Self join

Edinburgh Buses

Details of the database Looking at the data

```
stops(id, name)
route(num, company, pos, stop)
```

stops
id
name

route
num
company
pos
stop

Summary

1.

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

```
select count(distinct id)
from stops
```

Submit SQL Restore default

Correct answer

count(distinc..
246

2.

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

```
select id
from stops
where name='Craiglockhart'
```

Submit SQL Restore default

Correct answer

id
53

3.

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

```
select id, name
from stops join route on id=stop
where company='LRT' and num=4
order by pos
```

Submit SQL Restore default

Correct answer

id	name
19	Bingham
177	Northfield
149	London Road
194	Princes Street
115	Haymarket
53	Craiglockhart
179	Oxgangs
85	Fairmilehead

Routes and stops

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(*)>1
```

Submit SQL Restore default

Correct answer

company	num	COUNT(*)
LRT	4	2
LRT	45	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=149
```

Submit SQL Restore default

Correct answer

company	num	stop	stop
LRT	4	53	149
LRT	45	53	149

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'
```

Submit SQL Restore default

Correct answer

company	num	name	name
LRT	4	Craiglockhart	London Road
LRT	45	Craiglockhart	London Road

Using a self join

7.

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

```
SELECT distinct 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.id=115 and stopb.id=137
```

Submit SQL Restore default

Correct answer

company	num
LRT	12
LRT	2
LRT	22
LRT	25
LRT	2A
SMT	C5

8.

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

```
SELECT distinct 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'
```

Submit SQL Restore default

Correct answer

company	num
LRT	10
LRT	27
LRT	45
LRT	47

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 stopb.name, b.company, b.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'
```

Submit SQL Restore default

Correct answer

name	company	num
Silverknowes	LRT	10
Muirhouse	LRT	10
Newhaven	LRT	10
Leith	LRT	10
Leith Walk	LRT	10
Princes Street	LRT	10
Tollcross	LRT	10
Craiglockhart	LRT	10

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.

Hint

Self-join twice to find buses that visit Craiglockhart and Lochend, then join those on matching stops.

```
SELECT a.num, a.company, stopa.name, b.num, b.company
FROM route a
JOIN route b ON (a.num = b.num)
JOIN stops stopa ON (a.stop = stopa.id)
JOIN stops stopb ON (b.stop = stopb.id)
WHERE stopa.name = 'Craiglockhart'
```

Submit SQL Restore default

Result:

num	company	name	num	company

Show what the answer should be...

Clear your results

Self join Quiz