Introduction

The introductory lesson explains what folks can expect from this course and who the intended audience is.

Introduction

Throughout history, humans have tried to store information in one form or another. The Egyptians inscribed hieroglyphs on tomb walls, papyrus, and wood as long ago as 3200 B.C. to preserve their message for posterity. In recent centuries, paper has been used as a medium to store information. Fast forward to the 20th century, electronic storage and retrieval of information have become commonplace.

The relational database and its cousin, the Structured Query Language, have been empowering applications and offerings from popular internet companies since the beginning of the internet revolution. In fact, the most popular online developer destination, Stack Overflow, runs a SQL server. The advent of Big Data and NoSQL technologies hasn't put Relational Databases or SQL out of fashion. These technologies serve different and complementary use cases, albeit there is an overlap in the capabilities of both. Learning and mastering SQL can be a great asset when foraying into a tech career as any meaningful application will have a relational database running under the hood, and SQL is the de facto standard to

interact with it. Familiarity with SQL pays rich dividends even in the Big

Data realm, as several software systems such as Hive and Phoenix expose SQL-like interfaces and syntax to end-users.

Intended audience

We take a fresh look on learning the ubiquitous structured query language by providing users with an in-browser SQL prompt. Our choice of database is the wildly popular open-source, MySQL. The intended audience of this course is non-tech-savvy professionals who may occasionally use SQL for analytics and reporting. Additionally, folks who are beginners or struggling with SQL can also benefit from the course.

What to expect?

The course is structured to provide a brief and cursory overview of database theory and the related industry jargon. The course intentionally doesn't dive deep into database theory and instead focuses more on hands-on practical learning. For folks interested in theoretical learning, we recommend checking out the Database Design Fundamentals course by Educative. Later, we dive right into learning SQL with examples. Each lesson comes with an interactive shell so that the reader can fire up and try out the lesson queries at the prompt. The data comes pre-filled, just as most readers would experience in actual work settings. Finally, we wrap up the course with rigorous quizzes and common interview questions. The goal is to level up the SQL skills of the reader to an intermediate level and enable the reader to hop through beginner/intermediate-level SQL job interviews easily.