



NYU | STERN

Introduction to SQL

Sarah Beckett-Hile

E-mail

sbh355@stern.nyu.edu

COURSE DESCRIPTION

Massive and speedy databases are at the core of many businesses. In technology and retail, every page view, click, transaction or even cursor movement needs to be recorded and easily recalled. To handle their data storage needs, the majority of businesses rely on relational database management systems (RDBMS), including Microsoft's SQL Server, Oracle, MySQL and SQLite. In this class, students will learn how to read and synthesize information stored in relational databases by learning the basics of SQL (Structured Query Language). By gaining an understanding of relational databases structure and of SQL, they will also gain the tools to independently expand their skillset as needed in their future careers.

WHY SQL?

SQL vs. Reporting Dashboards

Many companies provide dashboards to allow less tech-savvy employees to access some information from their database. However, dashboard users will always be limited to the predetermined structure of the reports that dashboards can generate. SQL allows its users the freedom to ask whatever question they want. As long as the information is somewhere in the database, SQL will help you find it.

SQL vs. Excel

Anyone familiar with Excel's pivot tables knows that Excel has come a long way in its ability to group, sort, organize, and filter data. However, a pivot table with more than a few hundred thousand rows of data gets sluggish or even crashes. Excel isn't built for the massive amount of data that most companies house in their databases. It also has difficulty with relational tables, although Excel 2013 has made a lot of progress. However, even if Excel could handle unlimited data, you would still need to learn the basics of SQL to pull the necessary information from your company's database before pasting it into Excel. SQL lets you operate directly with databases, saving you a lot of time.

SQL doesn't replace these other tools; most people use SQL in conjunction with standard reporting dashboards and Excel. Dashboards are great for standard monitoring, and Excel is amazing for quick data visualization. SQL lets you investigate. Your workflow might look like this: you check your KPIs (Key Performance Indicators) on a standard dashboard, and some metric looks out of whack. You use SQL to ask more detailed questions and figure out what's going on. Then you copy and paste your results into Excel to make a graph for easy synthesis. Ultimately, SQL heightens the usefulness of these other resources and takes your analysis to the next level.

Covered Concepts:

By the end of this course, students will be able to:

Read Relational Database Structures – examine the overall structure of relational databases and learn simple queries to gain insight into any new database and its tables.

Read and Organize Data Sets – use the commands that you’ll find at the core of the simplest and most complex queries to pull, filter, and organize information from a specific table.

Join Tables - leverage the core strength of relational databases and their ability to link data from multiple tables.

Apply Functions – sum, average, find the largest or smallest item; there are countless functions in SQL to help you find the information you’re seeking.

Group Data – create groups to compare metrics and gain deeper insights into your data.

Apply Boolean Logic – use if/then statements to reshape data and find more specific answers.

Nest and Unite Multiple Queries – combine queries to better synthesize data.

Optimize Queries – ensure queries run quickly and efficiently to ensure speedy data return and minimal stress on your servers (your system administrators will be grateful).

Create and Update Tables – create simple new tables and update existing ones.

Course Methods

Lessons are designed to both illustrate the utility of the concepts that we will cover, and to allow students to create and modify their own queries. The class will use iPython as a shell for learning SQL as an alternative to other relational database reading programs, but they will not be required to read or write in Python. Rather, iPython will work as an interactive playground where lessons, example queries, and opportunities to practice are all united in a single location. We will be using a SQLite database to avoid the need for a central server, but we will note when there are differences between SQL in SQLite and SQL in other RDBMS options like Oracle, MySQL and SQL Server (the differences are minimal). Students will access this SQLite database to run queries regarding a fictional company and its transaction information. Lessons are aimed at illustrating how SQL can be leveraged to gain analytical insights into a company’s business practices.

Required Materials. Download and install Anaconda prior to class. Anaconda will let you launch and run iPython, and it comes with the packages necessary to make the lesson code function. Instructions on how to download Anaconda can be found in this link, be sure to follow them carefully so you download the correct version:

<http://pages.stern.nyu.edu/~dbackus/bootcamp/python0.html> - python-on-your-computer

Additionally, you will need to download the two files below. It will not matter where you save them, so long as you save both of them in the same folder.

https://www.dropbox.com/s/hz7w3cmzk0zbtfr/SQL_Intro.ipynb?dl=0

https://www.dropbox.com/s/5khcl9utqzdox9v/SQL_support_code.py?dl=0

Instructor

Sarah Beckett-Hile is an MBA1 at New York University Stern School of Business specializing in Strategy and Business Analytics. Prior to enrolling at Stern, she spent five years at Peanut Labs, a technology startup that produces a digital platform to monetize the websites of partnering online game publishers and virtual communities. In each of the roles that Ms. Beckett-Hile held while at Peanut Labs (Product Manager, Director of Accounts and Director of Business Intelligence), she leveraged SQL to gain insights from her company's database and create data-driven strategies for product development, client relationship management, and process optimization. Ms. Beckett-Hile also codes in Python. She holds a B.A. in English Literature from Washington University in St. Louis.

Executive Producer

Glenn A. Okun is a professor of finance and management at New York University Stern School of Business where he teaches courses in entrepreneurship, private equity, venture capital, investment management and other finance topics. Mr. Okun advises corporations on financial and investment matters. He was President of Mitchum, Jones & Templeton, a merchant bank and broker dealer headquartered in San Francisco, California from 1998 to 2001. He previously served as a Director of Allen & Company Incorporated in New York. Mr. Okun invested in early and later stage financings of private companies in various industries. He also ran a small cap emerging growth stock hedge fund and a special situations portfolio. Mr. Okun has advised corporate clients on mergers, acquisitions and restructurings and has underwritten public offerings and private placements of securities. Mr. Okun began his investment career at the IBM Retirement Fund where he invested in mezzanine private placements, real estate, public emerging growth equities and oil and gas assets. Mr. Okun holds JD and MBA degrees from the joint law and business program of Harvard University and a BA degree from Wesleyan University. At Stern he has been awarded numerous teaching awards, including Professor of the Year.

SCHEDULE

Introduction to Relational Databases

Session #1 Why SQL?

Topics: Overview of Relational Databases and the most popular management systems; terminology; SQL structure, idiosyncrasies, and examples; basic queries to view tables

Terms: PRAGMA TABLE_INFO
 SELECT
 *
 DISTINCT
 FROM

The Core Clauses

Session #2 Pulling, filtering, and organizing data sets

Topics: Selecting specific columns; filtering, ordering, and limiting data in the result-set

Terms: WHERE
 LIKE
 ORDER BY
 LIMIT

Combining Tables and Columns

Session #3 Aliases, Joining Tables, & Simple Operators

Topics: Using aliases to shorten queries and simplify joins; joining tables to combine data from both to gain better insights; overview of the different types of table joins and the merits/detriments of each; using simple operators to add, subtract, multiple, divide and concatenate values in columns

Terms: AS
 INNER JOIN
 LEFT JOIN
 OUTER JOIN
 + - * / |

Gaining Analytical Insights with SQL

Session #4 Functions and Groups

Topics: Using functions to get more insight from your queries; creating groups to make the most out of functions and compress your result-sets into usable segments; rules for using “having” vs. “where” to filter data.

Terms: SUM
COUNT
MIN
MAX
AVG
ROUND
CONCAT
GROUP BY
HAVING
ROLLUP

Conditional Expressions and Combining Queries

Session #5 Using conditional logic; Nesting and Uniting

Topics: Using queries within queries (nesting) and multiple queries in a list (uniting) in order to produce more specific or more comprehensive result-sets.

Terms: IF
CASE WHEN
“nesting” (no specific term)
UNION

Optimization and Create/Update/Delete Basics

Session #6 Avoiding the wrath of your system administrator

Topics: Ensuring queries are written for maximum efficiency to avoid unnecessary stress to your system’s servers; basics of creating and updating tables.

Terms: CREATE
INSERT INTO
UPDATE
DROP