MISSION NAME : SUBQUERIES

* SELECT Major, ShareWomen FROM recent-grads WHERE ShareWomen > (SELECT AVG(ShareWomen) FROM recent-grads)

* A subquery is a query nested within other subquery.

* A subquery must ALWAYS be contained in Parenthesis.

* Display all mws whose unemployment rate is below average:

Soln :-

SELECT Major, Unemployment-rate from recent-grads WHERE Unemployment-rate < (SELECT AVG(Unemployment-rate) FROM recent-grads)

* Subquery in select :-

SELECT CAST(COUNT(*) AS FLOAT) /
(SELECT COUNT(*) FROM recent-grads)
AS proportion-abv-avg FROM recent-grads
WHERE ShareWomen > (SELECT AVG(ShareWomen)
FROM recent-grads);

Operators :-

Highest

Precedence

Lowest

||
* / %
+ -
<< >> & |
< <= > >=
= == != <> IS IS NOT IN LIKE GLOB MATCH REGEXP
AND
OR

* IN operator :

Using IN we can specify a list of values that we want to
match (by equality) against the WHERE clause.

* Eg:-
SELECT Major-category, Major
FROM recent grads
WHERE Major-category
IN ('Business', 'Humanities & Liberal Arts', 'Education')

* Returning Multiple Results in Subqueries:

Eg :-
SELECT Major-category, Major
FROM recent-grads
WHERE Major-category IN (SELECT Major-category
                          FROM recent grads
                          GROUP BY Major-category
                          ORDER BY SUM(Total) DESC
                          LIMIT 3)

* Write a query that returns the average ratio between
Sample-size and Total for all the majors :-

SELECT AVG(CAST(Sample-size AS Float)/Total) AS
avg-ratio FROM recent grads.

* Integrating subquery with the outer subquery :-

Eg:- SELECT Major, Major-category, CAST(Sample-size AS FLOAT)/
                                    Total AS Ratio.
     FROM recent grads
     WHERE ratio > (SELECT AVG(CAST(Sample-size AS Float)/Total)
            AS avg-ratio FROM recent-grads
            );