**09 Exercise: SQL Queries**

**Simple queries:**

**7. List full details of all hotels**

SELECT \* FROM hotel;

**8. List full detalis of all hotels in Lomdon**

SELECT \* FROM hotel WHERE city = 'New York';

**9. List the names and adress of all guest living in London, alphabetically ordered by name.**

SELECT \* FROM booking INNER JOIN guest ON booking.fk\_guest = guest.guest\_id INNER JOIN hotel ON booking.fk\_hotel = hotel.hotel\_id WHERE city = 'New York' ORDER BY guestName ASC

**10. List all double or family rooms with a price below €40.00 pr night, in ascending order of price**

SELECT \* FROM room WHERE type = 'Dobbeltseng' AND price <= 200 OR type = 'Enkeltseng' AND price <= 200;

**11. List the bookings for which no dateTo has been specified**

SELECT \* FROM booking WHERE dateTo = ' ';

**Aggregate functions:**

**12. How many hotels are there?**

SELECT count(hotel\_id) AS hotelCount FROM hotel

**13. What is the average price of a room?**

SELECT AVG(price) FROM room

**14. What is the total revenue per night from all double rooms?**

SELECT sum(price) AS total FROM room WHERE type = 'Dobbeltseng';

**15. How many different quests have made bookings for August?**

SELECT count(DISTINCT booking\_id) FROM booking WHERE dateFrom BETWEEN '2014-08-01' AND '2014-08-31';

Eller for at få alle år i tabellen

WHERE dateFrom LIKE ’20\_\_-08%’ OR dateTo LIKE ’20\_\_-08%’;

**Subqueries :**

**16. List the price and type of all rooms at the Grosvenor Hotel?**

SELECT type, price FROM room WHERE fk\_hotel\_id = 1 ;

**17. List all guests currently staying at the Grosvenor Hotel?**

**19 What is the total income from bookings for the Grosvenor Hotel today?**

**20 List the rooms that are currently unoccupied at the Grosvenor Hotel.**

**21 What is the lost income from unoccupied rooms at the Grosvenor Hotel?**

**Grouping**

**22 List the number of rooms in each hotel.**

SELECT type, price, hotelName, COUNT(fk\_hotel\_id) AS countRoom

FROM room

INNER JOIN hotel ON room.fk\_hotel\_id = hotel.hotel\_id

GROUP BY hotelName;

SELECT hotelNo, COUNT(roomNo) AS numberOfRooms FROM room GROUP BY hotelNo

**23 List the number of rooms in each hotel in London.**

SELECT type, price, hotelName, city, COUNT(fk\_hotel\_id) AS countRoom

FROM room

INNER JOIN hotel ON room.fk\_hotel\_id = hotel.hotel\_id

WHERE city = 'New York'

GROUP BY hotelName;

SELECT hotelNo, COUNT(roomNo) AS count FROM Room r, Hotel h

WHERE Room.hotelNo = Hotel.HotelNo AND city = ’London’

GROUP BY hotelNo;

**24 What is the average number of bookings for each hotel in August?**

SELECT AVG(booking\_id)

FROM booking

WHERE dateFrom

BETWEEN '2014-08-01' AND '2014-08-31';

**25 What is the most commonly booked room type for each hotel in London?**

SELECT type, COUNT(room\_id) AS occurence

FROM booking

INNER JOIN hotel ON booking.fk\_hotel = hotel.hotel\_id

INNER JOIN room ON booking.fk\_room = room.room\_id

WHERE city = 'New York'

GROUP BY room\_id

ORDER BY occurence;

**26 What is the lost income from unoccupied rooms at each hotel today?**

SELECT type, price, hotelName, city, SUM(price) AS totalLoss

FROM room

INNER JOIN hotel ON room.fk\_hotel\_id = hotel.hotel\_id

GROUP BY hotelName

**Populating tables**

**27 Insert a row into a table.**

INSERT INTO hotel (hotelName, city)

VALUES ('Roskilde Hotel', 'Roskilde');

**28 Update the price of all rooms by 5%.**

UPDATE room

SET price = ( 1.05 \* price)

WHERE type = type

**29 Delete a row in a table**

DELETE FROM hotel

WHERE hotelName = 'Roskilde Hotel' AND city = 'Roskilde';

**Exercise from class**

**Subqueries**

**Get a list of propertyNo, type, rooms rentet out by Marie Howe**

SELECT propertyNo, type, rooms

FROM PropertyForRent

WHERE staffNo = (SELECT staffNo FROM Staff WHERE fName = 'Marie' AND lName = 'Howe');

**INNER JOINS fælles mængden:**

Eks. 1

SELECT c.clientNo, fName, lName, propertyNo, comment

FROM client c

JOIN Viewing v ON c.clientNo = v.clientID

ORDER BY propertyNo

Eks. 2

SELECT fName, lName, position, b.branchNo, b.street, b.city, b.postcode, count(p.propertyNo) AS numProAdmin

FROM Staff s

JOIN Branch b ON s.branchNo = b.branchNo

JOIN PropertyForRent p ON s.staffNo = p.staffNo

GROUP BY fName;

**LEFT JOIN vil have alt ud også dem der ikke er kiggget på :**

SELECT c.clientNo, fName, count(v.ID)

FROM Client c

LEFT JOIN Viewing v ON c.clientNo = v.clientID

GROUP BY c.clientNo

**RIGHT JOIN :**

SELECT p.propertyNo, count(v.ID)

FROM PropertyForRent p

RIGHT JOIN Viewing v ON p.propertyNo = v.propertyNo

GROUP BY p.propertyNo

**UNION :**

**General**

**30 Show that a query using the HAVING clause has an equivalent formulation**

**without a HAVING clause.**