Tutorial 2

Basic SELECT Statement (revision)

```
SELECT A1, A2, ...., An          (Attributes)
FROM R1, R2, ...., Rn            (Relations/Tables)
WHERE condition                  (Selection)
```

```
SELECT Name
FROM STUDENT
WHERE STUDENTID LIKE "%2017%";
```

```
SELECT *
FROM STUDENT;
```

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q1. List full details of all hotels.

- Q2. List full details of all hotels in London.

Single Column Ordering (REVISION)

ORDER BY and an attribute sorts the values according to the attribute (alphabetically for characters or numerically for numbers)

DESC (for descending) sorts it in reverse order.

```
SELECT staffNo, fName, lName, salary
FROM Staff
ORDER BY salary DESC;
```

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q3. List the names and addresses of all guests in London, alphabetically ordered by name.

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q4. List all double or family rooms with a price below 40.00 per night, in ascending order of price.

SELECT Statement – Aggregates (revision)

- COUNT (how many) MIN (minimum) MAX (maximum) AVG (average) SUM (sum)

- Apart from COUNT(*), each function eliminates nulls first and operates only on remaining non-null values.

- Can use DISTINCT before column name to eliminate duplicates.

- DISTINCT has no effect with MIN/MAX, but may have with SUM/AVG.

NULL Search Condition (revision)

Test for or null explicitly using special keyword IS NULL:

(opposite IS NOT NULL)

```
SELECT clientNo, viewDate
FROM Viewing
WHERE propertyNo = "PG4" AND
        comment IS NULL;
```

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q5. List the bookings for which no dateTo has been specified.

- Q6. How many hotels are there?

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q7. What is the average price of all the rooms?

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

Q8. How many different guests have made bookings for August 2013 (inclusive)?

Simple Join (Revision)

Join two tables by including WHERE and setting columns equal (you can use AND to join more than one column). You can use more than two tables.

```
SELECT c.clientNo, fName, lName,
       propertyNo, comment
FROM Client c, Viewing v
WHERE c.clientNo = v.clientNo;
```

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q9. List the price and type of all rooms at the Grosvenor Hotel.

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q10. List all information for all guests currently staying at the Grosvenor Hotel. (CURRENT_DATE() function in MySQL)

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q11. What is the total income from bookings for people staying in the hotel with hotelNo of 015 today?

Revision: Nested subquery: use of IN

List properties handled by staff at '163 Main St'.

```
SELECT   propertyNo,   street,   city,   postcode,
type, rooms, rent
FROM PropertyForRent
WHERE staffNo IN
(SELECT staffNo
 FROM Staff
 WHERE branchNo =
         (SELECT branchNo
          FROM Branch
          WHERE street = '163 Main St'));
```

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q12. List information of the rooms that are currently unoccupied. (Hint: find rooms that are occupied currently first)

SQL JOIN(Revision)

NATURAL JOIN

JOIN USING

JOIN ON

LEFT JOIN – if left table has no match add NULL

RIGHT JOIN – if right table has no match add NULL

FULL JOIN – if either table has no match add NULL

```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q13. List the guestNo of guests who have booked executive rooms.
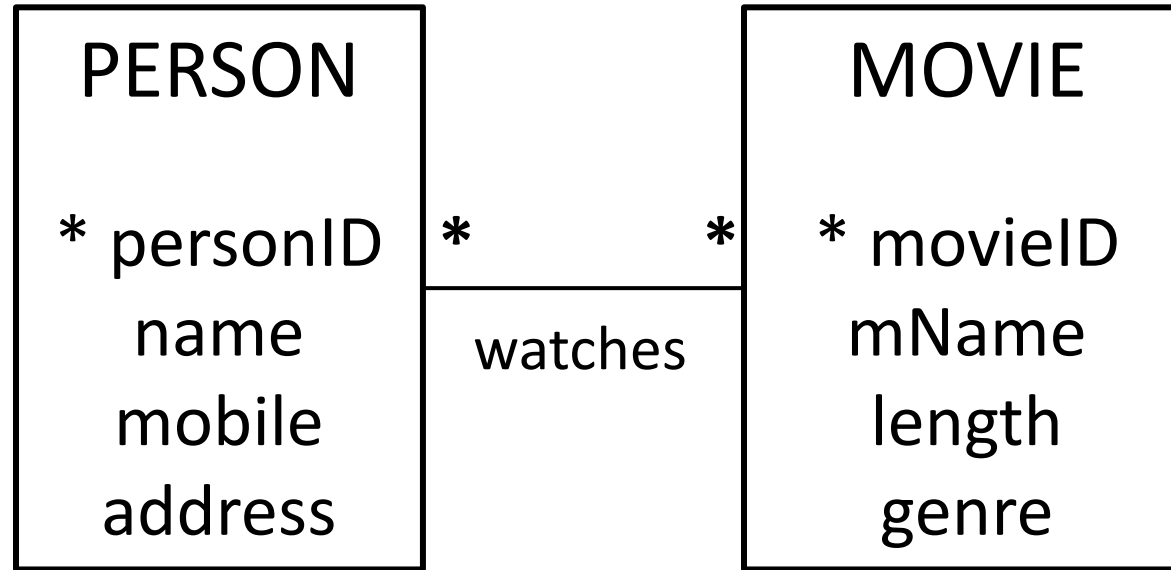
```
Hotel(hotelNo, name, address)
Room(roomId, hotelNo, type, price, number)
Booking(roomId, guestNo, dateFrom, dateTo)
Guest(guestNo, name, address)
```

- Q14. List all guests and their booking information, display NULL for the booking if the guest has not booked a room.

Q15. Derive relational schema for the entities and relationships in the ER model below.

PERSON
* personID
name
mobile
address

* ——— watches ——— *

MOVIE
* movieID
mName
length
genre

Q16. Derive relational schema for the entities and relationships in the ER model below.

SPACE

spaceID*
location

STAFF
*staffNo
name
telephone
licenceNo

(Mandatory, Or)

UNCOVERED

dailyFee

COVERED

floorNO
weeklyFee

BOOKING
*staffNo
*spaceNo
*startDate
period

1

*

1

*

23