### **Simple SQL Queries**

**Retrieve everything from a table.**

1. How can you retrieve all the information from the Members table? Submit your SQL code for this query.

select \* from cd.members;

**Retrieve specific columns from a table.**

1. How can you retrieve a list of members' full names and their telephone numbers? Submit your SQL code for this query.

select name, telephone from cd.members;

**Control which rows are retrieved.**

1. How can you produce a list of facilities that charge a guest cost of greater than $50? Submit your SQL code for this query.

select \* from cd.facilities where membercost > 50;

1. Which facilities charge a guest cost between $20 and $50?

select \* from cd.facilities where membercost < 50 and membercost > 20

**Control which rows are retrieved – part 2**

1. How can you produce a list of facilities that (1) have a member cost and (2) have a guest fee that is more than three times the member cost? Only list the facility name, the member cost, and the guest fee. Submit your SQL code for this query.

Select name, membercost, guestfee

 from cd.facilities

 where

 guestfee > 0 and

 (guestfee > membercost\*3);

**Basic string searches**

1. You want a list of all of the different types of courts in this database. Give me a query that lists all information about any facility that has 'Court' in its name.

select \*

 from cd.facilities

 where

 name like '%Court%';

**Matching against multiple possible values**

 7. Write two queries that gives me a list of all members that live in areas with zip codes 4321, 87630, or 28563. Produce a table that only lists their firstname, surname, and telephone number. List two queries – one where you use an OR operator and one where you do not use an OR operator.

select firstname, surname, telephone

 from cd.members

 where

 zipcode in (4321, 87630, 28563);

………………………………………………………………………………..

select firstname, surname, telephone

 from cd.members

 where

 zipcode = 4321, 87630 or 28563;

**Classify results into buckets**

8. I want you to produce a query that will list facilities based on whether they have a cheap, reasonable, or expensive guest cost. Guest costs of under $20 are cheap, costs of between $21 – $50 are reasonable, and costs of over $50 are expensive. Your SQL query should return a column of the names of the facilities and a column labeled 'CostForGuests' that lists whether the guest cost is cheap, reasonable, or expensive. Submit your SQL code. (1 pt)

Hint: You can repeat 'when... then...' clauses in order to produce multiple categories.

Hint 2: Make sure that you do not add a space to 'GuestCost'.

select name,

 case when (guestcost > 50) then

 'expensive'

 case when (guestcost > 21 and guescost < 50) then

 'reasonable'

case when (guestcost < 50) then

 'cheap'

 end as costforguests

 from cd.facilities;

**Working with dates**

9. Submit code that produces a list of members who joined before August 10, 2012, your output should include: firstname, surname and date joined.

select memid, surname, firstname, joindate

 from cd.members

 where joindate <= '2012-08-10';

**Removing duplicates and ordering results**

10. Create an ordered list of guest costs that does not contain any duplicates. Submit your SQL code.

select distinct guestcost

 from cd.members

order by guestcost

### **Joins and Subqueries**

**Retrieve the start times of members' bookings**

11. Give me a query that would list the start times for all bookings by member Tim Rownam.

select bks.starttime

 from

 cd.bookings bks

 inner join cd.members mems

 on mems.memid = bks.memid

 where

 mems.firstname='Tim'

 and mems.surname='Rownam';

12. Give me a query that would list all of the facility IDs booked by member Tim Rownam. Make sure that there are no duplicates, and organize the list so that they are in order by facility ID.

select bks.facid

from

 cd.bookings bks

 inner join cd.members mems

 on mems.memid = bks.memid

 where

 mems.firstname='Tim'

 and mems.surname='Rownam';

**Produce a list of all members who have recommended another member**

13. Produce a query that will produce the first and last name of the member that recommended Gerald Butters.

select recs.firstname as firstname, recs.surname as surname

 from

 cd.members mems

 inner join cd.members recs

 on recs.memid = geraldbutters

**Produce a list of all members, along with their recommender**

14. Submit the query that you wrote for this exercise.

select mems.firstname as memfname, mems.surname as memsname, recs.firstname as

recfname, recs.surname as recsname

 from

 cd. members mems

 left outer join cd.members recs

 on recs.memid = mems.recommendedby

order by memsname, memfname;

**Produce a list of all members who have used a tennis court**

15. Produce a query that will list all members that paid at least $5 in member costs to rent a facility (any facility). Include in your output the member name (formatted as a single column), and then a single column that lists the facility name and the member cost (separated by a comma). The second column should include values such as: “Tennis Court 2, 5”. Label the second column 'facilityandcost'. Ensure no duplicate data, and order by the member name.

select distinct mems.firstname || ' ' || mems.surname as member, facs.name as facility

 from

 cd.members mems

 inner join cd.bookings bks

 on mems.memid = bks.memid

 inner join cd.facilities facs

 on bks.facid = facs.facid

 where

 facs.name in ('Tennis Court 2','Tennis Court 5')

 and

 (mems.memid = 0 and bks.slots\*facs.guestcost > 5) or

 (mems.memid != 0 and bks.slots\*facs.membercost > 5)

 )

order by facility, cost

### **Aggregation**

**Count the number of facilities**

16. Give me a SQL command that lists the total number of rows in the cd.members table. Submit the command.

select row,

 (select count (\*) from cd.members)

 from cd.members

17. Give me a SQL command that lists the total number of distinct members, without duplicates. Submit the command.

select memid,

 (select count (\*) from cd.members)

 from cd.members

**Count the number of expensive facilities**

18. Give me a command that lists the total number of facilities that cost a member more than $5. Submit the command.

select count (\*) from cd.facilities where membercost >= 5;

**List the total slots booked per facility**

19. List the command you produced to complete this exercise.

select facid, sum(slots) as “Total Slots”

 from cd.bookings

 group by facid

order by facid;

**Find the total revenue of each facility**

20. Give me a command that lists only the total revenue generated from members, per facility.

select facs.name, sum(slots \* case

 when memid = 0 then facs.guestcost

 else facs.membercost

 end) as revenue

from cd.bookings bks

inner join cd.facilities facs

on bks.facid = facs.facid

group by facs.name

order by revenue;