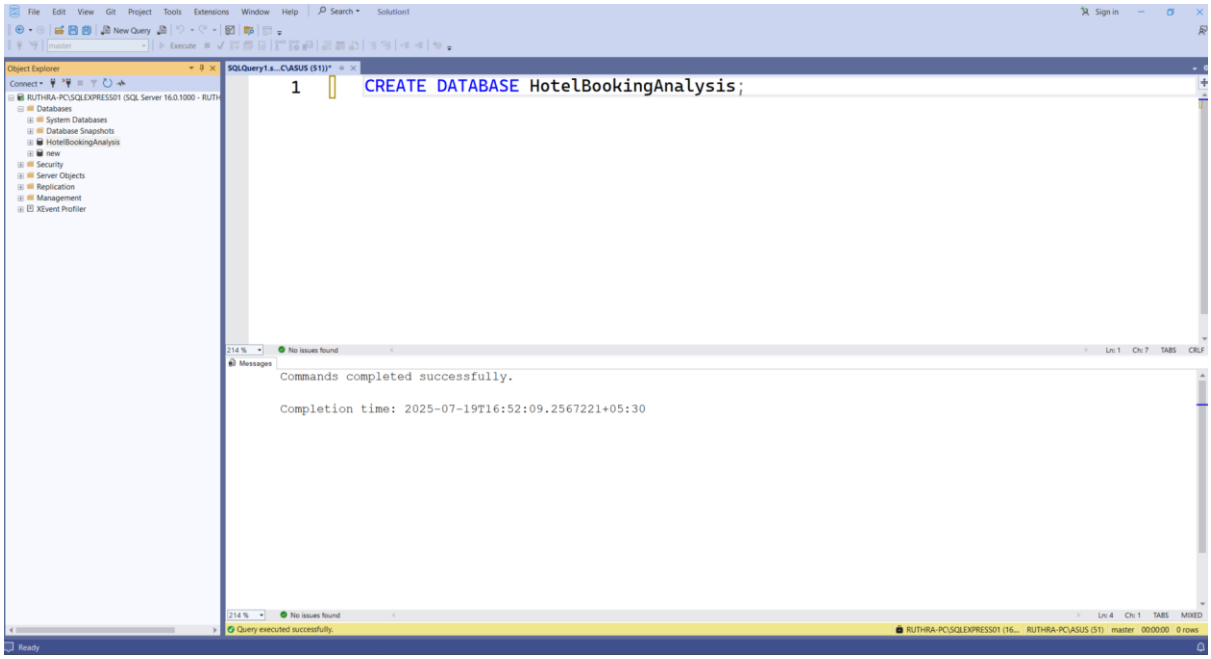
OYO Booking Analysis Case Study

Step 1: Database Setup

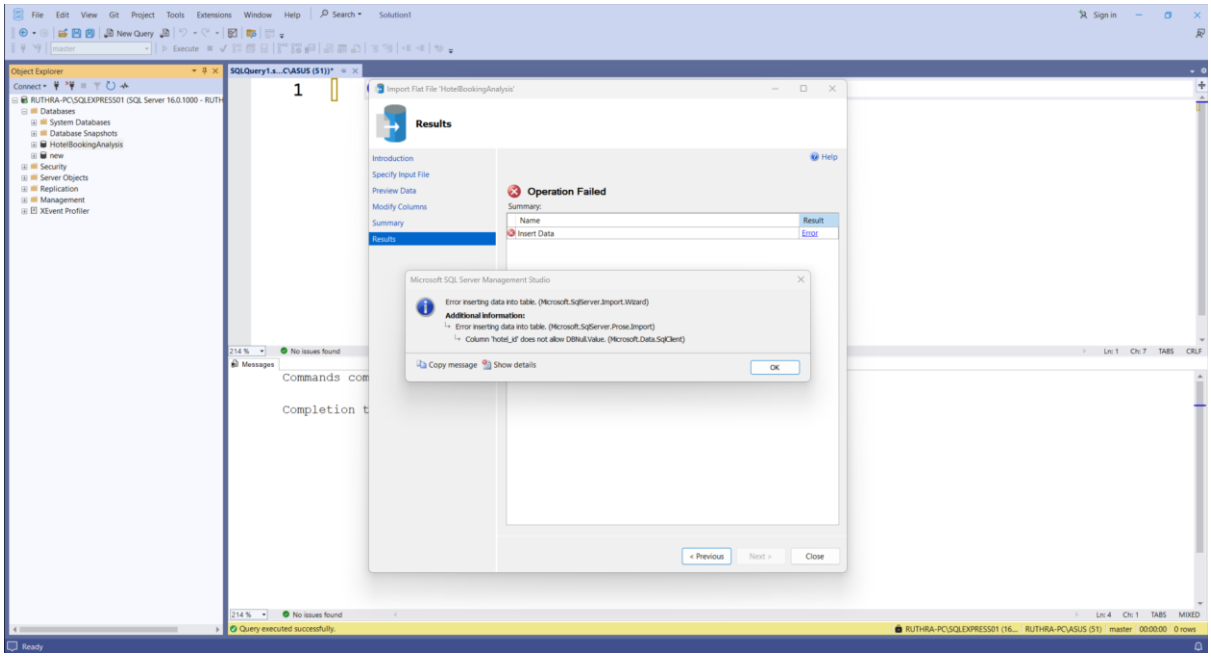
First, we'll create a database.



Step 2: Import Data

Since you have data in Excel, you can:

1. Save your Excel files as CSV
2. In SSMS, right-click your database → Tasks → Import Data
3. Follow the wizard to import your CSV files to the respective tables



There are some empty fields in the data, so the data was not imported properly. Therefore, we have to clean the data, and I chose to clean the data using pandas.`

Data Cleaning:

city_df

```
print(len(city_df))
```

```
9994
```

```
print(city_df.dtypes)
```

```
hotel_id    float64
city         object
dtype: object
```

```
city_df.isnull().sum()
```

```
hotel_id    9637
city        9637
dtype: int64
```

```
percent_null = (city_df["hotel_id"].isnull().sum() * 100) / len(city_df)
percent_null
```

```
96.42785671402842
```

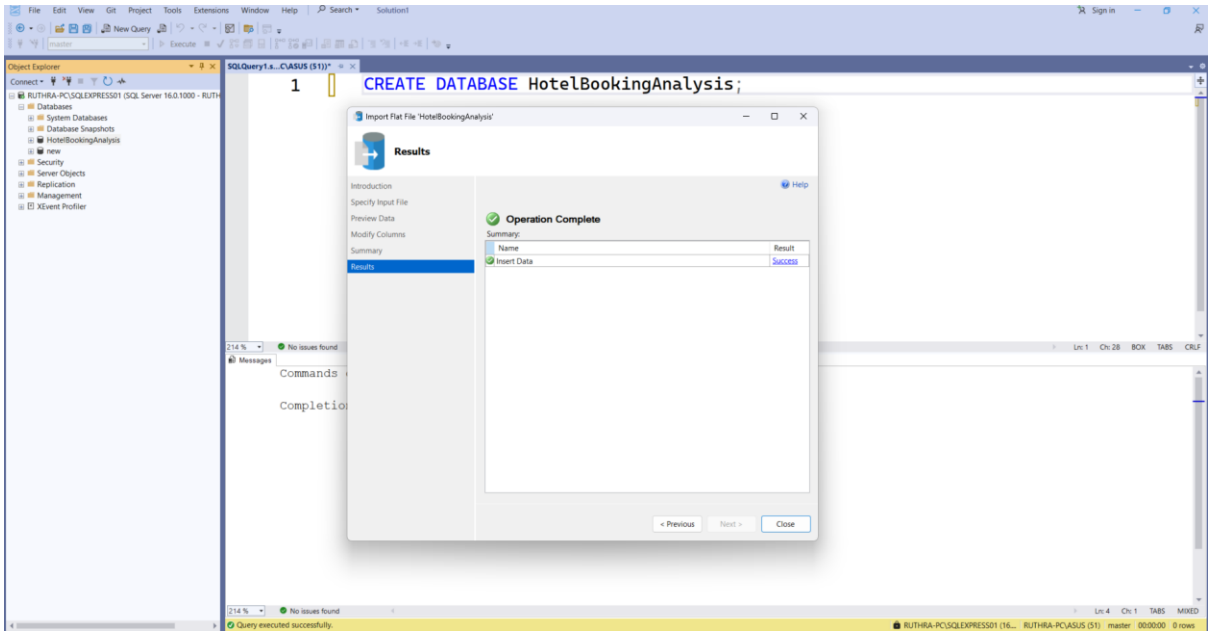
```
clean_city_df = city_df.dropna(subset=["hotel_id"])
```

```
clean_city_df.isna().sum()
```

```
hotel_id    0
city        0
dtype: int64
```

```
df.to_csv("oyo_city_cleaned.csv", index=False)
```

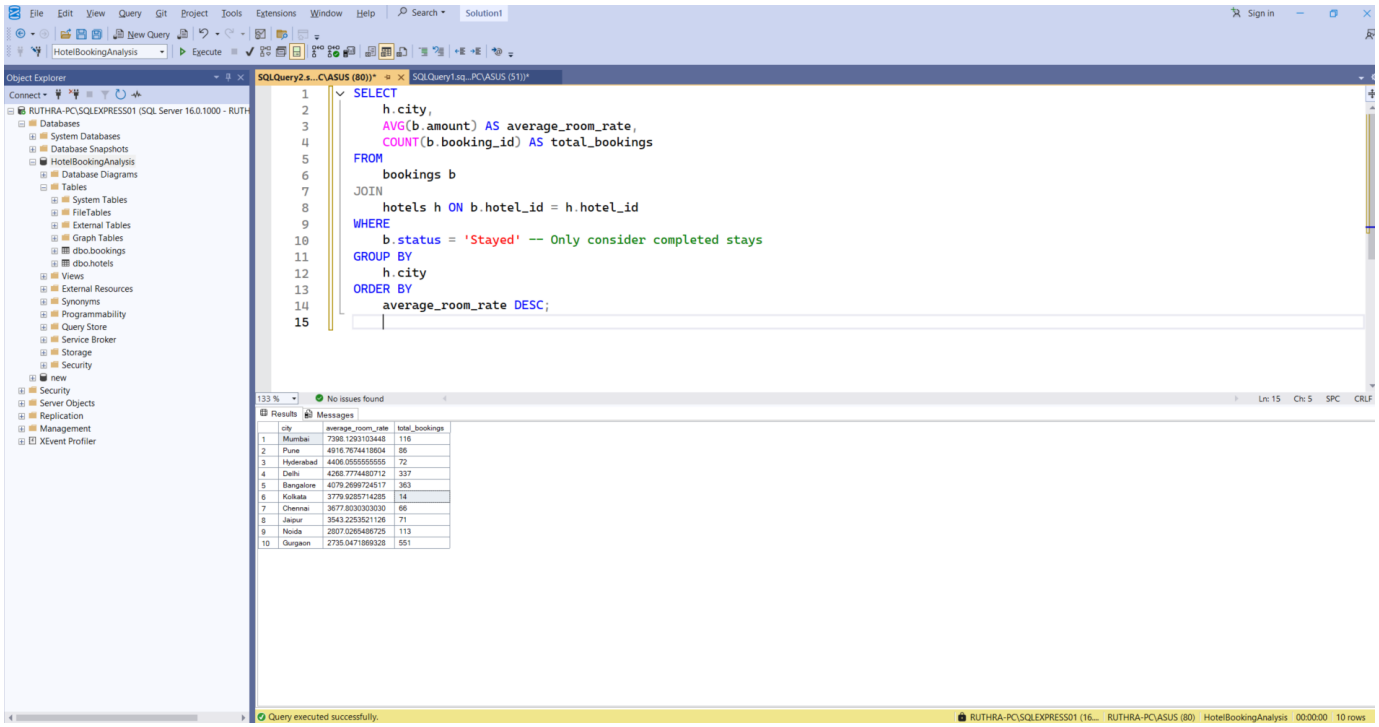
Data Insertion is successful after cleaning the Data.



Data Base is Populated **Successfully** by importing the data from the csv file.

Step 3: Analysis Queries

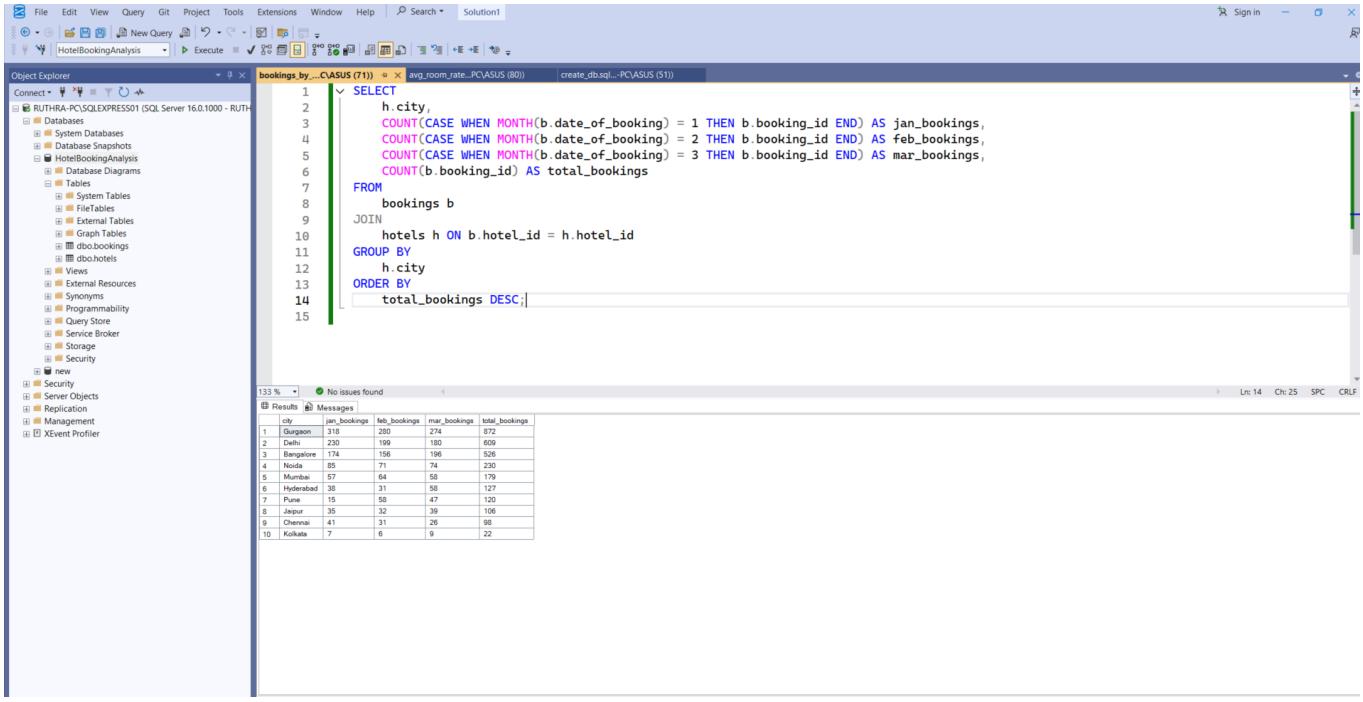
1. Average Room Rates by City



```
SELECT
    h.city,
    AVG(b.amount) AS average_room_rate,
    COUNT(b.booking_id) AS total_bookings
FROM
    bookings b
JOIN
    hotels h ON b.hotel_id = h.hotel_id
WHERE
    b.status = 'Stayed' -- Only consider completed stays
GROUP BY
    h.city
ORDER BY
    average_room_rate DESC;
```

	city	average_room_rate	total_bookings
1	Mumbai	7398.1293103448	116
2	Pune	4916.7674418604	86
3	Hyderabad	4406.0555555555	72
4	Delhi	4268.7774480712	337
5	Bangalore	4079.2699724517	363
6	Kolkata	3779.9285714285	14
7	Chennai	3677.8030303030	66
8	Jaipur	3543.2253521126	71
9	Noida	2807.0265486725	113
10	Gurgaon	2735.0471869328	551

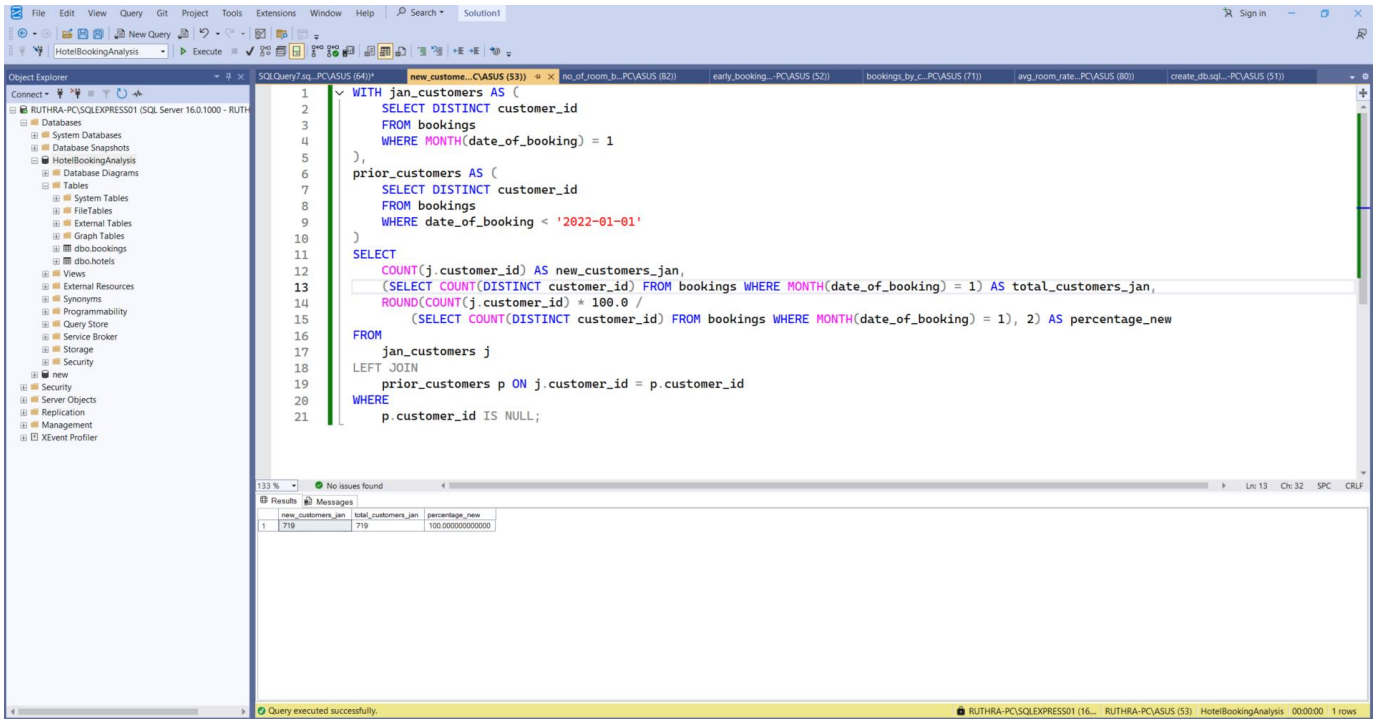
2. Bookings by City (Jan-Feb-Mar)



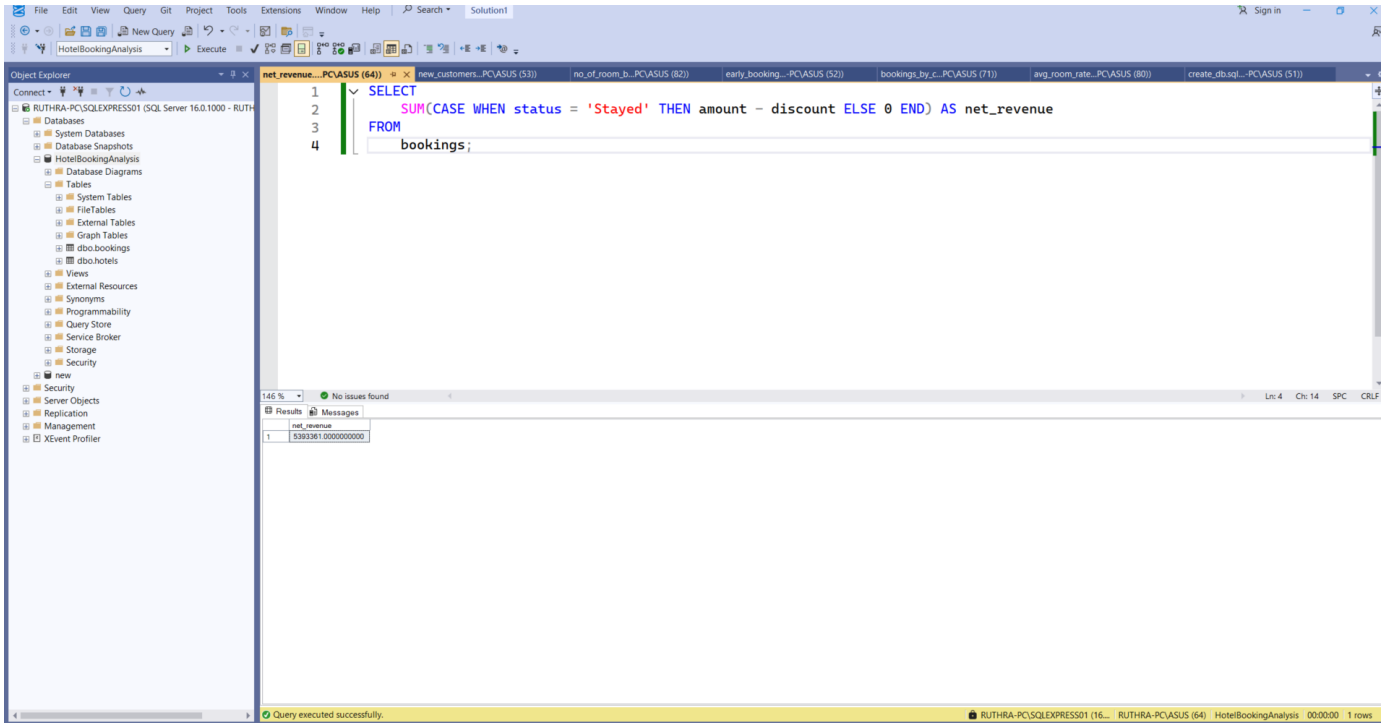
```
SELECT
    h.city,
    COUNT(CASE WHEN MONTH(b.date_of_booking) = 1 THEN b.booking_id END) AS jan_bookings,
    COUNT(CASE WHEN MONTH(b.date_of_booking) = 2 THEN b.booking_id END) AS feb_bookings,
    COUNT(CASE WHEN MONTH(b.date_of_booking) = 3 THEN b.booking_id END) AS mar_bookings,
    COUNT(b.booking_id) AS total_bookings
FROM
    bookings b
JOIN
    hotels h ON b.hotel_id = h.hotel_id
GROUP BY
    h.city
ORDER BY
    total_bookings DESC;
```

	city	jan_bookings	feb_bookings	mar_bookings	total_bookings
1	Gurgaon	318	280	274	872
2	Delhi	230	199	180	609
3	Bangalore	174	156	196	526
4	Noida	85	71	74	230
5	Mumbai	57	64	58	179
6	Hyderabad	38	31	58	127
7	Pune	15	58	47	120
8	Jaipur	35	32	39	106
9	Chennai	41	31	26	98
10	Kolkata	7	6	9	22

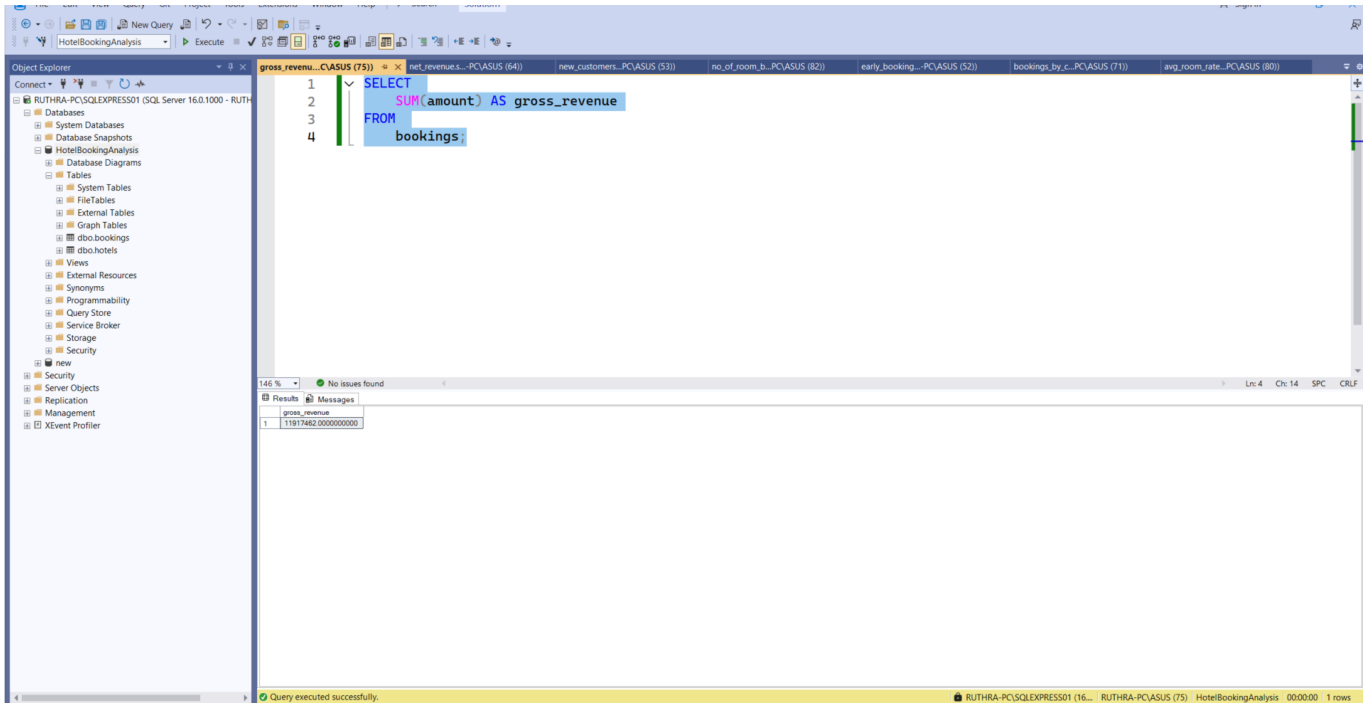
5. New Customers in January



6. Net Revenue (After Cancellations)



7. Gross Revenue (Total Booked Amount)



8. Cancellation Rate by City