Data Warehouse

Analyse de Marché d'Hôtellerie à Marrakech

Encadré par Pr Imade Benelallam

AARABA ANASS

aaraba@insea.ac.ma

DSE

OMAR FAROUQ LAFDIL

o.lafdil@insea.ac.ma

DSE

Année académique 2021/2022

Table des matières :

Introduction :

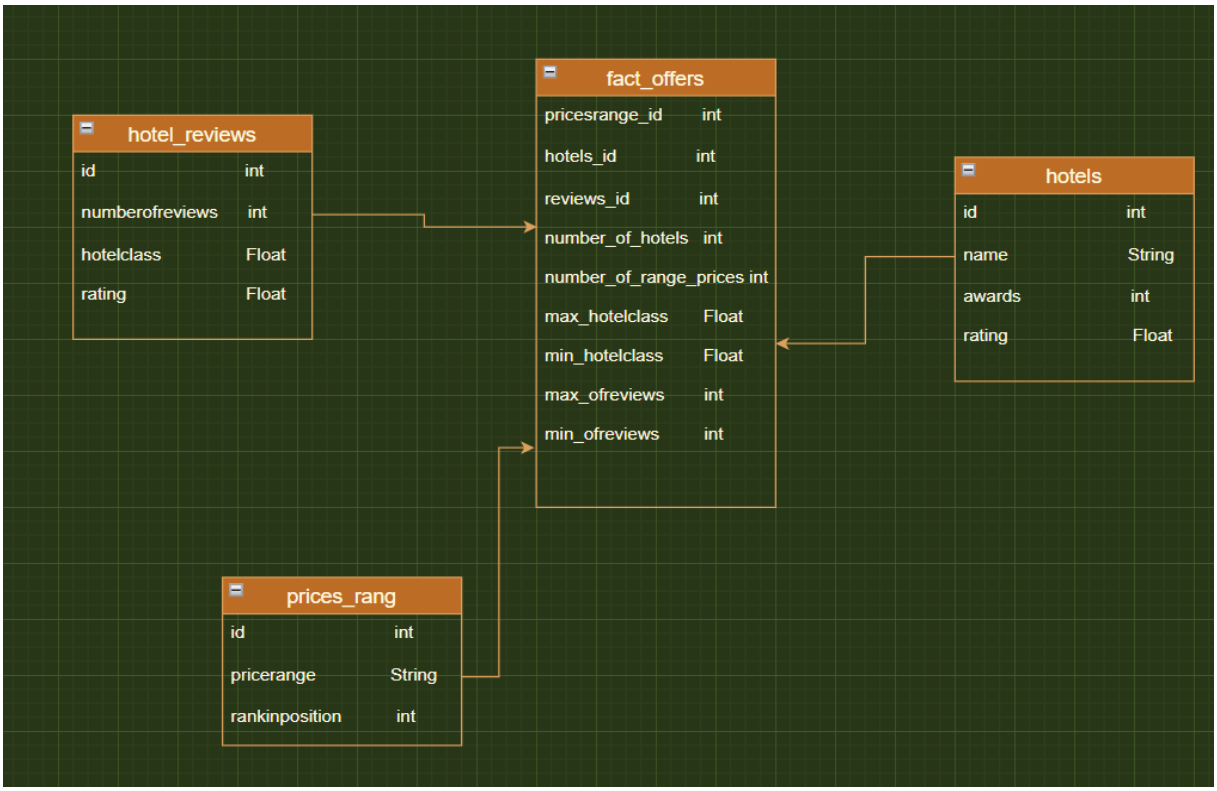
La ville de Marrakech est la capitale touristique du Maroc. L'hôtellerie dans Marrakech est le facteur d'étude de tourisme dans la ville. Après la réouverture des frontières connue pendant la pandémie de COVID-19, les responsables du secteur touristique, notamment le ministère du tourisme, pourrait demander l'acquisition un système décisionnel permettant de gérer les offres des hôtels à Marrakech.

Ce projet s'inscrit dans le cadre de l'élément de module Data Warehouse & Business Analytics. Il est réalisé en utilisant Airflow, MySQL Workbench, Mysql server, Apify tripadvisor scraper et Power BI.

I- Conception du datamart :

On veut produire un entrepôt de données pour étudier les offres des hôtels dans la ville de Marrakech à partir de trois tables de dimensions : hotel_reviews, hotels et prices_rang.

Le schéma en étoile pour l'étude des offres d'hôtels est :



Sous Mysql-python, son implémentation est donnée par :

```python
import mysql.connector as msql

conn = msql.connect(host='127.0.0.1', user='root',password='Anasaar123@', database="tp2_mysql")
cursor = conn.cursor()

#tables des dimensions
cursor.execute("CREATE TABLE IF NOT EXISTS prices_rang(id INT PRIMARY KEY NOT NULL AUTO_INCREMENT, pricerange VARCHAR "
               "(30), "
               "rankingposition INT)")
cursor.execute("CREATE TABLE IF NOT EXISTS hotels(id INT PRIMARY KEY NOT NULL AUTO_INCREMENT, name VARCHAR (100), "
               "awards INT, "
               "rating DOUBLE )")
cursor.execute("CREATE TABLE IF NOT EXISTS hotel_reviews(id INT PRIMARY KEY NOT NULL AUTO_INCREMENT, numberofreviews "
               "INT, "
               "hotelclass DOUBLE , "
               "rating DOUBLE )")
#Table de fait
cursor.execute("CREATE TABLE IF NOT EXISTS fact_offers(pricesrange_id INT, hotels_id INT, reviews_id INT, "
               "number_of_hotels INT, number_ofrange_prices  INT, max_hotelclass INT, min_hotelclass INT, "
               "max_ofreviews INT, min_ofreviews INT,KEY pricesrange_id (pricesrange_id), KEY hotels_id (hotels_id), "
               "KEY reviews_id (reviews_id), CONSTRAINT fact_offers_ibfk_1 FOREIGN KEY (pricesrange_id) REFERENCES "
               "prices_rang (id), CONSTRAINT fact_offers_ibfk_2 FOREIGN KEY (hotels_id) REFERENCES hotels (id), "
               "CONSTRAINT fact_offers_ibfk_3 FOREIGN KEY (reviews_id) REFERENCES hotel_reviews (id))")
```

Résultats dans MYSQL SERVER :

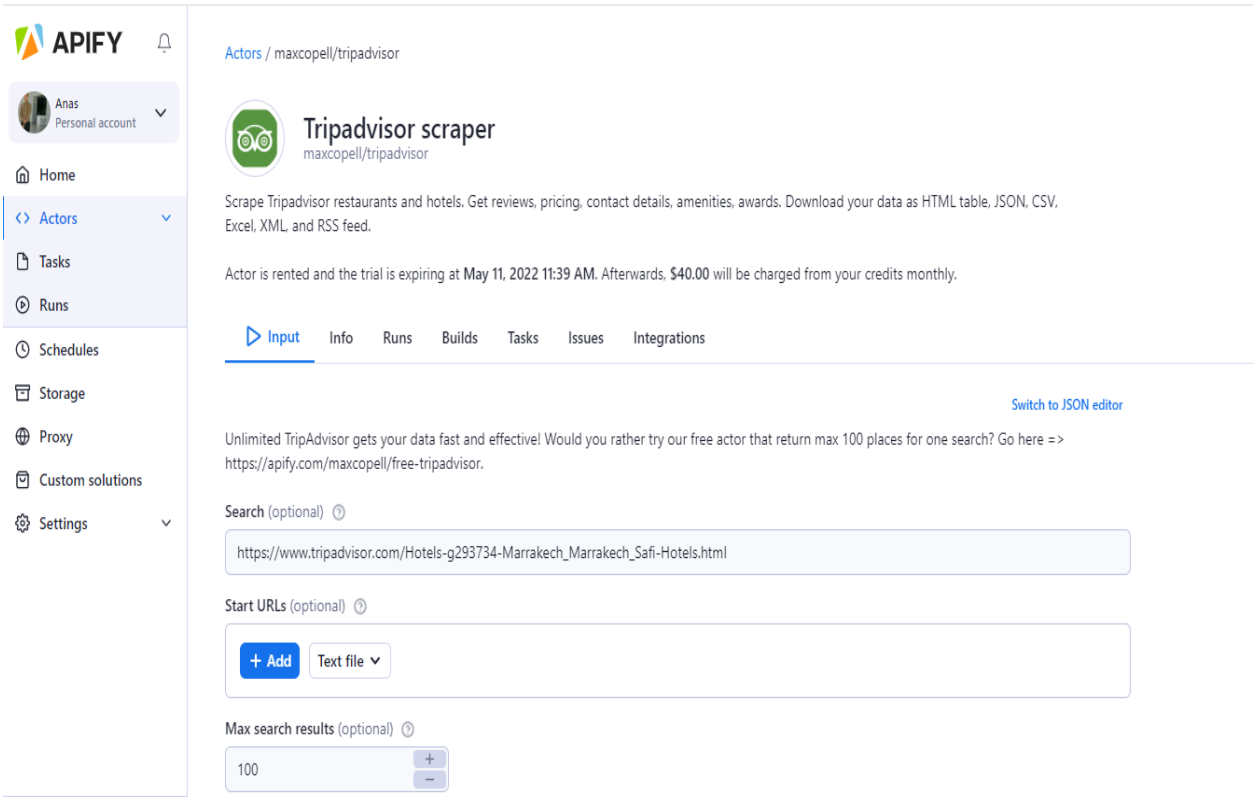
```
mysql> show tables;
+----------------------+
| Tables_in_tp2_mysql  |
+----------------------+
| fact_offers          |
| hotel_reviews        |
| hotels               |
| prices_rang          |
+----------------------+
4 rows in set (0.00 sec)
```

II- Apify TripAdvisor Scraper :

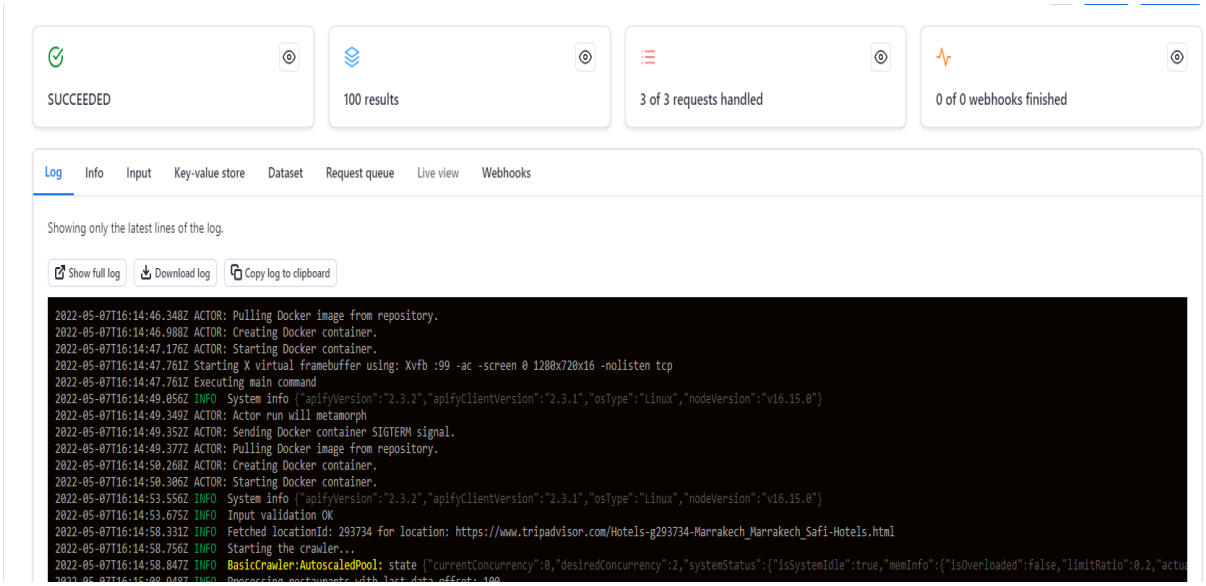
C’est un robot web hébergé qui permet à toute personne possédant des compétences élémentaires en programmation d'extraire des données structurées de n'importe quel site web. Contrairement aux scraping web par pointer-cliquer, Apify fonctionne sur des sites web modernes de plus en plus complexes et dynamiques. Il peut être utilisé par une large gamme d’utilisateurs, Quiconque a besoin de données provenant du web, qu'il s'agisse d'un étudiant, un journaliste, une start-up ou une grande entreprise.

Il s’agit d’explorer des sites Web, d’en extraire des données structurées et de les exporter dans des formats tels qu'Excel, CSV ou JSON.

Pour notre cas on utilise les hôtels dans la ville de Marrakech et on l’exporte sous JSON :



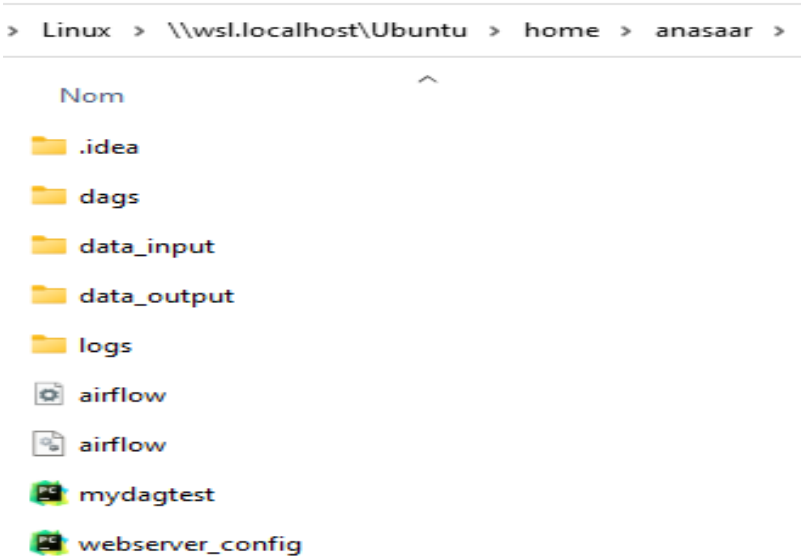
<u>Les résultats sont :</u>



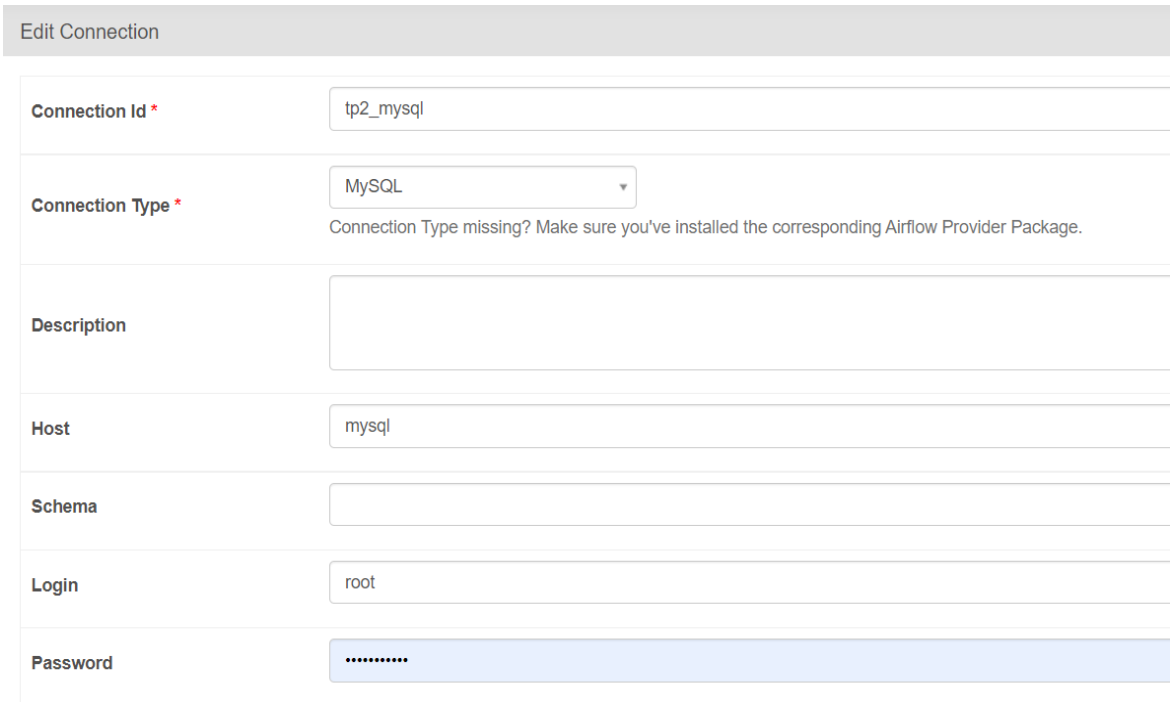
III- Préparation de l'environnement de Pycharm et Airflow sous WSL (Ubunto) :

1- Préparation de Airflow sous WSL (Ubunto) :

Organisation du dossier Airflow sous WSL :

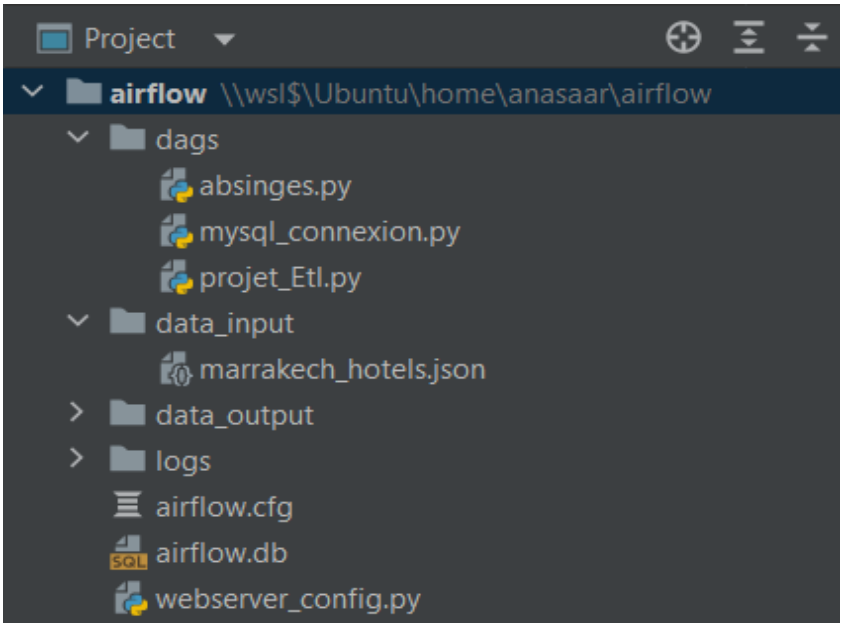


Connexion d'airflow avec Mysql server :



2- <u>Préparation de l'environnement du Pycharm :</u>

L'environnement Pycharm jouera un rôle fondamental dans ce projet. Il contiendra dans un dossier un code python qui traitera les différentes bases de données et le Dag utilisée pour nos objectifs. Sa structure est la suivante :



Les dossiers et fichiers peuvent se décrire comme suit :

- Projet_Etl.py : Noyau du projet contenant le dag utilisé.

- Mysql_connexion.py : Celui responsable de créer les tables de dimension ainsi que la table de fait.
- Marrakech_hotels.json : Le fichier contenant les données extrait de TripAdvisor.
- Data_output : Le dossier qui va contenir les fichiers CSV tirés du fichier JSON.

IV- Création et exécution du DAG TripAdvisor_Hotels_dag sous Airflow :

Airfow est une plateforme qui permet de créer, de planifier et de surveiller des workflows (flux de travail) par le biais de la programmation informatique. Un workflow peut se définir comme une suite de tâches mis en place selon un calendrier ou déclenché par un évènement. La mise en œuvre d'un workflow passe par la création d'un DAG.

1- Création du DAG TripAdvisor_Hotels_dag :

- Load_file :

Permet d'importer les données depuis le dossier input :

```
def load_file():
    with open(input, 'r') as f:
        data = json.load(f)
    return data
```

- Transformation du data de fichier JSON au fichiers CSV :

```python
def transform_apifydata(path: str):
    apifydata = load_file()
    transformed_data = []
    try:
        for DataRow in apifydata:
            transformed_data.append({
                'ID': DataRow['id'],
                'Name': DataRow['name'],
                'Type': DataRow['type'],
                'rating': DataRow['rating'],
                'Awards': len(DataRow['awards']),
                'RankingPosition': DataRow['rankingPosition'],
                'HotelClass': DataRow['hotelClass'],
                'NumberOfReviews': DataRow['numberOfReviews'],
                'priceRange': DataRow['priceRange']
            })
    except:
        pass
    marakech_df1 = pd.DataFrame(transformed_data)
    marakech_df1.to_csv(path, index=False)


def prices_rang_data(path: str):
    apifydata = load_file()
    transformed_data = []
    try:
        for DataRow in apifydata:
            transformed_data.append({
                'priceRange': DataRow['priceRange'],
```

```
        for DataRow in apifydata:
            transformed_data.append({
                'priceRange': DataRow['priceRange'],
                'RankingPosition': DataRow['rankingPosition']
            })
    except:
        pass
    prices_rang_df1 = pd.DataFrame(transformed_data)
    prices_rang_df1.to_csv(path, index=False)


def hotels_data(path: str):
    apifydata = load_file()
    transformed_data = []
    try:
        for DataRow in apifydata:
            transformed_data.append({
                'Name': DataRow['name'],
                'Awards': len(DataRow['awards']),
                'rating': DataRow['rating']
            })
    except:
        pass
    hotels_data_df1 = pd.DataFrame(transformed_data)
    hotels_data_df1.to_csv(path, index=False)


def hotel_reviews_data(path: str):
    apifydata = load_file()
    transformed_data = []
```

```
def hotel_reviews_data(path: str):
    apifydata = load_file()
    transformed_data = []
    try:
        for DataRow in apifydata:
            transformed_data.append({
                'NumberOfReviews': DataRow['numberOfReviews'],
                'HotelClass': DataRow['hotelClass'],
                'rating': DataRow['rating']
            })
    except:
        pass
    hotel_reviews_df1 = pd.DataFrame(transformed_data)
    hotel_reviews_df1.to_csv(path, index=False)
```

- Insertion des données dans la base donnée crée :

```python
def data_dimensions_insertion():
    csvdata = pd.read_csv('/home/anasaar/airflow/data_output/marrakech1.csv', index_col=False, delimiter=',')
    csvdata.head()
    for i, row in csvdata.iterrows():
        sql = "INSERT INTO prices_rang(pricerange, rankingposition) VALUES (%s,%s)"
        ms.cursor.execute(sql, tuple(row))
        ms.conn.commit()

    csvdata1 = pd.read_csv('/home/anasaar/airflow/data_output/marrakech2.csv', index_col=False, delimiter=',')
    csvdata1.head()
    for i, row in csvdata1.iterrows():
        sql = "INSERT INTO hotels( name, awards, rating) VALUES (%s,%s,%s)"
        ms.cursor.execute(sql, tuple(row))
        ms.conn.commit()

    csvdata2 = pd.read_csv('/home/anasaar/airflow/data_output/marrakech3.csv', index_col=False, delimiter=',')
    csvdata2.head()
    for i, row in csvdata2.iterrows():
        sql = "INSERT INTO hotel_reviews(numberofreviews, hotelclass, rating) VALUES (%s,%s,%s)"
        ms.cursor.execute(sql, tuple(row))
        ms.conn.commit()


def data_fact_insertion():
    sql="insert into fact_offers(number_of_hotels, number_ofrange_prices, max_hotelclass, min_hotelclass, " \
        "max_ofreviews, min_ofreviews) select count(name), count(pricerange), max(hotelclass), min(hotelclass), " \
        "max(numberofreviews), min(numberofreviews) from hotels, prices_rang, hotel_reviews; "
    ms.cursor.execute(sql)
    ms.conn.commit()
```

- Définition du DAG :

```python
with DAG(
        dag_id='TripAdvisor_Hotels_dag',
        start_date=pendulum.datetime(2022, 1, 1, tz="UTC"),
        schedule_interval='@Daily',
        catchup=False
) as dag:
```

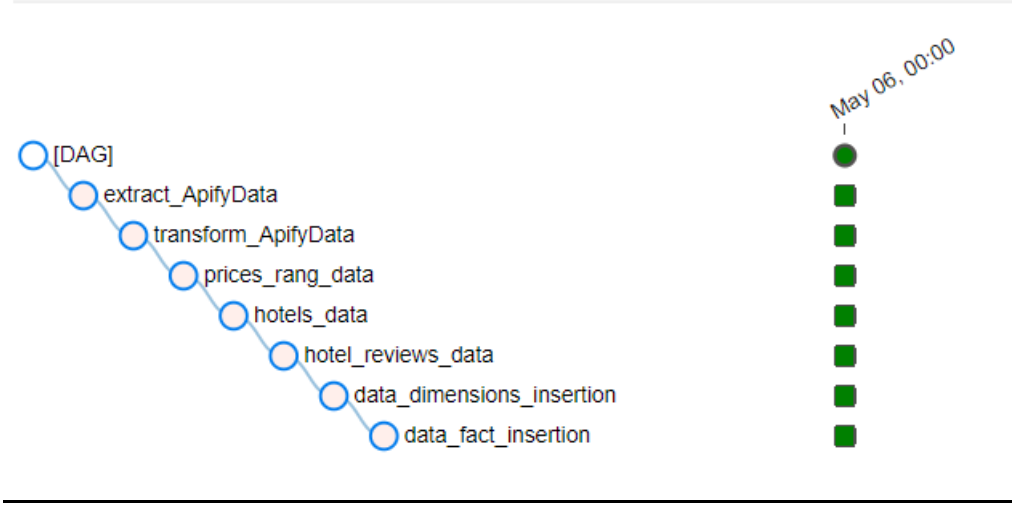
- Définition des taches utilisées :

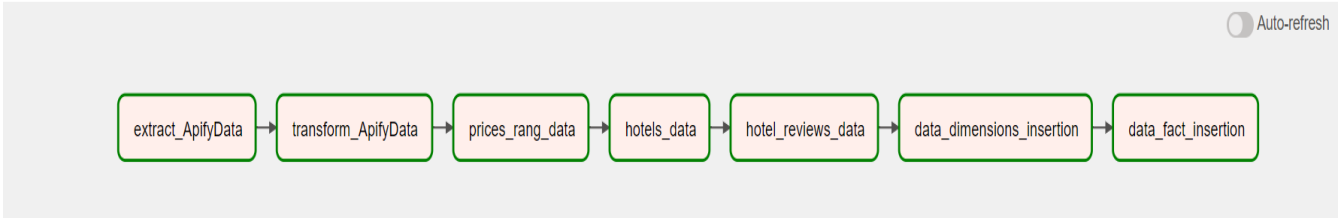
```
task_extract_ApifyData = PythonOperator(
    task_id='extract_ApifyData',
    python_callable=load_file,
    dag=dag
)
task_transform_ApifyData = PythonOperator(
    task_id='transform_ApifyData',
    python_callable=transform_apifydata,
    op_kwargs={'path': '/home/anasaar/airflow/data_output/marrakech.csv'},
    dag=dag
)
task_prices_rang_data = PythonOperator(
    task_id='prices_rang_data',
    python_callable=prices_rang_data,
    op_kwargs={'path': '/home/anasaar/airflow/data_output/marrakech1.csv'},
    dag=dag
)
task_hotels_data = PythonOperator(
    task_id='hotels_data',
    python_callable=hotels_data,
    op_kwargs={'path': '/home/anasaar/airflow/data_output/marrakech2.csv'},
    dag=dag
)
task_hotel_reviews_data = PythonOperator(
    task_id='hotel_reviews_data',
    python_callable=hotel_reviews_data,
    op_kwargs={'path': '/home/anasaar/airflow/data_output/marrakech3.csv'},
    dag=dag
)
```

```
task_insertion_dimension_tables = PythonOperator(
    task_id='data_dimensions_insertion',
    python_callable=data_dimensions_insertion,
    dag=dag
)
task_insertion_fact_table = PythonOperator(
    task_id='data_fact_insertion',
    python_callable=data_fact_insertion,
    dag=dag
)

task_extract_ApifyData >> task_transform_ApifyData >> task_prices_rang_data >> task_hotels_data >>\
 task_hotel_reviews_data >> task_insertion_dimension_tables >> task_insertion_fact_table
```

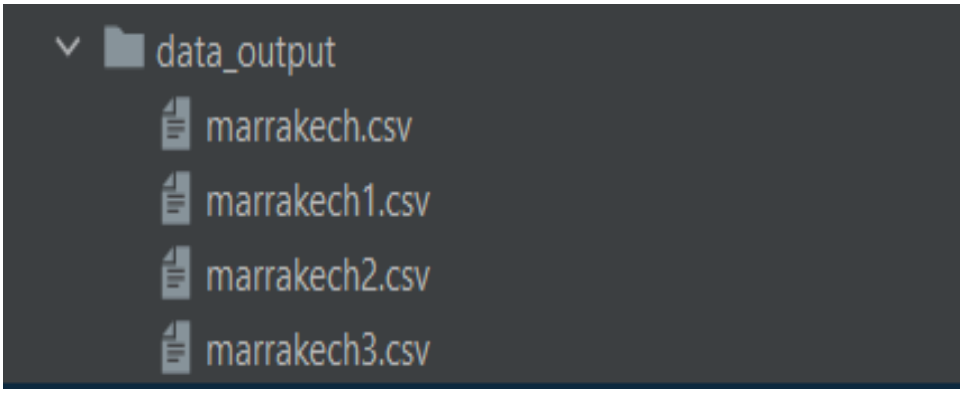
2- Exécution du DAG TripAdvisor_Hotels_dag :





- Visualisation des résultats du DAG :

Pour les fichiers CSV :



Pour la base données Mysql (Dans le Workbench) :

Table prices_rang :

	id	pricerange	rankingposition
▶	11	$82 - $155	52
	12	$126 - $201	51
	13	$99 - $124	56
	14	$354 - $817	59
	15	$126 - $327	57
*	NULL	NULL	NULL

Table hotel_reviews :

	id	numberofreviews	hotelclass	rating
▶	101	1241	4	5
	102	582	4	5
	103	272	3.5	5
	104	864	5	5
	105	531	3	5
	106	1133	5	4.5
	107	1023	3.5	5
	108	955	5	5
	109	310	5	5
	110	331	4	5
	111	914	4	5

Table hotels :

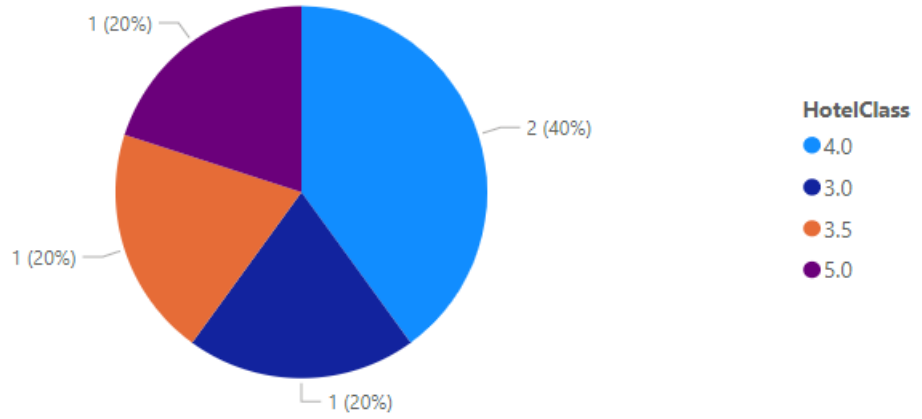
	id	name	awards	rating
▶	331	Riad Dar Najat	10	5
	332	Zamzam Riad	7	5
	333	Riad Tzarra	5	5
	334	Villa des Orangers	11	5
	335	Palais Khum	7	5
	336	Club Med Marrakech le Riad	10	4.5
	337	Riad Miski	10	5
	338	Villa Makassar	10	5
	339	Ksar Char-Bagh Small Luxury hotel	7	5
	340	Riad La Terrasse des Oliviers	8	5
	341	Riad 58 Blu	11	5

Table fact_offers :

	views_id	number_of_hotels	number_ofrange_prices	max_hotelclass	min_hotelclass	max_ofreviews	min_ofreviews
▶	L	27500	27500	5	0	2117	141

V- <u>Dashbording sous Power BI :</u>

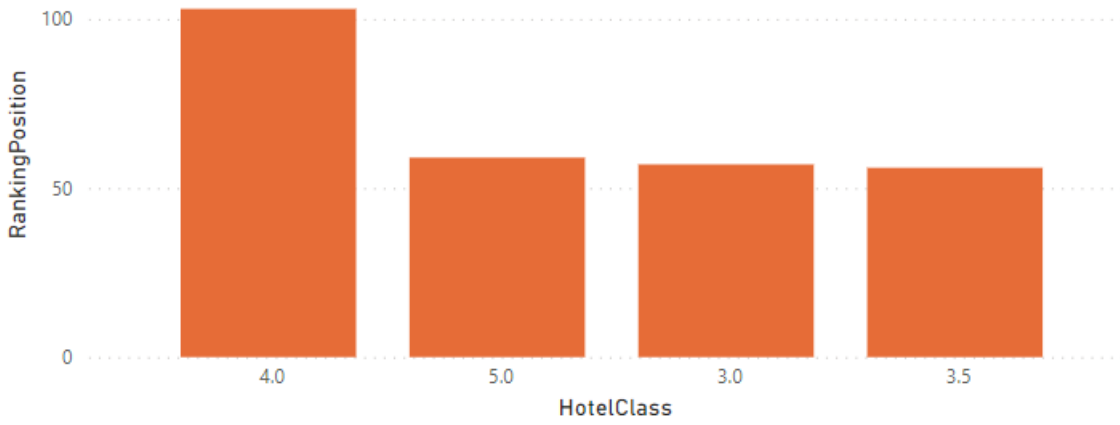
ID par HotelClass

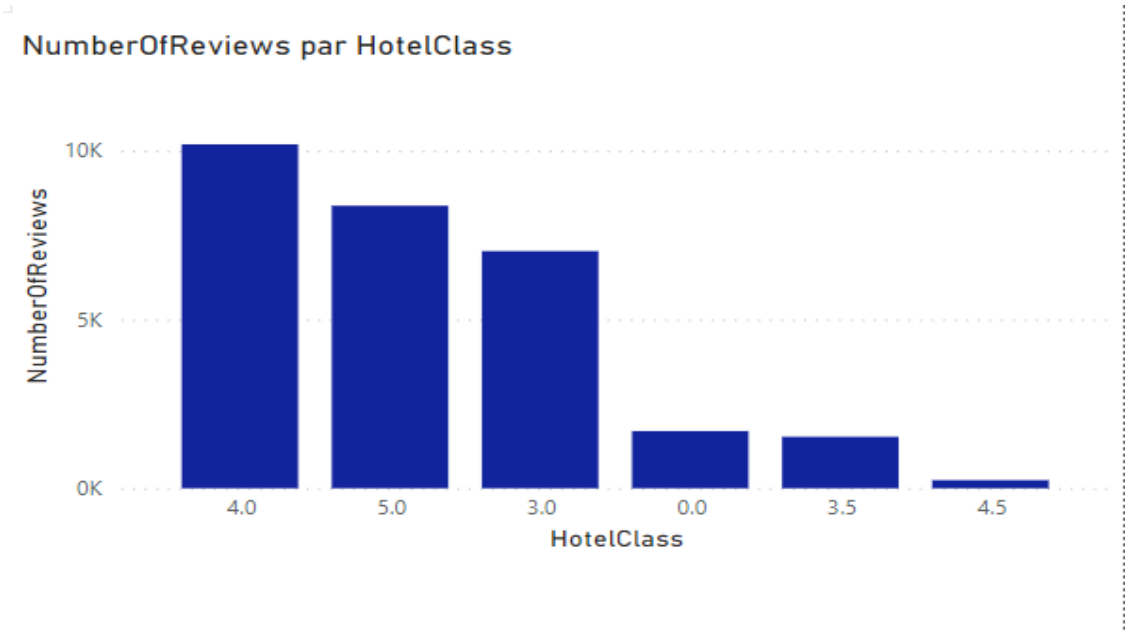


NumberOfReviews par HotelClass

NumberOfReviews par HotelClass

RankingPosition par HotelClass





Conclusion :

Ce travail nous a permis de bien saisir des notions pratiques relatives aux technologies tel que Airflow, Apify, MySQL, MySQL server et Power BI. Aussi ce travail est conçu pour permettre aux gens dans les postes de décision dans les hôtels à Marrakech et dans le secteur du tourisme d'essayer d'améliorer sa qualité et arriver à mieux gérer les hôtels et l'industrie d'hôtellerie au Maroc.