In [3]:
```
%load_ext sql
%sql sqlite://
```

```
/usr/local/lib/python2.7/site-packages/IPython/config.py:13: ShimW
arning: The `IPython.config` package has been deprecated. You shou
ld import from traitlets.config instead.
  "You should import from traitlets.config instead.", ShimWarning)
/usr/local/lib/python2.7/site-packages/IPython/utils/traitlets.p
y:5: UserWarning: IPython.utils.traitlets has moved to a top-level
traitlets package.
  warn("IPython.utils.traitlets has moved to a top-level traitlets
package.")
```

Out[3]:
```
'Connected: None@None'
```

Equivalence Activity

First load the following tables:

In [30]:
```
%%sql
DROP TABLE IF EXISTS courses;
CREATE TABLE courses (course TEXT, staff TEXT, students int, hours
int);
INSERT INTO courses VALUES ('CS103','Keith', 320, 4);
INSERT INTO courses VALUES ('CS145','Chris', 200, 3);
INSERT INTO courses VALUES ('CS245','Chris', 100, 4);
INSERT INTO courses VALUES ('CS161','Virginia', 400, 3);
INSERT INTO courses VALUES ('CS267','Virginia', 50, 4);
INSERT INTO courses VALUES ('CS224N','Chris', 250, 4);
INSERT INTO courses VALUES ('CS276','Chris', 300, 3);
```

```
Done.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
```

Out[30]:
```
[]
```

In [29]:
```
%%sql
DROP TABLE IF EXISTS researchers;
CREATE TABLE researchers (topic TEXT, staff TEXT, phds int);
INSERT INTO researchers VALUES ('Databases','Chris', 6);
INSERT INTO researchers VALUES ('NLP','Chris', 8);
INSERT INTO researchers VALUES ('Graph Theory','Virginia', 5);
```

```
Done.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
```

Out[29]: []

Overview of some SQL statements:

We will be learning about:

- HAVING
- ANY (not supported in SQLite)
- ALL (not supported in SQLite)

As well as DeMorgan's Laws for using NOT before these clauses

HAVING

We can use this to condition on aggregates, as well as grouping by specific attributes

In [31]:
```
%%sql
SELECT staff
FROM courses
GROUP BY staff
HAVING Sum(students) > 500
```

```
Done.
```

Out[31]:

staff
Chris

What happens when you try selecting something that is not grouped? See for yourself, does this make any sense?

In [35]:
```
%%sql
SELECT students
FROM courses
GROUP BY staff
HAVING Sum(students) > 500
```

```
Done.
```

Out[35]:

students
300

Activity 1

Can you rewrite the following query using the HAVING clause?

In [50]:
```
%%sql
SELECT staff
FROM courses
WHERE students = (
    SELECT MAX(students)
    FROM courses
)
```

```
Done.
```

Out[50]:

staff
Virginia

In [46]:
```
# Fill in your SQL query here, using the HAVING clause
```

ANY

This clause checks to see if a specific attribute is equivalent to any (at least one) tuple generated by a subquery.

SELECT DISTINCT staff FROM courses WHERE staff = ANY (SELECT staff FROM researchers)

This query is not possible in SQLite as ANY is not supported, but we can model the functionality of it using the "IN" clause:

In [42]:
```
%%sql
SELECT DISTINCT staff
FROM courses
WHERE staff in (
    SELECT staff
    FROM researchers
)
```

```
Done.
```

Out[42]:

staff
Chris
Virginia

ALL

The ALL clause is almost always used in conjunction with <> (not equals), as comparing a specific attribute with the results of a subquery using ALL would require the subquery to contain all duplicate tuples. What would the following query result in if ALL were supported in SQLite?

SELECT DISTINCT staff FROM courses WHERE staff <> ALL (SELECT staff FROM researchers)

Activity 2

Can you find all the queries (without running them) that are equivalent to the below query?

In [53]:
```
%%sql
SELECT c.course FROM courses as c WHERE hours = 4 AND EXISTS (
    SELECT * FROM researchers as r WHERE r.staff = c.staff
)
```

```
Done.
```

Out[53]:

course
CS245
CS267
CS224N

Query 1: SELECT course FROM courses WHERE hours = 4 AND staff in (SELECT staff FROM researchers)
Query 2: SELECT course FROM courses as c WHERE hours = 4 AND staff <> ALL (SELECT staff FROM researchers as r WHERE c.staff != r.staff)Query 3: SELECT course FROM courses as c, researchers as r WHERE c.hours = 4 AND c.staff = r.staff

Answer:

In [ ]: