

# SQL Server Bootcamp

Learn to extract information from databases by writing SQL queries, joining tables, aggregating data, and filtering results so you can turn data into actionable insights.

Group classes in NYC and onsite training is available for this course. For more information, email [hello@nobledesktop.com](mailto:hello@nobledesktop.com) or visit: <https://www.nobledesktop.com/classes/sql-server-bootcamp>



[hello@nobledesktop.com](mailto:hello@nobledesktop.com) • [\(212\) 226-4149](tel:2122264149)

## Course Outline

This package includes these courses

- SQL Level 1 (8 Hours)
- SQL Level 2 (8 Hours)
- SQL Level 3 (8 Hours)

## Foundations of SQL & Databases

### SQL Fundamental Concepts

- What is SQL & why is it used?
- Flavors of SQL: SQL Server vs Postgres vs MySQL
- Database Tables, Rows, & Columns

### Exploring Databases & Writing SQL Statements (with the free SQL Server Management Studio app)

- Connecting to a Database (and Creating our Training Database using a provided SQL script)
- Object Explorer
- SQL Query Editor
- Using Code Hints
- Viewing the Results of your SQL query
- Setting Preferences (Options)

## Writing SQL Queries

### Writing SELECT Statements

- Syntax of a SELECT statement
- Selecting all columns or specific columns from a table
- Limiting the number of results using TOP
- Ordering the results using ORDER BY
- Returning only DISTINCT records (eliminating duplicates)

## Filtering Results

- Data Types (Strings vs Numbers)
- Comparison Operators: equal to, greater or less than, not equal to, etc.
- Filtering results using WHERE, AND, OR, IN, and NOT
- Pattern Matching: Wildcard Filters
- Case Sensitivity

## Using Joins to Combine Data from Multiple Tables

### Understanding Table Relationships

- What are Primary vs Primary Keys
- Creating ER (Entity Relationship) Diagrams to visual what's in a database
- Database Relations: One-to-One, One-to-Many, & Many-to-Many

### Inner Joins

- The difference between Inner & Outer Joins
- Inner Joins
- Column & Table Aliases

### Outer Joins

- Left Join
- Right Join
- Full Join

## Manipulating, Aggregating, & Filtering Data

### Using CAST to Change Data Types

Why and how to use CAST to make a data type fit your query's needs

### Aggregate Functions

- Using Aggregate Functions to perform common statistical calculations
- Using SUM, COUNT, AVG, MAX & MIN

### Working with Dates & Time

- Date Functions: Getting the desired part of a date/time (Year, Month, Day, etc.)
- Formatting dates, including the day of the week (Sunday, Monday, etc.)
- Calculating the difference between 2 dates

## Grouping Data & Filtering Grouped Data

### Grouping Data with Aggregate Functions

- Using GROUP BY to organize results into categories
- Applying Aggregate Functions to groups

### Filtering Grouped Data with HAVING

- Using HAVING to filter the results of a GROUP BY
- HAVING vs WHERE

## Other Types of Joins: Self Joins & Union

### Self Joins

- How a Self Join is different from other types of joins
- Using a Self Join

### Union

- How UNION is different from other types of joins
- Combining result sets with UNION

## Advanced Query Techniques

### Conditionals

- Adding If-Else Logic with CASE
- Simple CASE vs Searched CASE

### String Functions

- Manipulating text using string functions
- Using SUBSTRING, LEFT/RIGHT, CHARINDEX, UPPER, etc.
- Combining with Inline IF statements (IIF)

### Subqueries

- Subqueries: Queries within queries
- Single-Value vs Multi-Value Subqueries

## Views

### Creating & Using Views

- What are Views?
- The Advantages & Disadvantages of using Views
- Creating & Deleting a View
- Querying a View

## Variables & User-Defined Functions

### Variables

- What are variables and why would you use them?
- Creating & Using Variables

### User-Defined Functions

- Creating User-Defined Functions
- Using Variables in User-Defined Functions

## Stored Procedures

### Creating & Using Stored Procedures

- The difference between Views, User-Defined Functions, and Stored Procedures
- Creating & Using Stored Procedures