Recap

Commands

You have already learned a lot about writing code in SQL! Let's take a moment t
covered before moving on:

Statement	How to Use It	Other Details
SELECT	SELECT **Col1**, **Col2**, ...	Provide the columns you war
FROM	FROM **Table**	Provide the table where the c
LIMIT	LIMIT **10**	Limits based number of rows
ORDER BY	ORDER BY **Col**	Orders table based on the co **DESC**.
WHERE	WHERE **Col > 5**	A conditional statement to fil
LIKE	WHERE **Col LIKE '%me%'**	Only pulls rows where colum text
IN	WHERE **Col IN ('Y', 'N')**	A filter for only rows with colu
NOT	WHERE **Col NOT IN ('Y', 'N')**	**NOT** is frequently used with **I**
AND	WHERE **Col1 > 5 AND Col2 < 3**	Filter rows where two or mor true
OR	WHERE **Col1 > 5 OR Col2 < 3**	Filter rows where at least one true
BETWEEN	WHERE **Col BETWEEN 3 AND 5**	Often easier syntax than usin

Other Tips

Though SQL is **not case sensitive** (it doesn't care if you write your statements a
lowercase), we discussed some best practices. **The order of the key words doe**
you know so far, you will want to write your statements as:

```
SELECT col1, col2
FROM table1
WHERE col3  > 5 AND col4 LIKE '%os%'
ORDER BY col5
LIMIT 10;
```

Notice, you can retrieve different columns than those being used in the **ORDER**
statements. Assuming all of these column names existed in this way (`col1`, `co`
`col5`) within a table called `table1`, this query would run just fine.

Looking Ahead

In the next lesson, you will be learning about **JOIN**s. This is the real secret (well r
behind the success of SQL as a language. **JOIN**s allow us to combine multiple tal
operations we learned here will still be important moving forward, but we will b
more complex questions by combining information from multiple tables! You ha
much - potentially writing your first code ever, but it is about to get so much bet