Oprea Sergiu-Daniel / Gr. 30322 – Homework – Lab3

More JOIN operations

1. List the films where the **yr** is 1962 [Show **id**, **title**]

SELECT id, title

FROM movie

WHERE yr=1962

2. Give year of 'Citizen Kane'.

SELECT yr

FROM movie

WHERE title = 'Citizen Kane'

3. List all of the Star Trek movies, include the **id**, **title** and **yr** (all of these movies include the words Star Trek in the title). Order results by year.

SELECT id, title, yr FROM movie

WHERE title LIKE '%Star Trek%'

ORDER BY yr

4. What **id** number does the actor 'Glenn Close' have?

SELECT id FROM actor

WHERE name = 'Glenn Close'

5. What is the **id** of the film 'Casablanca'?

SELECT id FROM movie

WHERE title = 'Casablanca'

6. Obtain the cast list for 'Casablanca'.

Use **movieid=11768**, (or whatever value you got from the previous question)

SELECT name FROM casting JOIN actor ON (id=actorid)

WHERE movieid=11768

7. Obtain the cast list for the film 'Alien'

SELECT name FROM casting

JOIN actor ON (actor.id=actorid)

JOIN movie ON (movie.id=movieid)

WHERE title = 'Alien'

8. List the films in which 'Harrison Ford' has appeared

SELECT title FROM casting

JOIN movie ON (movie.id = movieid)

JOIN actor ON (actor.id = actorid)

WHERE name = 'Harrison Ford'

9. List the films where 'Harrison Ford' has appeared - but not in the starring role. [Note: the **ord** field of casting gives the position of the actor. If ord=1 then this actor is in the starring role]

SELECT title FROM casting

JOIN movie ON (movie.id = movieid)

JOIN actor ON (actor.id = actorid)

WHERE name = 'Harrison Ford' AND ord > 1

10. List the films together with the leading star for all 1962 films.

SELECT title, name FROM casting

JOIN movie ON (movie.id = movieid)

JOIN actor ON (actor.id = actorid)

WHERE yr = 1962 and ord = 1

11. Which were the busiest years for 'Rock Hudson', show the year and the number of movies he made each year for any year in which he made more than 2 movies.

SELECT yr,COUNT(title) FROM

movie JOIN casting ON movie.id=movieid

JOIN actor ON actorid=actor.id

WHERE name='Rock Hudson'

GROUP BY yr

HAVING COUNT(title) > 2  
12. List the film title and the leading actor for all of the films 'Julie Andrews' played in.

SELECT title, name FROM casting

JOIN movie ON movie.id = movieid

JOIN actor ON actor.id = actorid

WHERE ord = 1

AND movie.id IN

(SELECT movie.id FROM movie

JOIN casting ON movie.id = movieid

JOIN actor ON actor.id = actorid

WHERE actor.name = 'Julie Andrews')  
13. Obtain a list, in alphabetical order, of actors who've had at least 15 **starring** roles.

SELECT DISTINCT name FROM casting

JOIN movie ON movie.id = movieid

JOIN actor ON actor.id = actorid

WHERE actorid IN (

SELECT actorid FROM casting

WHERE ord = 1

GROUP BY actorid

HAVING COUNT(actorid) >= 15)

ORDER BY name

14. List the films released in the year 1978 ordered by the number of actors in the cast, then by title.

SELECT title, COUNT(actorid)

FROM movie, casting

WHERE yr = '1978'

AND casting.movieid = movie.id

GROUP BY title

ORDER BY 2 DESC, 1;

15. List all the people who have worked with 'Art Garfunkel'.

SELECT DISTINCT name FROM casting

JOIN actor ON actorid = actor.id

WHERE name != 'Art Garfunkel'

AND movieid IN (

SELECT movieid

FROM movie

JOIN casting ON movieid = movie.id

JOIN actor ON actorid = actor.id

WHERE actor.name = 'Art Garfunkel')

# Using Null

1. List the teachers who have NULL for their department.

SELECT name FROM teacher

WHERE dept IS NULL

2. Note the INNER JOIN misses the teachers with no department and the departments with no teacher.

SELECT teacher.name, dept.name

FROM teacher INNER JOIN dept

ON (teacher.dept=dept.id)

3. Use a different JOIN so that all teachers are listed.

SELECT teacher.name, dept.name

FROM teacher LEFT JOIN dept

ON (teacher.dept=dept.id)

4. Use a different JOIN so that all departments are listed.

SELECT teacher.name, dept.name

FROM teacher RIGHT JOIN dept

ON (teacher.dept=dept.id)

5. Use COALESCE to print the mobile number. Use the number '07986 444 2266' if there is no number given. **Show teacher name and mobile number or '07986 444 2266'**

SELECT teacher.name, COALESCE(teacher.mobile,'07986 444 2266') FROM teacher

6.Use the COALESCE function and a LEFT JOIN to print the teacher **name** and department name. Use the string 'None' where there is no department.

SELECT teacher.name, COALESCE(dept.name,'None') FROM teacher

LEFT JOIN dept ON teacher.dept = dept.id

7. Use COUNT to show the number of teachers and the number of mobile phones.

SELECT COUNT(name), COUNT(mobile) FROM teacher

8. Use COUNT and GROUP BY **dept.name** to show each department and the number of staff. Use a RIGHT JOIN to ensure that the Engineering department is listed.

SELECT dept.name, COUNT(teacher.dept) FROM teacher

RIGHT JOIN dept ON dept.id = teacher.dept

GROUP BY dept.name

9. Use CASE to show the **name** of each teacher followed by 'Sci' if the teacher is in **dept** 1 or 2 and 'Art' otherwise.

SELECT name, CASE WHEN dept IN (1,2) THEN 'Sci'

ELSE 'Art'

END

FROM teacher

10. Use CASE to show the name of each teacher followed by 'Sci' if the teacher is in dept 1 or 2, show 'Art' if the teacher's dept is 3 and 'None' otherwise.

SELECT name, CASE WHEN dept IN (1,2) THEN 'Sci'

WHEN dept = 3 THEN 'Art'

ELSE 'None'

END

FROM teacher

# Self join

1. How many **stops** are in the database.

SELECT DISTINCT COUNT(\*) FROM stops

2. Find the **id** value for the stop 'Craiglockhart'

SELECT id FROM stops

WHERE name = 'Craiglockhart'

3. Give the **id** and the **name** for the **stops** on the '4' 'LRT' service.

SELECT id, name

FROM stops, route

WHERE stops.id = route.stop

AND num = '4'

AND company = 'LRT';  
4. The query shown gives the number of routes that visit either London Road (149) or Craiglockhart (53). Run the query and notice the two services that link these **stops** have a count of 2. Add a HAVING clause to restrict the output to these two routes.

SELECT company, num, COUNT(\*)

FROM route WHERE stop=149 OR stop=53

GROUP BY company, num

HAVING COUNT(\*) = 2

5. Execute the self join shown and observe that b.stop gives all the places you can get to from Craiglockhart, without changing routes. Change the query so that it shows the services from Craiglockhart to London Road.

SELECT a.company, a.num, a.stop, b.stop

FROM route a JOIN route b ON

(a.company=b.company AND a.num=b.num)

WHERE a.stop=53 AND b.stop = (SELECT id FROM stops WHERE name = 'London Road')

6. The query shown is similar to the previous one, however by joining two copies of the **stops** table we can refer to **stops** by **name** rather than by number. Change the query so that the services between 'Craiglockhart' and 'London Road' are shown. If you are tired of these places try 'Fairmilehead' against 'Tollcross'

SELECT a.company, a.num, stopa.name, stopb.name

FROM route a JOIN route b ON

(a.company=b.company AND a.num=b.num)

JOIN stops stopa ON (a.stop=stopa.id)

JOIN stops stopb ON (b.stop=stopb.id)

WHERE stopa.name='Craiglockhart' and stopb.name = 'London Road'

7. Give a list of all the services which connect stops 115 and 137 ('Haymarket' and 'Leith').

SELECT DISTINCT R1.company, R1.num

FROM route R1, route R2

WHERE R1.num = R2.num

AND R1.company = R2.company

AND R1.stop = '115'

AND R2.stop = '137';

8. Give a list of the services which connect the **stops** 'Craiglockhart' and 'Tollcross'

SELECT a.company, a.num

FROM route a

JOIN route b ON (a.company = b.company AND a.num = b.num)

JOIN stops stopa ON a.stop = stopa.id

JOIN stops stopb ON b.stop = stopb.id

WHERE stopa.name = 'Craiglockhart'

AND stopb.name = 'Tollcross';

9. Give a distinct list of the **stops** which may be reached from 'Craiglockhart' by taking one bus, including 'Craiglockhart' itself, offered by the LRT company. Include the company and bus no. of the relevant services.

SELECT DISTINCT name, a.company, a.num

FROM route a

JOIN route b ON (a.company = b.company AND a.num = b.num)

JOIN stops ON a.stop = stops.id

WHERE b.stop = 53;

10. Find the routes involving two buses that can go from **Craiglockhart** to **Lochend**.  
Show the bus no. and company for the first bus, the name of the stop for the transfer,  
and the bus no. and company for the second bus.

SELECT S.num, S.company, S.name, T.num, T.company

FROM

(SELECT DISTINCT a.num, a.company, sb.name

FROM route a JOIN route b ON (a.num = b.num and a.company = b.company)

JOIN stops sa ON sa.id = a.stop

JOIN stops sb ON sb.id = b.stop

WHERE sa.name = 'Craiglockhart' AND sb.name <> 'Craiglockhart'

)S

JOIN

(SELECT x.num, x.company, sy.name

FROM route x JOIN route y ON (x.num = y.num and x.company = y.company)

JOIN stops sx ON sx.id = x.stop

JOIN stops sy ON sy.id = y.stop

WHERE sx.name = 'Lochend' AND sy.name <> 'Lochend'

)T

ON (S.name = T.name)

ORDER BY S.num, S.name, T.num

Top of Form

Top of Form

Top of Form

Top of Form