**ATLIQ HOSPITALITY ANALYSIS**

EDA -

CREATE DATABASE IF NOT EXISTS ATLIQ\_HOSPITAL\_ANALYSIS;

USE ATLIQ\_HOSPITAL\_ANALYSIS;

CREATE OR REPLACE TABLE dim\_date(

`date` DATE PRIMARY KEY,

mmm\_yy VARCHAR(10),

week\_no INT,

day\_type VARCHAR(8)

)

CREATE OR REPLACE TABLE dim\_hotels(

property\_id INT PRIMARY KEY,

property\_name VARCHAR(20),

category VARCHAR(10),

city VARCHAR(10)

)

CREATE OR REPLACE TABLE dim\_rooms(

room\_id VARCHAR(5) PRIMARY KEY,

room\_class VARCHAR(15)

)

CREATE OR REPLACE TABLE fact\_aggregated\_bookings(

property\_id INT,

check\_in\_date DATE,

room\_category VARCHAR(5),

successful\_bookings INT,

capacity INT,

FOREIGN KEY (property\_id) REFERENCES dim\_hotels(property\_id),

FOREIGN KEY (check\_in\_date) REFERENCES dim\_date(`date`),

FOREIGN KEY (room\_category) REFERENCES dim\_rooms(room\_id)

)

CREATE OR REPLACE TABLE fact\_bookings(

booking\_id VARCHAR(20),

property\_id INT,

booking\_date DATE,

check\_in\_date DATE,

check\_out\_date DATE,

no\_guests INT,

room\_category VARCHAR(5),

booking\_platform VARCHAR(15),

ratings\_given INT,

booking\_status VARCHAR(15),

revenue\_generated INT,

revenue\_realized INT,

FOREIGN KEY (property\_id) REFERENCES dim\_hotels(property\_id),

FOREIGN KEY (check\_in\_date) REFERENCES dim\_date(`date`),

FOREIGN KEY (room\_category) REFERENCES dim\_rooms(room\_id)

)

SELECT \* FROM fact\_bookings;

--Sum of revenue\_realized

SELECT SUM(revenue\_realized) AS REVENUE FROM fact\_bookings;

--Count of booking\_id in fact\_bookings

SELECT COUNT(booking\_id) AS TOTAL\_BOOKINGS FROM fact\_bookings;

--Average of ratings\_given

SELECT ROUND(AVG(ratings\_given),2) AS AVG\_RATING FROM fact\_bookings WHERE ratings\_given <> 0;

--Sum of capacity

SELECT SUM(capacity) AS TOTAL\_CAPACITY FROM fact\_aggregated\_bookings;

--Sum of successful\_bookings from fact\_bookings

SELECT SUM(successful\_bookings) AS TOTAL\_SUCCESSFUL\_BOOKINGS FROM fact\_aggregated\_bookings;

--Ratio of Total Successful Bookings to Total Capacity

SELECT

SUM(capacity) AS TOTAL\_CAPACITY,

SUM(successful\_bookings) AS TOTAL\_SUCCESSFUL\_BOOKINGS,

ROUND(TOTAL\_SUCCESSFUL\_BOOKINGS/TOTAL\_CAPACITY \* 100, 2) AS OCCUPANCY

FROM fact\_bookings

FULL JOIN fact\_aggregated\_bookings

ON fact\_bookings.property\_id = fact\_aggregated\_bookings.property\_id;

--Count of booking\_id in which booking\_status = "Cancelled"

SELECT COUNT(booking\_id) AS TOTAL\_CANCELLED\_BOOKINGS FROM fact\_bookings WHERE booking\_status = 'Cancelled';

--Ratio of 'Total Cancelled Bookings' to 'Total Bookings'

SELECT

COUNT(booking\_id) AS TOTAL\_BOOKINGS,

(SELECT COUNT(booking\_id) FROM fact\_bookings WHERE booking\_status = 'Cancelled') AS TOTAL\_CANCELLED\_BOOKINGS,

ROUND(TOTAL\_CANCELLED\_BOOKINGS/TOTAL\_BOOKINGS \* 100, 2) AS Cancellation\_Rate

FROM fact\_bookings;

--Total number of bookings by room\_id wise

SELECT dr.room\_id, dr.room\_class, COUNT(fb.booking\_id) AS TOTAL\_BOOKINGS

FROM fact\_bookings fb

FULL JOIN dim\_rooms dr

ON fb.room\_category = dr.room\_id

GROUP BY 1, 2;

--Total number of bookings in each property

SELECT dh.property\_name, COUNT(fb.booking\_id) AS TOTAL\_BOOKINGS

FROM dim\_hotels dh

LEFT JOIN fact\_bookings fb

ON dh.property\_id = fb.property\_id

GROUP BY 1

ORDER BY 2 DESC;