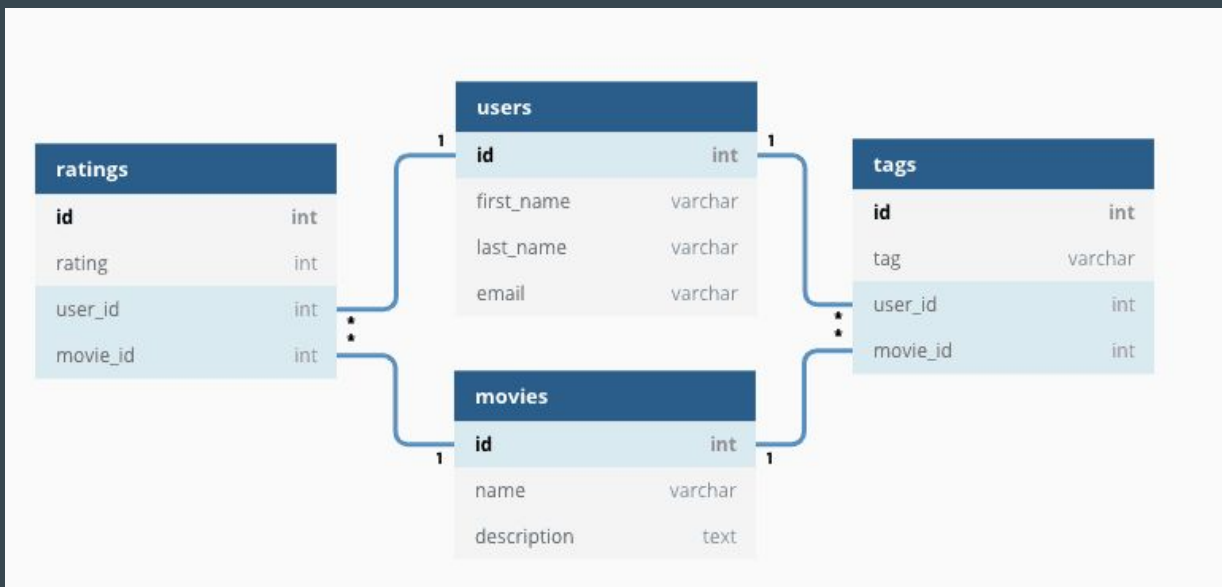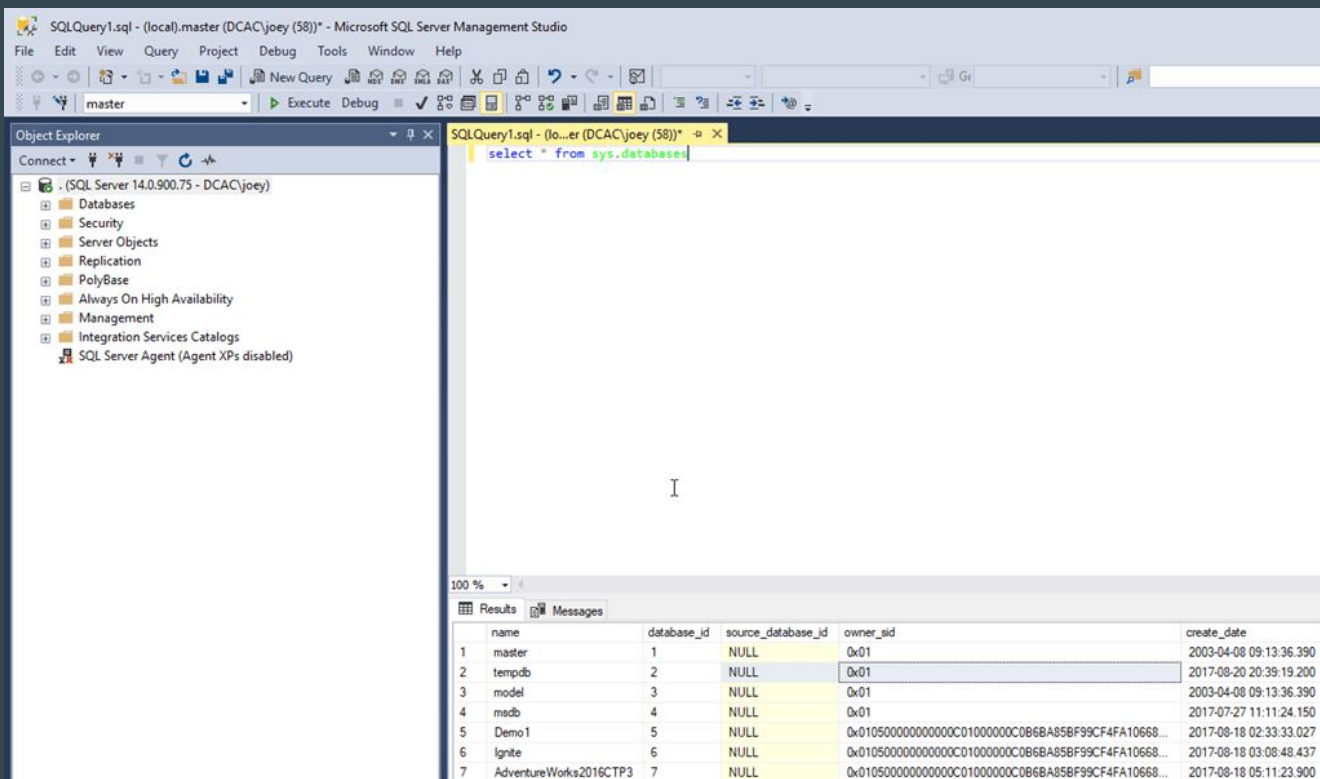# Intro to SQL

●●●

Emily Nomura

What is SQL?

- Relational database management system
- Main purposes:
  - Send, retrieve, and organize information

User Interface

Learning Outcomes

Selecting columns

Aggregate functions

Filtering rows

Sorting and grouping

SELECT Statements

countries			
	name	population	area	unemployment
1	Canada	37,694,085	9,984,670	6.30
2	Chile	18,186,770	756,102	6.70
3	Colombia	49,084,841	1,138,910	9.30
4	Canada	37,487,816	null	5.79

SELECT name, unemployment
FROM countries

	name	unemployment
1	Canada	6.30
2	Chile	6.70
3	Colombia	9.30
4	Canada	5.79

SELECT *
FROM countries

CIA Factbook, Guide to Country Comparisons

SELECT Statements with DISTINCT and COUNT

SELECT DISTINCT name
FROM countries

	name
1	Canada
2	Chile
3	Colombia

SELECT DISTINCT area
FROM countries

	area
1	null
2	9,984,670
3	756,102
4	1,138,910

SELECT COUNT(*)
FROM countries

	count
1	4

SELECT COUNT(area)
FROM countries

	count
1	3

Combining DISTINCT and COUNT

countries			
	name	population	area	unemployment
1	Canada	37,694,085	9,984,670	6.30
2	Chile	18,186,770	756,102	6.70
3	Colombia	49,084,841	1,138,910	9.30
4	Canada	37,487,816	null	5.79

Quiz Q!

SELECT COUNT(DISTINCT area)
FROM countries

Filtering Rows

= equal

<> not equal

< less than

> greater than

<= less than or equal to

>= greater than or equal to

IS NULL

IS NOT NULL

SELECT name, population
FROM countries
WHERE name = 'Canada'

	name	population
1	Canada	37,694,085
2	Canada	37,487,816

SELECT COUNT(*)
FROM countries
WHERE unemployment < 7
AND area > 1,000,000

	count
1	1

SELECT name, area
FROM countries
WHERE area IS NULL

	name	area
1	Canada	null

Filtering Rows with BETWEEN and WHERE IN

SELECT name, area
FROM countries
WHERE area
BETWEEN 1e+6 AND 1e+7

	name	area
1	Canada	9,984,670
2	Colombia	1,138,910

SELECT name, unemployment
FROM countries
WHERE unemployment IN (6.3, 6.7)

	name	unemployment
1	Canada	6.30
2	Chile	6.70

Filtering Rows with LIKE and NOT LIKE

Wildcards:

'%'
- Placeholder for one or many integers or characters

'_'
- Placeholder for a *single* character

SELECT name
FROM countries
WHERE name LIKE 'Ca%'

	name
1	Canada
2	Canada

SELECT name
FROM countries
WHERE name NOT LIKE 'Ca%'

	name
1	Chile
2	Colombia

Aggregate Functions, Arithmetic, Aliasing

- AVG
- SUM
- MAX
- MIN

SELECT AVG(population), AVG(area)
FROM countries
WHERE area IS NOT NULL

	avg	avg
1	34,988,565.30	3,959,894

SELECT MAX(area) - MIN(area) AS difference
FROM countries

	difference
1	9,228,568

SELECT AVG(population) AS avg_pop,
       AVG(area) AS avg_area
FROM countries
WHERE area IS NOT NULL

	avg_pop	avg_area
1	34,988,565.30	3,959,894

Sorting with ORDER BY

SELECT DISTINCT name
FROM countries
ORDER BY name

	name
1	Canada
2	Chile
3	Colombia

SELECT name, unemployment
FROM countries
ORDER BY unemployment DESC

	name	unemployment
1	Colombia	9.30
2	Chile	6.70
3	Canada	6.30
4	Canada	5.79

Grouping with GROUP BY

SELECT name, MIN(unemployment)
FROM countries
GROUP BY name

	name	unemployment
1	Chile	6.70
2	Colombia	9.30
3	Canada	5.79

WHERE AVG(population) > 1,000,000 ❌

HAVING AVG(population) > 1,000,000 ✅

Additional Resources

DataCamp

W3Schools