Exploring the prevalence of SQL misconceptions: a study design
Miedema, D.; Aivaloglou, E.; Fletcher, G.; Seppälä, O.; Petersen, A.

Citation

Version: Publisher's Version
License: Licensed under Article 25fa Copyright Act/Law (Amendment Taverne)
Downloaded from: https://hdl.handle.net/1887/3250268

Note: To cite this publication please use the final published version (if applicable).
ABSTRACT
The Structured Query Language (SQL) is an established language for data manipulation in relational databases. It is widely used in industry, and therefore part of the typical Computer Science curriculum. From the large amounts of mistakes higher education students make while learning and using SQL, we know that this language is not easy to learn. Various researchers have examined the types of mistakes SQL novices make, and recently, the first step towards understanding the underlying reasons for these mistakes has been made. In this poster abstract, we propose a study to examine the prevalence of these origins, also called misconceptions. We hope the Computer Science Education community will help us reflect on and strengthen our methodology, and ultimately, our findings.

CCS CONCEPTS
• Social and professional topics → Computing education; • Information systems → Structured Query Language.

KEYWORDS
SQL, misconceptions, multiple-choice questions, study design

1 INTRODUCTION
Over the past decades, many researchers have focused on the difficulties that the Structured Query Language (SQL) poses to students. Most of these researchers examined error types. This started in the seventies, with Reisner’s insights on both minor impact errors (misspellings, synonyms, punctuation errors), and major impact errors that produce a -by definition incorrect- result, such as an empty result table, because of a contradictory WHERE clause [3]. Other recent research calculates query formulation success rates [4, 9], or creates more specific lists of errors of a certain type: syntactic [1], semantic [2] or both [11, 14]. Another newly introduced category is that of complications: queries that give the correct result but contain unnecessary elements [16]. Of these four categories, Taipalus and Perälä [14] found that logical errors and complications are more likely to persist throughout query formulation than syntax and semantic errors.

Student performance is most often measured through error frequency. These errors have underlying causes, which we call misconceptions. In recent work, Miedema, Aivaloglou and Fletcher introduce misconceptions that cause various SQL errors [8]. The paper explores SQL misconceptions that students have, as distilled from interviews. This knowledge is another step towards improving SQL education, as the paper gives insights into where the students’ mental models took a wrong turn. To address the reduction of such misconceptions, it is useful to identify which misconceptions are the most prevalent. Interventions based on these more prevalent misconceptions should lead to a large improvement on student performance.

With the study design described below, we aim to examine the prevalence of SQL misconceptions across a large student population. We propose to do this by offering students an optional, formative assessment in the shape of a multiple-choice test. We check whether a misconception is held consistently by means of multiple questions. The assessment is provided to the students before the final exam, such that they can use it as preparatory material. The formative setup means that participation in the study is valuable for both the students and the researchers.

The goal of this poster is twofold. First, we aim to discuss our study design with the Koli Calling community in order to strengthen it. Second, we intend to build a community of SQL teachers interested in researching and improving SQL Education.

2 BACKGROUND
Research suggests that to some extent, we can measure students’ programming knowledge with multiple-choice questions. Kuechler and Simkin reported that multiple-choice questions have many advantages in practice, for example, they are easy to score, easily capture a large amount of course material, and they are perceived as more objective [5]. Furthermore, in the case of programming material, performance on multiple-choice questions correlates with performance on open questions (although not all variance can be
explained by this correlation) [5]. Kleerekoper and Schofield find that formative assessment in the shape of practice tests can improve students’ score on the test [4], providing motivation for the students to participate in our study.

Evaluating misconceptions through (two- or three-tier) multiple-choice instruments is a common approach in literature, applied as early as 1986 [18]. It has been applied for misconceptions in a wide range of topics such as: chemistry [18], biology [17], mathematics [6], scientific literacy [19].

More specific to Computer Science Education, most studies that use multiple-choice questions to diagnose misconceptions are on the topic of programming. Swidan, Hermans and Smit looked at common misconceptions in Scratch users, and find evidence of Scratch-induced misconceptions [13]. Žanko, Mladenović and Boljat examined the presence of misconceptions on the topic of variables for K-12 students, they found no difference in misconceptions when comparing students learning Python versus Logo [24]. Mladenović, Boljat and Žanko considered misconceptions regarding loops held by K-12 students using different programming languages, and found that students learning Scratch held less misconceptions regarding loops, than students learning text-based languages (Python, Logo) [10]. Ma, Ferguson, Roper and Wood applied multiple-choice questions to see whether students held consistent conceptions of value assignment and reference assignment, and found that value assignment was typically done correctly, but reference assignment was not [7]. Wittie, Kurdia and Huggard looked at distractors (incorrect MCQ answers that indicate misconceptions) and then designed two Concept Inventory questions on parameters [23].

As SQL is a query language, and thus closely related to programming languages, we adopt the approach of the aforementioned researchers to find the prevalence of misconceptions. As a starting point, we use the misconceptions identified by Miedema et al. [8].

3 PROPOSED RESEARCH METHOD

We propose to set up a multiple-choice questionnaire that functions as a formative assessment that students can take to check their SQL knowledge. The questionnaire should examine the prevalence of previously identified misconceptions, and find whether they are widely held. We can support the students in their studying by providing explanations why a given answer is incorrect, what the correct answer is, and why.

Participants. To reduce the effect of teacher approach on student knowledge, we aim to recruit participants at various institutions across the globe, by calling on the authors’ networks.

A student’s teacher affects the way in which they learn in many ways, including the order of the material, the type of examples, and the database schema can all influence the student’s mental models. We thus require a diverse set of students with different teachers to gather representative results.

Materials. The main material is a three-tier multiple-choice questionnaire. Three-tier means that each question contains three elements: (1) a multiple-choice question concerning one misconception, (2) a text-field for the student to elaborate on why they chose this option, and (3) a Likert-scale to indicate how certain they are about their answer. The multiple-choice questions will give us data on whether students chose a distractor answer and may thus hold the corresponding misconception. The textfield may give us further insight into the details of the misconception. Finally, the certainty score indicates whether the error is due to a misconception, or merely an incorrect guess.

For the content of the questionnaire, we consider multiple-choice questions of various forms: we may ask our students to fill in the blank, indicate the correct result table, indicate the correct full query, an MCQ variant of Explain in plain English [20], and other options.

One open question regarding our study design is how to measure the validity and reliability of the questionnaire.

Analysis. Our main analysis will center around three aspects:

1. Misconception prevalence. Which misconceptions are the most prevalent? How often is each misconception held? Are there significant differences in prevalence between students from different institutions?
2. Misconception consistency. Are the participants consistent with regard to individual misconceptions? Do they always make the same type of mistake, or does this depend on the (type of) question?
3. Misconception interaction. Are there confounding misconceptions?

4 IMPLICATIONS

Once we have the answers to the questions mentioned above, we can make more informed decisions on how to address such misconceptions. The design of appropriate interventions depends on which misconceptions lead to the most frequent and biggest problems. Moreover, this information can support interesting research directions as identified by Taipalus and Seppänen [15].

Additionally, the longer a misconception is held, the more established it becomes in memory. Counterexamples for each misconception are important for weakening them. If we find that students don’t consistently hold certain misconceptions, the questions they answer correctly hint to appropriate counterexamples.

Taipalus and Seppänen mapped the literature on SQL education more widely [15]. Besides errors, some other types of research in the area of SQL education that they distinguish include: database types and complexity, teaching approaches, teacher workload, and supporting students in query formulation through visualizations [15]. However, as SQL Education is about the students and their learning process, the evaluation of interventions in such papers often returns to student performance in the end.

REFERENCES

Exploring the Prevalence of SQL Misconceptions: a Study Design

Koli Calling ’21, November 18–21, 2021, Joensuu, Finland


