24/12/21
ACA Data engineering program
Lusine Ghazaryan

The Aim of the project

The mission of the project is to create an OLTP process and to transform it into an OLAP process with the help of AWS tools, which is done in the example of an online booking system, that lets people book rooms in hotels, also some services /Cars, tours, and restaurants/. Below are listed some challenges and difficulties which I faced during the project and the solutions.

Challenges and Difficulties

1. Firstly, I had some problems when collecting and generating data.
In my project booking can be done by a guest or by a tour agent. So in every row, there must be or Guest_id or Tour_agent_id.
To solve this issue, I defined a function, which generated booking data according to my project aim Here is the code.

I added some Nan values to Guest_id values, which will help generate the table correctly

```
Guest_id = Guest['Guest_ID'].values
Na = ["Nan","Nan","Nan","Nan","Nan","Nan","Nan","Nan","Nan","Nan"]
Guest_id=np.concatenate((Guest_id, Na), axis=0)
List=[1,1,1,1,2]

import datetime
m=pd.date_range(start="2010-01-01",end="2022-03-01").to_pydatetime().tolist()


def gen():
    rand = random.choice(Guest_id)
    if rand == "Nan":
        data = [randint(10000000, 99999999), random.choice(Rooms['Room_id'].values), 0,
                random.choice(Tour_Agent['Tour_Agent'].values), random.choice(m), random.choice(List)]
        return data
    else:
        data = [randint(10000000, 99999999), random.choice(Rooms['Room_id'].values), rand, 0,
random.choice(m),
                random.choice(List)]
        return data
```

2. The Second problem which I came across during the project was the generation of the

booking_review table.
In this case, I took hotel reviews CSV file from the Internet, made some changes in the data frame in order to have reviews that have a max of 300 letters, and divided the dataset into good and bad reviews.
So in my generator function a random number between 1-10 is generated as a review score, and if the number is less than 5, then in the generator function randomly chooses review from the bad reviews data frame, and if the number is more than 5, then the otherwise. Here is the code

```
Room_review=pd.read_csv(r"hotel-reviews.csv", index=False)
Room_review=Room_review[Room_review['Description'].str.len() < 300]
Room_review_happy=Room_review[Room_review['Is_Response']=='happy']
Room_review_not_happy=Room_review[Room_review['Is_Response']=='not happy']

def gen_review():
    rand=randint(1,10)
    if rand <= 5:
        data=[("R"+
str(randint(10000000,99999999))),random.choice(Booking_status_conf['Booking_id'].values),
              rand,random.choice(Room_review_not_happy['Description'].values)]
        return data
    else:
        data=[("R"+
str(randint(10000000,99999999))),random.choice(Booking_status_conf['Booking_id'].values),rand,
              random.choice(Room_review_happy['Description'].values)]
        return data
```

3. I had also some troubles during transforming data from CSV files into PostgreSQL database tables. But I used the pscopgy2 copy CSV file command and it solved the issue.

Here is an example

```
cur = conn.cursor()

cur.execute(''' CREATE TABLE Restaurants(
 Restaurant_id VARCHAR(5) PRIMARY KEY,
 Name VARCHAR(50) NOT NULL,
 Cuisine_Style VARCHAR(150),
 Rating FLOAT NOT NULL
 )
 ''')

with open('Final_restaurant.csv', 'r') as f:
```

```
 next(f)
 cur.copy_from(f, 'restaurants', sep=',')
conn.commit()
```

4. There were some other challenges while creating lambda functions in AWS. It took me a while to understand how to create Lambda layers, how to make rules, give permissions, make scheduling of Lambda functions with the help of EventBridge, and so on.

I haven't finished the project yet, so I guess there are more challenges and difficulties that are waiting for me.