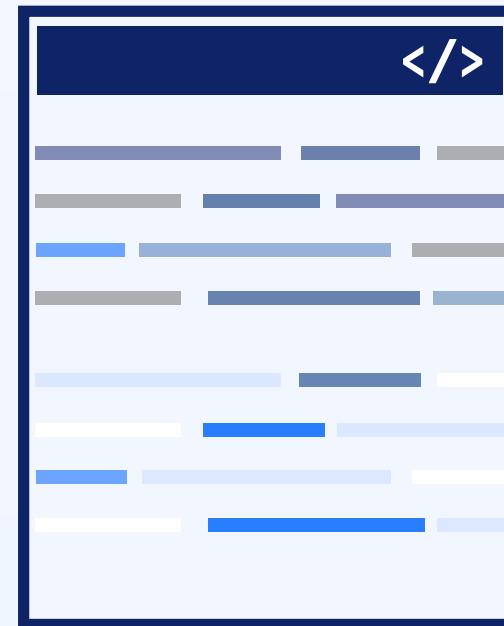
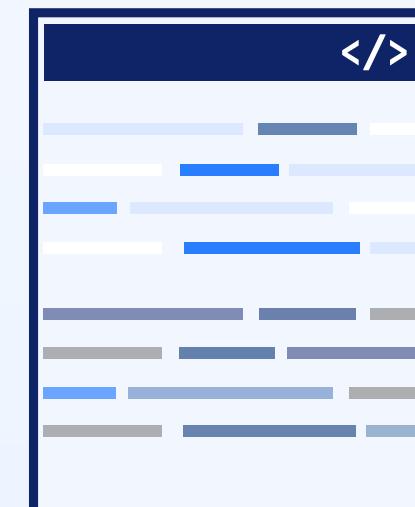


LEARN SQL

for Data Engineering



SQL



In Just 10 Days



Disclaimer

This 10-day guide is structured to build your SQL skills from the ground up, empowering you with the knowledge to query and analyze data efficiently.

Remember, practice is key to mastering SQL, so continue to challenge yourself with new queries and datasets beyond this guide.



DAY 1

SQL Basics

→ Overview of SQL

Understand what SQL is and why it's essential for Data Engineering.

→ Setting Up Your Environment

Install SQL Server or any SQL database system and set up your first database.

→ Basic Syntax

Learn SELECT statements, FROM clause, and how to execute simple queries.

Practice Question

- Write a query to select all columns from a **customers** table.
- Write a query to select the **name** and **email** columns from a **users** table



DAY 2

Filtering, Sorting, and Operators

→ Filtering Data

Use the WHERE clause to filter datasets based on specific criteria.

→ Sorting Results

Learn to use ORDER BY to sort your query results.

→ Operators

Dive into comparison and logical operators to refine your data queries.

Practice Question

- Write a query to select all records from an `orders` table where the `amount` is greater than 100 and sort the results by the `order_date` in descending order.
- Select all `employees` who work in the 'Marketing' department and have been employed after January 1st, 2020. Order the results by `employee_name`.



DAY 3

Functions and Aggregations

→ SQL Functions

Explore built-in SQL functions like COUNT(), SUM(), AVG(), MIN(), and MAX().

→ Grouping Data

Use GROUP BY to aggregate data and perform calculations on groups.

→ HAVING Clause

Learn to filter aggregated data using the HAVING clause.

Practice Question

- Find the average salary and total number of **employees** in the employees table.
- Select the department and the highest salary in each department from the **employees** table.



DAY 4

Joining Tables

→ Understanding Joins

Introduction to INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL OUTER JOIN.

→ Querying Multiple Tables

Practice writing queries that pull data from multiple tables simultaneously.

→ Aliasing

Use aliases to simplify your queries and improve readability.

Practice Question

- Write a query to join **orders** table with **customers** table on the customer ID and select the order ID, order date, and customer name for all orders.
- Find all employees and their department names. Assume you have **employees** and **departments** tables.



DAY 5

Subqueries and Nested Queries

→ Introduction to Subqueries

Learn to write nested queries for complex data extraction.

→ Using Subqueries

Practice using subqueries in the SELECT, FROM, and WHERE clauses.

→ Correlated Subqueries

Understand and apply correlated subqueries to perform more advanced analysis.

Practice Question

- Select all products that have a price above the average price of all products. Use a subquery to determine the average price.
- Write a query to find the names of employees who have made more than 10 sales. Use a subquery to calculate the number of sales per employee.



DAY 6

Set Operations

→ Set Theory Basics

Quick overview of set theory principles relevant to SQL.

→ Using UNION, INTERSECT, and EXCEPT

Learn to combine, intersect, and exclude datasets using set operations.

Practice Question

- Use UNION to combine the result sets of two queries: one that selects all Python developers from an `employees` table and another that selects all Data Scientists.
- Find employees who are both Python developers and Data Scientists using INTERSECT.



DAY 7

Working with Dates and Times

→ Date and Time Functions

Explore how to manipulate and query date and time data.

→ Formatting Dates

Learn to format date and time values for better readability and analysis.

Practice Question

- Select all orders made in the last 30 days from the [orders](#) table.
- Write a query to find the number of orders made each month. Format the month as "YYYY-MM".



DAY 8

Advanced SQL Features

→ Window Functions

Introduction to OVER(), PARTITION BY, and windowing functions for advanced data analysis.

→ CTEs (Common Table Expressions)

Learn how to use WITH clause for complex queries and improving query organization.

Practice Question

- Use a window function to rank employees in each department by their salaries.
- Write a query with a CTE that selects all employees who have a salary above the average salary in their department.



DAY 9

Indexes and Performance Tuning

→ Understanding Indexes

Learn what indexes are and how they improve query performance.

→ Creating Indexes

Practice creating and managing indexes on your tables.

→ Query Optimization

Tips and tricks for writing efficient SQL queries.

Practice Question

- Assuming an `orders` table with columns `order_id`, `customer_id`, and `order_date`, write a query to create an index on `order_date`. (Note: Just the concept, syntax might vary based on the SQL database.)
- Write a query to explain the execution plan of a select statement that joins the `employees` table with the `departments` table. (Note: Use your SQL database's specific method to show the execution plan.)



DAY 10

Real-World SQL Project

→ Project Overview

Apply what you've learned on a real-world dataset.

→ Data Exploration and Analysis

Perform comprehensive data exploration and analysis using advanced SQL queries.

→ Insights and Reporting

Extract meaningful insights and prepare a report on your findings.

Practice Question

- Given a dataset of eCommerce transactions ([transactions](#) table), write a query to find the top 3 most purchased products.
- Using the [customers](#) table, write a query to find the month-over-month growth rate in new customer sign-ups. Compare the counts of new sign-ups each month to the previous month.



Resources and Practice:

→ Hands-on Practice

Use online platforms like LeetCode, HackerRank, or SQLZoo to practice your SQL skills daily.

→ Sample Datasets

Experiment with different datasets available on platforms like Kaggle to explore various data scenarios.

→ Documentation and Reading

Refer to the official documentation of the SQL version you're using and supplement your learning with books or online courses tailored to Data Engineering.





WHY BOSSCODER?

➤ **1000+** Alumni placed at Top Product-based companies.

➤ More than **136% hike** for every **2 out of 3** working professional.

➤ Average package of **24LPA**.

The syllabus is most up-to-date and the list of problems provided covers all important topics.

Lavanya
 Meta



Course is very well structured and streamlined to crack any MAANG company

Rahul .
 Google



[EXPLORE MORE](#)