Working With Data Sources

Course Name: SQL Fundmentals

Mission Name: Introduction to SQL

* A database structures data much like how we would see it in a spreadsheet file, where data is organised in different tables which in turn are organised into rows and columns.

* SQLite is a lightweight DBMS and is the most popular database in the world.

* SELECT * FROM recent_grads;
  ↳ Selects all the columns

The above command will print entire table recent_grads.

* LIMIT → SQL allows us to limit the number of rows that are returned using this clause.

Eg:- SELECT * FROM recent_grads LIMIT 3;

* SQL dialects exists and we should watch out for differences when switching between DBMS's.

* SELECT MAJOR, ShareWomen FROM recent_grads;

The above query selects columns named Major & ShareWomen from the table recent_grads;

* To filter rows by specific criteria, we can use the WHERE statement.

Eg:- SELECT Major, ShareWomen from recent_grads
WHERE ShareWomen <0.5;

* Expressing Multiple Filter Criteria Using 'AND' :-
SELECT Major, Major_Category, Median, ShareWomen FROM recent_grads WHERE ShareWomen>0.5 and Median < 50000

* Returning one of several conditions with OR :-

```
SELECT Major, Median, Unemployed from recent_grad
WHERE Median >= 10000 OR Men > Women LIMIT 20
```

* Grouping Operators With Parentheses:-

```
SELECT Major, Major_Category, ShareWomen,
       Unemployment_Rate
FROM recent_grads
WHERE Major_Category = "Engineering"
AND (ShareWomen > 0.5 OR Unemployment_rate
                          < 0.051)
```

* To have more control over how the results are ordered, we can specify the order using ORDER BY clause.

Eg:-
```
SELECT Major, ShareWomen, Unemployment_rate
FROM recent_grads
WHERE (ShareWomen > 0.3 AND Unemployment_rate
                          < 0.1)
ORDER BY ShareWomen DESC.
```

*
```
SELECT Major_category, Major, Unemployment_rate
FROM recent_grads
WHERE (Major_Category = 'Engineering" OR
       Major_Category = "Physical Engineering")
ORDER BY Unemployment_rate
```