SQL Basics Review

The Big 6 are the foundation clauses used in SQL

```
SELECT   grade_level          -> Columns to display
         AVG(gpa) AS avg_gpa
FROM     students             -> table to pull data from
WHERE    school_lunch = 'Yes' -> Criteria to filter the rows
GROUP BY grade_level          -> Column to group the rows by
HAVING   avg_gpa < 3.3        -> Criteria to filter the grouped rows
ORDER BY grade_level          -> Column to sort values by
```

the clauses must always be written in this order

Start Fridays With Grandmas Homemade Oatmeal

The only required clause in a SQL query is the SELECT

Common SQL keywords

in addition to the Big 6, there are common SQL keywords used in queries.
these are popular keywords found in the SELECT clause

```
SELECT DISTINCT grade_level -> DISTINCT returns unique values
FROM students;
```

```
SELECT COUNT(DISTINCT grade_level) -> Aggregate functions like COUNT, SUM, AVG, MIN, MAX are used to make calculations
FROM students
```

```
SELECT MAX(gpa) - MIN(gpa) AS gpa_range
FROM students;
```

AS renames a column or table to an alias

Math operators include +, -, x, /, %

SELECT

These are popular Key-words found in the WHERE clause:

```
SELECT *
FROM students
WHERE grade_level < 12 AND school_lunch = 'yes';
```

Comparison operators include =, !=, <, >, <=, >=

Logical operators include AND OR NOT

```
SELECT *
FROM students
~~ORDER BY gpa DESC;~~
~~WHERE grade_level < 12 AND school~~
WHERE grade_level IN (10,11,12);
```

Comparison Keywords include IN, LIKE, BETWEEN, AND, IS NULL

```
SELECT *
FROM students
WHERE email LIKE '%.com';
```

Other Popular Keywords

```
SELECT student_name, gpa
FROM students
ORDER BY gpa DESC;
```

→ DESC stands for "descending", while the default parameter is ASC

```
SELECT * *
FROM students
LIMIT 10
```

→ LIMIT specifies the number of rows in the output (TOP in SQL server)

```
SELECT student_name, grade_level
    CASE WHEN grade_level = 9 THEN 'freshman'
         WHEN grade_level = 10 THEN 'Sophomore'
         WHEN grade_level = 11 THEN 'Junior'
         ELSE 'Senior' END AS student_class
FROM students;
```

this creates a new student_class column based on the grade level for each student

Case statements use the following syntax to do If-ELSE logic within SQL:

```
CASE WHEN ... THEN ...
     WHEN ... THEN ...
     ELSE ... END
```