 1

  month = date.today().month
  year = date.today().year

  print('Starting...')
```

```
  print('Here it is : ', time.ctime())

  headers = {
    "Ocp-Apim-Subscription-Key":
  }

  results = []

  count = 50000
  for start in range(1, 350000, count):
    print('Sleepy Time')

    url =
f'https://api.pjm.com/api/v1/da_hrl_lmps?&rowCount={count}&startRow={start}&datetim
e_beginning_ept={month}-{day}-{year} 00:00 to {month}-{day}-{year}
23:59&format=json&subscription-key=<b9a2919b311b41d8a02454eb11f3195f>'
    resp = requests.get(url, headers=headers)

    print(resp.status_code)

    if resp.status_code != 200:
      with open('errors.txt', 'a') as the_file:
        the_file.write(f'Start {count} && Count {start}')
        the_file.write('\n')

    print(f'Working on {start}', time.ctime())
    result = json.loads(resp.text)['items']
    # result = json.load(resp.text())['items']
    print(f'Now writing {start} from {count}', time.ctime())
    result_normed = pd.json_normalize(result)
    # .to_csv(f'da_hrl_lmps_2021_{month}{day}_round_{start}.csv', index=False)

    os.environ["GOOGLE_APPLICATION_CREDENTIALS"] = r'auth_key.json'

    # storage_client = storage.Client.from_service_account_json(
    #     'https://storage.cloud.google.com/keytexttestbucket/client-project-
322017-16db11d5b285.json')

    storage_client = storage.Client()
    bucket = storage_client.bucket('keytex_bucket')


bucket.blob(f'2021_da/2021_csvs/da_hrl_lmps_2021_daily_{month}{day}_round_{start}.c
sv').upload_from_string(
      result_normed.to_csv(index=False), 'text/csv')


day_ahead()
```

i.

b. **How to compare:** Once the data is loaded I would do something simple like this:

i.
```
select provider_id from sapphire_.provider_id
where provider_id not in (select distinct provider_id from
new_table_from_csv) ntfc
```

ii. Otherwise, I could make a sql query to pull the ids in the table, parse them in as a DataFrame with pandas and compare the csv using a simple ~.isin() for those not found to be the result.

6. Sally is looking for a doctor who lives near her town in the Rutherford area and knows she a couple of options. Write a query that will list the doctors in her town using a single where clause.

```
a. select pp.provider_id, provider_name, city
   from sapphire_.provider_table pt
   right join sapphire_.provider_practice pp
   on pt.provider_id=pp.provider_id
   where pp.city like '%Rutherford%'
```

7. John has come to you and noticed some of the query performance he is experiencing is extremely slow. How would you approach this situation to figure out the performance issue?
    a. Review query diagram
    b. Remove/rewrite any bottlenecks
    c. If on cloud architecture scale machines vertically or horizontally depending on your setup
    d. Query based on an index rather than flatout
    e. Sort where possible and try to cut the size of data being queried in halves where possible

8. Write an index using the query you compiled in question 5.
    a. Adding this the name table as it seems relevant to the work this person would be doing based on the other questions in the take-home.

```
b. CREATE INDEX provider_id_index
   ON sapphire_.provider_table (provider_id, provider_name)
```

9. You have decided to delete doctors that contained the name 'John' in their name. You have a function in the lib schema called item_remove where the first item in the function is the table and the second item is the provider_id. Write a shell script that will connect to the database and loop through the provider_id into the function.
    a. Shell scripting is not my forte. I would do this all in Python or sql directly.
    b. https://stackoverflow.com/questions/2836829/shell-script-that-iterates-through-a-sql-delete-statement

10. Create a shell script that contains a variable called "pg_con" which will connect to postgresql and write a sql statement using the sql from question 3 against the variable that output just the value and nothing else.
    a. See my answer to #9. Below is an example of something I've done within a Python script to update a database after scraping a website. I wanted to move the scraped data into a local database on my pc.

```
11.  import requests
   from bs4 import BeautifulSoup
   import time
   from random import randint
   import mysql.connector
   import re

   # -- for the database setup

   HOST = "database ip address"
   USERNAME = "my username"
   PASSWORD = "the password used"
   DATABASE = "the database used"
   cnx = mysql.connector.connect(user=USERNAME, password=PASSWORD,
                                 host=HOST,
```

```
                                database=DATABASE,
                                auth_plugin='mysql_native_password')  # auth_plugin
might give you errors depending on how the password is setup around the database
{google the error}
cursor = cnx.cursor()

sql = """INSERT INTO body_columns(use_case_deployment_scope, pros, cons, roi,
competitors_considered,
          support_rating_usability_recommendation, other_questions, others_used,
app_name) VALUES (%s,
          %s, %s, %s, %s, %s, %s, %s, %s)"""

        valso = [use_case_deployment_scope, pros, cons, roi,
competitors_considered,
                 support_rating_usability_recommendation, other_questions,
others_used, app_name]

        cursor.execute(sql, valso)
        cnx.commit()
```