/* Q1: Some of the facilities charge a fee to members, but some do not.
Write a SQL query to produce a list of the names of the facilities that do. */

SELECT name
FROM Facilities
WHERE membercost > 0.0;

name
Tennis Court 1
Tennis Court 2
Massage Room 1
Massage Room 2
Squash Court

/* Q2: How many facilities do not charge a fee to members? */

SELECT COUNT(name) AS have_no_member_fee
FROM Facilities
WHERE membercost = 0.0;

have_no_member_fee
4

/* Q3: Write an SQL query to show a list of facilities that charge a fee to members,
where the fee is less than 20% of the facility's monthly maintenance cost.
Return the facid, facility name, member cost, and monthly maintenance of the
facilities in question. */

SELECT facid, name, membercost, monthlymaintenance
FROM Facilities
WHERE membercost > 0.0
        AND membercost < (monthlymaintenance * 0.2);

facid	name	membercost	monthlymaintenance
0	Tennis Court 1	5.0	200
1	Tennis Court 2	5.0	200
4	Massage Room 1	9.9	3000
5	Massage Room 2	9.9	3000
6	Squash Court	3.5	80

/* Q4: Write an SQL query to retrieve the details of facilities with ID 1 and 5.
Try writing the query without using the OR operator. */

SELECT *
FROM Facilities
WHERE facid IN (1, 5);

facid	name	membercost	guestcost	initialoutlay	mon
1	Tennis Court 2	5.0	25.0	8000	
5	Massage Room 2	9.9	80.0	4000	

/* Q5: Produce a list of facilities, with each labelled as
'cheap' or 'expensive', depending on if their monthly maintenance cost is
more than $100. Return the name and monthly maintenance of the facilities
in question. */

SELECT name, monthlymaintenance,
        (CASE WHEN monthlymaintenance < 100 THEN 'cheap'
    ELSE 'expensive' END) AS maintenance_category
FROM Facilities;

name	monthlymaintenance	maintenance_category
Tennis Court 1	200	expensive
Tennis Court 2	200	expensive
Badminton Court	50	cheap
Table Tennis	10	cheap
Massage Room 1	3000	expensive
Massage Room 2	3000	expensive
Squash Court	80	cheap
Snooker Table	15	cheap
Pool Table	15	cheap

/* Q6: You'd like to get the first and last name of the last member(s)
who signed up. Try not to use the LIMIT clause for your solution. */

```
SELECT firstname, surname
FROM Members
WHERE joindate =
        (SELECT MAX(joindate)
    FROM Members);
```

firstname	surname
Darren	Smith

/* Q7: Produce a list of all members who have used a tennis court.
Include in your output the name of the court, and the name of the member
formatted as a single column. Ensure no duplicate data, and order by
the member name. */

```
SELECT DISTINCT f.name, CONCAT(m.firstname, ' ', m.surname) AS member_name
FROM Bookings b
INNER JOIN Facilities f
        ON b.facid = f.facid
INNER JOIN Members m
        ON b.memid = m.memid
WHERE f.name LIKE '%TENNIS COURT%'
ORDER BY 1,2;
```

ame	member_name
Tennis Court 1	Anne Baker
Tennis Court 1	Burton Tracy
Tennis Court 1	Charles Owen
Tennis Court 1	David Farrell
Tennis Court 1	David Jones
Tennis Court 1	David Pinker
Tennis Court 1	Douglas Jones
Tennis Court 1	Erica Crumpet
Tennis Court 1	Florence Bader
Tennis Court 1	Gerald Butters

Tennis Court 1	GUEST GUEST
Tennis Court 1	Jack Smith
Tennis Court 1	Janice Joplette
Tennis Court 1	Jemima Farrell
Tennis Court 1	Joan Coplin
Tennis Court 1	John Hunt
Tennis Court 1	Matthew Genting
Tennis Court 1	Nancy Dare
Tennis Court 1	Ponder Stibbons
Tennis Court 1	Ramnaresh Sarwin
Tennis Court 1	Tim Boothe
Tennis Court 1	Tim Rownam
Tennis Court 1	Timothy Baker
Tennis Court 1	Tracy Smith
Tennis Court 2	Anne Baker
Tennis Court 2	Burton Tracy
Tennis Court 2	Charles Owen
Tennis Court 2	Darren Smith
Tennis Court 2	David Farrell
Tennis Court 2	David Jones
Tennis Court 2	Florence Bader
Tennis Court 2	Gerald Butters
Tennis Court 2	GUEST GUEST
Tennis Court 2	Henrietta Rumney
Tennis Court 2	Jack Smith
Tennis Court 2	Janice Joplette
Tennis Court 2	Jemima Farrell

Tennis Court 2	John Hunt
Tennis Court 2	Millicent Purview
Tennis Court 2	Nancy Dare
Tennis Court 2	Ponder Stibbons
Tennis Court 2	Ramnaresh Sarwin
Tennis Court 2	Tim Boothe
Tennis Court 2	Tim Rownam
Tennis Court 2	Timothy Baker
Tennis Court 2	Tracy Smith

/* Q8: Produce a list of bookings on the day of 2012-09-14 which
will cost the member (or guest) more than $30. Remember that guests have
different costs to members (the listed costs are per half-hour 'slot'), and
the guest user's ID is always 0. Include in your output the name of the
facility, the name of the member formatted as a single column, and the cost.
Order by descending cost, and do not use any subqueries. */

```sql
SELECT f.name, CONCAT(m.firstname, ' ', m.surname) AS member_name,
       (CASE WHEN b.memid = 0 THEN (b.slots * f.guestcost)
    ELSE (b.slots * f.membercost) END) AS booking_cost
FROM Bookings AS b
INNER JOIN Facilities AS f
       ON b.facid = f.facid
INNER JOIN Members AS m
       ON b.memid = m.memid
WHERE b.starttime LIKE '2012-09-14%'
GROUP BY b.bookid, b.memid, b.slots, f.name, m.firstname, m.surname, f.membercost,
f.guestcost
HAVING (CASE WHEN b.memid = 0 THEN (b.slots * f.guestcost)
    ELSE (b.slots * f.membercost) END) > 30.0
ORDER BY booking_cost DESC;
```

name	member_name	booking_cost ⌄ 1
Massage Room 2	GUEST GUEST	320.0
Massage Room 1	GUEST GUEST	160.0
Massage Room 1	GUEST GUEST	160.0
Massage Room 1	GUEST GUEST	160.0
Tennis Court 2	GUEST GUEST	150.0
Tennis Court 2	GUEST GUEST	75.0
Tennis Court 1	GUEST GUEST	75.0
Tennis Court 1	GUEST GUEST	75.0
Squash Court	GUEST GUEST	70.0
Massage Room 1	Jemima Farrell	39.6
Squash Court	GUEST GUEST	35.0
Squash Court	GUEST GUEST	35.0

/* Q9: This time, produce the same result as in Q8, but using a subquery. */

```
SELECT f.name, CONCAT(m.firstname, ' ', m.surname) AS member_name,
subquery.booking_cost
FROM Bookings AS b
INNER JOIN Facilities AS f
       ON b.facid = f.facid
INNER JOIN Members AS m
       ON b.memid = m.memid
INNER JOIN (SELECT b.bookid, (CASE WHEN b.memid = 0 THEN (b.slots * f.guestcost)
                        ELSE (b.slots * f.membercost) END) AS booking_cost
           FROM Bookings AS b
                   INNER JOIN Facilities AS f
                           ON b.facid = f.facid
                   INNER JOIN Members AS m
                           ON b.memid = m.memid
       WHERE b.starttime LIKE '2012-09-14%'
                   ) AS subquery
           ON b.bookid = subquery.bookid
WHERE subquery.booking_cost > 30.0
ORDER BY booking_cost DESC;
```

name	member_name	booking_cost	1
Massage Room 2	GUEST GUEST	320.0	
Massage Room 1	GUEST GUEST	160.0	
Massage Room 1	GUEST GUEST	160.0	
Massage Room 1	GUEST GUEST	160.0	
Tennis Court 2	GUEST GUEST	150.0	
Tennis Court 2	GUEST GUEST	75.0	
Tennis Court 1	GUEST GUEST	75.0	
Tennis Court 1	GUEST GUEST	75.0	
Squash Court	GUEST GUEST	70.0	
Massage Room 1	Jemima Farrell	39.6	
Squash Court	GUEST GUEST	35.0	
Squash Court	GUEST GUEST	35.0	