Exercises – SQL 2

members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

facilities

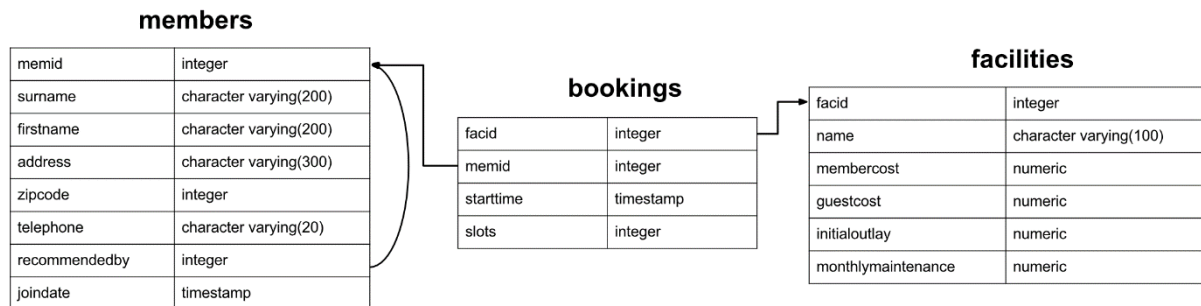| facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

1. Find surnames of all members who have made at least one booking.

```
SELECT DISTINCT mem.surname
FROM cd.members mem, cd.bookings book
WHERE mem.memid = book.memid;
```

surname
Worthington-Smyth
Joplette
Stibbons
Pinker
Dare
Rumney
Bader
Mackenzie
Sarwin
Boothe
Crumpet
Farrell
Tracy
Hunt
Owen
Tupperware
Smith
Butters
Rownam
Baker
GUEST
Genting
Jones
Purview
Coplin

2. Find surnames of all memebers, who have made more than 100 bookings.

```
SELECT mem.surname, count(*)
FROM cd.members mem, cd.bookings book
WHERE mem.memid = book.memid
GROUP BY mem.surname
HAVING COUNT(*) > 100;
```

surname	count
Joplette	159
Stibbons	104
Dare	117
Bader	120
Mackenzie	126
Boothe	188
Farrell	114
Tracy	176
Owen	131
Smith	560
Butters	164
Rownam	408
Baker	284
GUEST	883
Jones	129

3. **Find memid of all members, who have booked either "Squash Court" or "Pool Table".**

```
SELECT DISTINCT book.memid
FROM cd.facilities fac, cd.bookings book
WHERE fac.facid = book.facid and
      (fac.name = 'Squash Court' OR fac.name = 'Pool Table');
```

memid
14
27
8
12
17
28
1
15
10
26
11
4
30
0
16
33
6
29
2
21
3
35
20
5
13
22
9
7
24

4. **Find memid of all members, who have booked "Squash Court" or their surname begins with the letter 'R'.**

```
SELECT DISTINCT book.memid
FROM cd.facilities fac, cd.bookings book
WHERE fac.facid = book.facid AND (fac.name = 'Squash Court' )
UNION
SELECT mem.memid
FROM cd.members mem
WHERE surname LIKE 'R%' ;
```

memid
14
27
8
12
17
28
1
15
10
26
11
4
30
0
16
33
6
2
21
3
35
5
13
22
9
24

5. **Find memid of all members, who have booked "Squash Court" and "Pool Table".**

	memid
SELECT DISTINCT book.memid FROM cd.facilities fac, cd.bookings book WHERE fac.facid = book.facid AND (fac.name = 'Squash Court') INTERSECT SELECT DISTINCT book.memid FROM cd.facilities fac, cd.bookings book WHERE fac.facid = book.facid AND (fac.name = 'Pool Table');	14 27 8 12 17 28 1 15 10 26 11 4 30 0 16 33 6 2 21 5 13 22 9 24

6. **Upgrade the solution of the previous task (Task 5) by displaying the names and surnames of all members, who have booked "Squash Court" and "Pool Table".**
   HINT: Nested SELECT

	firstname	surname

SELECT mem.firstname, mem.surname
FROM cd.members mem
WHERE mem.memid IN (
    SELECT DISTINCT book.memid
    FROM cd.facilities fac, cd.bookings book
    WHERE fac.facid = book.facid AND (fac.name = 'Squash Court')
    INTERSECT
    SELECT DISTINCT book.memid
    FROM cd.facilities fac, cd.bookings book
    WHERE fac.facid = book.facid AND (fac.name = 'Pool Table')
);
	Jack
	Henrietta
	Tim
	Anne
	David
	David
	Darren
	Florence
	Charles
	Douglas
	David
	Janice
	Millicent
	GUEST
	Timothy
	Hyacinth
	Burton
	Tracy
	Anna
	Gerald
	Jemima
	Joan
	Ponder
	Ramnaresh
		Smith
		Rumney
		Boothe
		Baker
		Pinker
		Farrell
		Smith
		Bader
		Owen
		Jones
		Jones
		Joplette
		Purview
		GUEST
		Baker
		Tupperware
		Tracy
		Smith
		Mackenzie
		Butters
		Farrell
		Coplin
		Stibbons
		Sarwin

7. **Find surnames of all members, who have joined after the member whose surname is "Owen".**

```
SELECT DISTINCT mem.surname, mem.joindate
FROM  cd.members mem
WHERE  mem.joindate > ANY (SELECT mem.joindate
                          FROM cd.members mem
                          WHERE mem.surname = 'Owen'
) ;
```

surname	joindate
Tupperware	2012-09-18 19:32:05
Purview	2012-09-18 19:04:01
Hunt	2012-09-19 11:32:45
Jones	2012-09-02 18:43:05
Baker	2012-08-10 14:23:22
Bader	2012-08-10 17:52:03
Mackenzie	2012-08-26 09:32:05
Crumpet	2012-09-22 08:36:38
Jones	2012-08-06 16:32:55
Farrell	2012-08-10 14:28:01
Sarwin	2012-09-01 08:44:42
Farrell	2012-09-15 08:22:05
Rumney	2012-09-05 08:42:35
Pinker	2012-08-16 11:32:47
Smith	2012-08-10 16:22:05
Baker	2012-08-15 10:34:25
Genting	2012-08-19 14:55:55
Coplin	2012-08-29 08:32:41
Worthington-Smyth	2012-09-17 12:27:15
Smith	2012-09-26 18:08:45

8. **Find names and surnames of all members, who have booked one of the tennis courts (»Tennis Court 1«, »Tennis Court 2« …).**

```
SELECT DISTINCT mem.firstname, mem.surname, fac.name
FROM cd.members mem, cd.bookings book, cd.facilities fac
WHERE name LIKE 'Tennis%' AND
      mem.memid = book.memid AND
      book.facid=fac.facid;
```

firstname	surname	name
Ramnaresh	Sarwin	Tennis Court 2
Charles	Owen	Tennis Court 1
John	Hunt	Tennis Court 1
Florence	Bader	Tennis Court 2
Jemima	Farrell	Tennis Court 1
Ponder	Stibbons	Tennis Court 2
Tim	Boothe	Tennis Court 1
Burton	Tracy	Tennis Court 2
David	Farrell	Tennis Court 1
Erica	Crumpet	Tennis Court 1
Tim	Rownam	Tennis Court 2
Douglas	Jones	Tennis Court 1
GUEST	GUEST	Tennis Court 2
Janice	Joplette	Tennis Court 2
Charles	Owen	Tennis Court 2
Ramnaresh	Sarwin	Tennis Court 1
Nancy	Dare	Tennis Court 2
Timothy	Baker	Tennis Court 2
Ponder	Stibbons	Tennis Court 1
Matthew	Genting	Tennis Court 1
Jemima	Farrell	Tennis Court 2
Florence	Bader	Tennis Court 1
Jack	Smith	Tennis Court 2
John	Hunt	Tennis Court 2
Gerald	Butters	Tennis Court 2

9. **List the facility numbers (facid) that have the highest monthly maintenance**

```
SELECT fac.facid, fac.monthlymaintenance
FROM cd.facilities fac
WHERE fac.monthlymaintenance = (
              SELECT MAX(fac1.monthlymaintenance)
              FROM cd.facilities fac1);
```

facid	monthlymaintenance
4	300
5	300

10. List the facility numbers (facid) that have the second highest monthly maintenance

```
SELECT fac.facid, fac.monthlymaintenance
FROM cd.facilities fac
WHERE fac.monthlymaintenance =
(SELECT MAX(fac1.monthlymaintenance)
 FROM cd.facilities fac1
 WHERE fac1.monthlymaintenance <> (SELECT
                                MAX(fac2.monthlymaintenance)
                                FROM cd.facilities fac2));
```

facid	monthlymaintenanc
0	20
1	20

11. List the names of all members who have reserved 6 slots.

```
SELECT DISTINCT mem.firstname
FROM cd.members mem
WHERE EXISTS (
            SELECT book.memid
            FROM cd.bookings book
            WHERE slots = 6 AND mem.memid = book.memid
);
```

firstname
Henry
Ramnaresh
Charles
David
Timothy
Anne
Darren
Gerald
Tracy
Anna
Ponder
Tim
GUEST
Jack

12. List the name and surname of all visitors who have booked Badminton court more times than any person named "Timothy".

```
SELECT mem.firstname, mem.surname, COUNT(*)
FROM cd.members mem, cd.bookings book, cd.facilities fac
WHERE mem.memid = book.memid AND book.facid = fac.facid AND
fac.name = 'Badminton Court'
GROUP BY mem.memid
HAVING COUNT(*) > ALL (
            SELECT COUNT(*)
            FROM cd.members mem,
                cd.bookings book,
                cd.facilities fac
            WHERE mem.memid = book.memid
                AND book.facid = fac.facid
                AND fac.name = 'Badminton Court'
                AND mem.firstname = 'Timothy'
);
```

firstname	surname	count
Jack	Smith	12
Tim	Boothe	12
Anne	Baker	10
Darren	Smith	132
Florence	Bader	9
David	Jones	8
GUEST	GUEST	39
Tracy	Smith	32
Anna	Mackenzie	30
Gerald	Butters	20
Ponder	Stibbons	16
Nancy	Dare	10

13. List the name and surname of all visitors who have booked Badminton court more times than any person named "Timothy").

Solve in alternative way (use "INNER JOIN" or do it in the following way "R1.id = R2.id")

SELECT mem.firstname, mem.surname, COUNT(*)
FROM cd.bookings book
INNER JOIN cd.members mem ON book.memid = mem.memid
INNER JOIN cd.facilities fac ON book.facid = fac.facid
WHERE fac.name = 'Badminton Court'
GROUP BY mem.memid
HAVING COUNT(*) > ALL (
       SELECT COUNT(*)
       FROM cd.bookings book1
         INNER JOIN cd.members mem1 ON book1.memid =
mem1.memid
           INNER JOIN cd.facilities fac1 ON book1.facid = fac1.facid
       WHERE fac1.name = 'Badminton Court' AND
           mem1.firstname = 'Timothy'
);
	firstname **surname** **count**
	Jack Smith 12
	Tim Boothe 12
	Anne Baker 10
	Darren Smith 132
	Florence Bader 9
	David Jones 8
	GUEST GUEST 39
	Tracy Smith 32
	Anna Mackenzie 30
	Gerald Butters 20
	Ponder Stibbons 16
	Nancy Dare 10

14. List the names of the facilities that charge the highest price to guests.

SELECT fac.name, fac.guestcost
FROM cd.facilities fac
WHERE fac.guestcost = (
       SELECT MAX(fac1.guestcost)
       FROM cd.facilities fac1
);
	name **guestcost**
	Massage Room 1 80
	Massage Room 2 80

15. List start times of all bookings made by 'David Farrell' (Use INNER JOIN)

SELECT book.starttime FROM cd.bookings book INNER JOIN cd.members mem ON mem.memid = book.memid WHERE mem.firstname = 'Darren' AND mem.surname = 'Farrell' ;	**starttime** 2012-07-03 11:00:00 2012-07-03 08:00:00 2012-07-03 19:00:00 2012-07-03 10:00:00 2012-07-03 15:00:00 2012-07-04 15:30:00 2012-07-05 09:30:00 2012-07-05 19:00:00 2012-07-05 14:30:00 2012-07-06 17:00:00 2012-07-06 11:00:00 2012-07-06 14:00:00 2012-07-07 09:00:00 2012-07-07 11:30:00 2012-07-07 16:00:00 2012-07-07 10:30:00 2012-07-07 14:30:00 2012-07-08 15:00:00 2012-07-08 17:30:00 2012-07-08 11:30:00 2012-07-08 19:30:00 2012-07-08 16:30:00 2012-07-09 19:00:00 2012-07-09 09:00:00 2012-07-09 15:30:00 2012-07-10 11:00:00 2012-07-10 18:00:00 2012-07-11 08:00:00 2012-07-12 11:30:00 2012-07-12 09:00:00 2012-07-12 18:30:00

16. List start times ("starttime") of all bookings for facilities with IDs 0 and 1, for a date '2012-09-21'? Return a list of start times (starttime and object names). The list should be ordered by "starttime".

```
SELECT book.starttime as START, fac.name as NAME
FROM cd.facilities fac
        INNER JOIN cd.bookings book ON fac.facid = book.facid
WHERE fac.facid in (0,1) AND
        book.starttime >= '2012-09-21' AND
        book.starttime < '2012-09-22'
ORDER BY book.starttime;
```

start	name
2012-09-21 08:00:00	Tennis Court 1
2012-09-21 08:00:00	Tennis Court 2
2012-09-21 09:30:00	Tennis Court 1
2012-09-21 10:00:00	Tennis Court 2
2012-09-21 11:30:00	Tennis Court 2
2012-09-21 12:00:00	Tennis Court 1
2012-09-21 13:30:00	Tennis Court 1
2012-09-21 14:00:00	Tennis Court 2
2012-09-21 15:30:00	Tennis Court 1
2012-09-21 16:00:00	Tennis Court 2
2012-09-21 17:00:00	Tennis Court 1
2012-09-21 18:00:00	Tennis Court 2

17. List all members who have recommended another member at any time? (recommendedby). Get rid of all duplicates, the list should be sorted by name and surname.

```
SELECT DISTINCT mem.firstname , mem.surname
FROM cd.members mem
    INNER JOIN cd.members mems ON mems.memid =
mem.recommendedby
ORDER BY surname, firstname;
```

firstname	surname
Florence	Bader
Anne	Baker
Timothy	Baker
Tim	Boothe
Gerald	Butters
Joan	Coplin
Erica	Crumpet
Nancy	Dare
Matthew	Genting
John	Hunt
David	Jones
Douglas	Jones
Janice	Joplette
Anna	Mackenzie
Charles	Owen
David	Pinker
Millicent	Purview
Henrietta	Rumney
Ramnaresh	Sarwin
Jack	Smith
Ponder	Stibbons
Henry	Worthington-Smyth

18. Print a list of all members, including the members they have recommended (If none were recommended, return NULL). The list should be arranged by name and surname.

```
SELECT mem.firstname, mem.surname,
        rec.firstname as recFirstname,
        rec .surname as recSurname
FROM cd.members mem
        LEFT OUTER JOIN cd.members rec
            ON rec.memid = mem.recommendedby
ORDER BY firstname, surname;
```

firstname	surname	recfirstname	recsurname
Anna	Mackenzie	Darren	Smith
Anne	Baker	Ponder	Stibbons
Burton	Tracy	NULL	NULL
Charles	Owen	Darren	Smith
Darren	Smith	NULL	NULL
Darren	Smith	NULL	NULL
David	Farrell	NULL	NULL
David	Jones	Janice	Joplette
David	Pinker	Jemima	Farrell
Douglas	Jones	David	Jones
Erica	Crumpet	Tracy	Smith
Florence	Bader	Ponder	Stibbons
Gerald	Butters	Darren	Smith
GUEST	GUEST	NULL	NULL
Henrietta	Rumney	Matthew	Genting
Henry	Worthington-Smyth	Tracy	Smith
Hyacinth	Tupperware	NULL	NULL
Jack	Smith	Darren	Smith
Janice	Joplette	Darren	Smith
Jemima	Farrell	NULL	NULL
Joan	Coplin	Timothy	Baker
John	Hunt	Millicent	Purview
Matthew	Genting	Gerald	Butters
Millicent	Purview	Tracy	Smith
Nancy	Dare	Janice	Joplette
Ponder	Stibbons	Burton	Tracy
Ramnaresh	Sarwin	Florence	Bader
Tim	Boothe	Tim	Rownam
Tim	Rownam	NULL	NULL
Timothy	Baker	Jemima	Farrell
Tracy	Smith	NULL	NULL

19. Print a list of all members so that their last names and first names are displayed in the following format <'Surname, Firstname'> (Apply function CONCAT ())

```
SELECT CONCAT(surname , ', ', firstname ) AS name
FROM cd.members;

# alternatively

SELECT surname || ', ' || firstname AS name
FROM cd.members;
```

name
GUEST, GUEST
Smith, Darren
Smith, Tracy
Rownam, Tim
Joplette, Janice
Butters, Gerald
Tracy, Burton
Dare, Nancy
Boothe, Tim
Stibbons, Ponder
Owen, Charles
Jones, David
Baker, Anne
Farrell, Jemima
Smith, Jack
Bader, Florence
Baker, Timothy
Pinker, David
Genting, Matthew
Mackenzie, Anna
Coplin, Joan
Sarwin, Ramnaresh
Jones, Douglas
Rumney, Henrietta
Farrell, David
Worthington-Smyth, Henry
Purview, Millicent
Tupperware, Hyacinth
Hunt, John
Crumpet, Erica
Smith, Darren

20. Print the initials of all members (assume that each member has only one first name and one last name).

```
SELECT CONCAT(LEFT(surname ,1),'.',LEFT(firstname ,1),'.') AS name
FROM cd.members;
```

name
G.G.
S.D.
S.T.
R.T.
J.J.
B.G.
T.B.
D.N.
B.T.
S.P.
O.C.
J.D.
B.A.
F.J.
S.J.
B.F.
B.T.
P.D.
G.M.
M.A.
C.J.
S.R.
J.D.
R.H.
F.D.
W.H.
P.M.
T.H.
H.J.
C.E.
S.D.