

VIDEO 2 OF 8

SQL Basics

SELECT · WHERE · ORDER BY · Calculated Columns

SQL for Data Analysis with Google BigQuery

Instructor: Uttam Deb
Akaademy Ltd.



What We'll Cover Today



SELECT & DISTINCT

Retrieve specific columns,
find unique values

WHERE & Filtering

Filter rows using conditions,
AND / OR / IN / BETWEEN / LIKE

ORDER BY & LIMIT

Sort results and control
how many rows you see

Aliases & Calculated Columns

Rename columns, build
new metrics from existing ones



SELECT · FROM · LIMIT · DISTINCT

LIVE DEMO

- SELECT — choose which columns to return
- FROM — which table to query
- LIMIT — cap the number of rows returned
- DISTINCT — unique values only

⚠ Good habit: avoid SELECT * in BigQuery (costs more)

```
-- All rows, all columns (avoid for big tables!)
SELECT *
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
LIMIT 10;

-- Only the columns you need
SELECT OrderID, Amount, Profit, Category
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
LIMIT 10;

-- Unique values in a column
SELECT DISTINCT Category
FROM `nth-guide-448810-h6.sql_course.sales_dataset`;

SELECT DISTINCT State
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
ORDER BY State;
```

WHERE — Filtering Rows



LIVE DEMO

Comparison Operators

<code>=</code>	equals
<code>!=</code> or <code><></code>	not equal
<code>></code> / <code><</code>	greater / less than
<code>>=</code> / <code><=</code>	greater or equal / less or equal
AND	both conditions must be true
OR	either condition can be true

```
-- Orders from Florida only
SELECT *
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE State = 'Florida'
LIMIT 10;

-- High-value Electronics orders
SELECT OrderID, Amount, Category, State
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE Category = 'Electronics'
      AND Amount > 5000;

-- Furniture orders with a loss
SELECT OrderID, Amount, Profit, Sub_Category
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE Category = 'Furniture'
      AND Profit < 0;
```



WHERE — IN, BETWEEN, LIKE

LIVE DEMO

IN

Match any value in a list
Shorter than many OR conditions

BETWEEN

Match a range (inclusive)
Works on numbers and dates

LIKE

Pattern matching on strings
% = any characters, _ = one character

```
-- IN: multiple states at once
SELECT OrderID, Amount, State
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE State IN ('Florida', 'California', 'Texas');
```

```
-- BETWEEN: orders with amount in a range
SELECT OrderID, Amount, Category
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE Amount BETWEEN 5000 AND 9000;
```

```
-- LIKE: customer names containing 'Johnson'
SELECT CustomerName, Amount, State
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE CustomerName LIKE '%Johnson%';
```

```
-- LIKE: cities starting with 'New'
SELECT DISTINCT City
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE City LIKE 'New%';
```



ORDER BY & LIMIT

LIVE DEMO

- ORDER BY col ASC — lowest to highest (default)
- ORDER BY col DESC — highest to lowest
- ORDER BY multiple columns — first by col1, then col2
- LIMIT n — return only the first n rows
- Together: great for Top-N analysis

SQL execution order (important!)

**FROM → WHERE → SELECT → ORDER BY
→ LIMIT**

```
-- Top 10 highest amount orders
SELECT OrderID, CustomerName, Amount, Category
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
ORDER BY Amount DESC
LIMIT 10;

-- Lowest profit transactions
SELECT OrderID, Profit, Category, State
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
ORDER BY Profit ASC
LIMIT 10;

-- Sort by State, then by Amount within each state
SELECT State, CustomerName, Amount
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
ORDER BY State ASC, Amount DESC
LIMIT 20;
```



Aliases (AS) & Calculated Columns

LIVE DEMO

AS

Rename a column
in results

Arithmetic

+ - * / create
new
columns on the fly

Useful metrics we can calculate:

- Amount / Quantity → price per unit
- Profit / Amount * 100 → profit margin %
- Amount – Profit → estimated cost

```
-- Alias: rename columns for clarity
SELECT
  OrderID                AS order_id,
  Amount                 AS revenue,
  Profit                 AS profit
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
LIMIT 10;

-- Calculated: price per unit
SELECT
  OrderID,
  Category,
  Amount,
  Quantity,
  Amount / Quantity     AS price_per_unit
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
LIMIT 15;

-- Profit margin (%)
SELECT
  OrderID,
  Amount,
  Profit,
  ROUND(Profit / Amount * 100, 1) AS profit_margin_pct
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE Amount > 0
ORDER BY profit_margin_pct DESC
```



ROUND() & Putting It All Together

LIVE DEMO

ROUND(value, decimal_places)

ROUND(3.14159, 2) → 3.14

ROUND(1235.7, 0) → 1236

Real analysis: high-revenue, low-profit rows

- These are orders where revenue looks good but profitability is weak
- Could indicate pricing issues or high cost sub-categories
- Combine WHERE + ORDER BY + calculated columns

```
-- Find high-revenue but low-margin rows
```

```
SELECT
  OrderID,
  Category,
  Sub_Category,
  Amount,
  Profit,
  ROUND(Profit / Amount * 100, 1) AS margin_pct,
  Amount / Quantity AS price_per_unit
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE Amount > 3000
ORDER BY margin_pct ASC
LIMIT 15;
```

```
-- Interesting: large furniture orders with thin margins
```

```
SELECT
  Sub_Category,
  Amount,
  Profit,
  ROUND(Profit / Amount * 100, 1) AS margin_pct
FROM `nth-guide-448810-h6.sql_course.sales_dataset`
WHERE Category = 'Furniture'
  AND Amount > 5000
ORDER BY margin_pct ASC
LIMIT 10;
```

Video 2 — Quick Reference



Retrieve & Filter

```
SELECT col1, col2

SELECT DISTINCT col

FROM `project.dataset.table`

WHERE col = 'value'

WHERE col > 1000

WHERE col IN (...)

WHERE col BETWEEN a AND b

WHERE col LIKE '%text%'

AND / OR
```

Sort & Limit

```
ORDER BY col ASC

ORDER BY col DESC

ORDER BY col1, col2

LIMIT n
```

Execution order:

```
FROM → WHERE

→ SELECT

→ ORDER BY → LIMIT
```

Calculate & Rename

```
col AS alias_name

col1 + col2

col1 - col2

col1 * col2

col1 / col2

ROUND(value, n)
```

e.g.

```
Amount/Quantity AS
price_per_unit
```



Practice Task + What's Next

Try these queries yourself:

- 1 List all unique Sub-Categories
- 2 Find all orders from New York with Amount > 7000
- 3 Find orders where PaymentMode is 'UPI' OR 'Credit Card'
- 4 Find orders where Quantity BETWEEN 10 AND 20
- 5 Top 10 orders by Profit (descending)
- 6 Calculate price_per_unit for Furniture orders;
sort by price_per_unit DESC
- 7 Find orders with margin < 10% and Amount > 5000
(high revenue, low profit)

Next: Video 3

Aggregate Functions & GROUP BY

You'll learn:

- COUNT, SUM, AVG, MIN, MAX
- GROUP BY single and multiple columns
- Total revenue by category
- Profit by state and city
- Building summary reports