Data Engineering Project
Hotel Warehouse
Meri Sargsyan

Problems

- Overlapping check-in and check-out dates
- Importing necessary libraries
- Layers in Lambda function

Solutions

- In my database I have the table "Stay", where we cannot have the same room coinciding with the days of check-in and check-out.
  My solution was to insert unique rooms with proper check-in and check-out days. My check-in was the current day minus the days in the 1-1000 range, and the check-out was the check-in date plus days in the 1-30 range.

```python
start = list(range(1,1000))
end = list(range(1,31))

for i in range(500):
    rng = gen(pcg(seed = 359))
    stay_1["customer_id"].append(np.random.choice(customer.customer_id))
    stay_1["room_id"].append(np.random.choice(room.room_id))
    stay_1["check_in"].append( date.today() - timedelta(days = int(np.random.choice(start))) )
    stay_1["check_out"].append( stay_1["check_in"][i] + timedelta(days = int(np.random.choice(end))) )
    stay_1["num_of_people"].append(np.random.choice(list(range(1,6))) )

stay = pd.DataFrame(stay_1)
```

I have dropped the duplicates to make sure the rooms were unique.

```python
stay = stay.drop_duplicates(
    subset = ['room_id'],
    keep = 'last').reset_index(drop = True)
```

And while generating the new rows in the "Stay" table as the check-in date I took the check-out date of the existing room. The new check-out date was again the check-in date plus days in the 1-30 range. As I generate 10 new rows, to make sure the check-in dates will be correct I drop the duplicate room_ids if exists.

```
for k in range (10):
    stay['customer_id'].append(np.random.choice(customer['customer_id'].unique()))
    stay['room_id'].append(np.random.choice(room['room_id'].unique()))
    stay['check_in'].append(stays[stays['room_id'].isin(stay['room_id'])]['check_out'].max())
    stay['check_out'].append(stay['check_in'][k] + timedelta(days = int(np.random.choice(list(range(1,31))))))
    stay['num_of_people'].append(np.random.choice(list(range(1,6))))

stay = pd.DataFrame(stay)
stay = stay.drop_duplicates(
    subset = ['room_id'],
    keep = 'last').reset_index(drop = True)
```

- It was required to import the necessary libraries in order to run the generator tables with Lambda function. Unlike the numpy library, which exists in AWS layers, the pandas and psycopg2 libraries needed to be downloaded. For that I have
  1. Downloaded the whl_files,
  2. Unzipped the whl-files,
  3. Created a new folder structure,
  4. Moved the unzipped files to a new folder,
  5. Zipped the files,
  6. Created a new layer in AWS management console by uploading the zip-file.
- There was a problem regarding the layers when creating the Lambda functions. After some research, I finally found the right way to add the layers and create the functions. And finally, with the help of EventBridge, I have created the rule to run the lambda functions three times a day.