/\* Welcome to the SQL mini project. For this project, you will use

Springboard' online SQL platform, which you can log into through the

following link:

https://sql.springboard.com/

Username: student

Password: learn\_sql@springboard

The data you need is in the "country\_club" database. This database

contains 3 tables:

i) the "Bookings" table,

ii) the "Facilities" table, and

iii) the "Members" table.

Note that, if you need to, you can also download these tables locally.

In the mini project, you'll be asked a series of questions. You can

solve them using the platform, but for the final deliverable,

paste the code for each solution into this script, and upload it

to your GitHub.

Before starting with the questions, feel free to take your time,

exploring the data, and getting acquainted with the 3 tables. \*/

/\* Q1: Some of the facilities charge a fee to members, but some do not.

Please list the names of the facilities that do. \*/

SELECT name,membercost,guestcost

FROM `Facilities`

WHERE membercost!=0

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

SELECT COUNT(\*)

FROM `Facilities`

WHERE membercost=0

/\* Q3: How can you produce 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<.2\*monthlymaintenance

/\* Q4: How can you retrieve the details of facilities with ID 1 and 5?

Write the query without using the OR operator. \*/

SELECT \*

FROM `Facilities`

WHERE facid IN(1,5)

/\* Q5: How can you 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 cost\_type

FROM `Facilities`

/\* Q6: You'd like to get the first and last name of the last member(s)

who signed up. Do not use the LIMIT clause for your solution. \*/

SELECT memb.firstname,memb.surname

FROM `Bookings` book

LEFT JOIN `Members` memb

ON book.memid=memb.memid

WHERE starttime IN (SELECT max(starttime) FROM `Bookings`) AND book.memid!=0

/\* Q7: How can you 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 facilities.name AS Court\_Name,CONCAT(member.firstname,' ',member.surname) AS Full\_Name

FROM `Bookings` booking

LEFT JOIN `Facilities` facilities

ON booking.facid=facilities.facid

LEFT JOIN `Members` member

ON booking.memid=member.memid

WHERE facilities.facid IN(0,1) AND booking.memid!=0

GROUP BY booking.memid,member.firstname,member.surname

ORDER BY member.firstname

/\* Q8: How can you 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. \*/

SELECT f.name AS fac\_name,CONCAT(m.firstname,' ',m.surname) AS Full\_Name,

CASE WHEN b.memid=0 THEN f.guestcost\*b.slots

ELSE f.membercost\*b.slots END AS session\_cost

FROM `Bookings` b

LEFT JOIN `Facilities` f

ON b.facid=f.facid

LEFT JOIN `Members` m

ON b.memid=m.memid

WHERE starttime LIKE '%2012-09-14%'

HAVING session\_cost>30

ORDER BY fac\_name

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

SELECT sub.Fac\_Name, CONCAT(sub.firstname," ",sub.surname) AS Full\_Name,sub.session\_cost

FROM

(SELECT f.name as Fac\_Name,m.firstname,m.surname,

CASE WHEN b.memid=0 THEN f.guestcost\*b.slots

ELSE f.membercost\*b.slots END AS session\_cost

FROM Bookings b

JOIN Facilities f

ON b.facid=f.facid

JOIN Members m

ON b.memid=m.memid

WHERE b.starttime LIKE "%2012-09-14%") sub

WHERE sub.session\_cost>30

/\* Q10: Produce a list of facilities with a total revenue less than 1000.

The output of facility name and total revenue, sorted by revenue. Remember

that there's a different cost for guests and members! \*/

SELECT sub.fac\_name,SUM(sub.revenue) AS aggregate\_revenue

FROM

(SELECT f.name AS fac\_name,

CASE WHEN m.memid=0 THEN f.guestcost\*b.slots

ELSE f.membercost\*b.slots END AS revenue

FROM `Bookings` b

JOIN `Facilities` f

ON b.facid=f.facid

JOIN `Members` m

ON b.memid=m.memid

) sub

GROUP BY fac\_name

HAVING aggregate\_revenue<1000