What to Expect and What to Focus on in SQL Query Teaching

Toni Taipalus
University of Jyvaskyla
Jyvaskyla, Finland
toni.taipalus@jyu.fi

Piia Perälä
University of Jyvaskyla
Jyvaskyla, Finland
piia.m.h.perala@jyu.fi

ABSTRACT

In the process of learning a new computer language, writing erroneous statements is part of the learning experience. However, some errors persist throughout the query writing process and are never corrected. Structured Query Language (SQL) consists of a number of different concepts such as expressions, joins, grouping and ordering, all of which by nature invite different possible errors in the query writing process. Furthermore, some of these errors are relatively easy for a student to fix when compared to others. Using a data set from three student cohorts with the total of 744 students, we set out to explore which types of errors are persistent, i.e., more likely to be left uncorrected by the students. Additionally, based on the results, we contemplate which types of errors different query concepts seem to invite. The results show that syntax and semantic errors are less likely to persist than logical errors and complications. We expect that the results will help us understand which kind of errors students struggle with, and e.g., help teachers generate or choose more appropriate data for students to use when learning SQL.

CCS CONCEPTS

• **Social and professional topics → Computing education**; *Computational thinking*; *Student assessment*;

KEYWORDS

SQL, error, query language, database education, relational database

ACM Reference Format:
Toni Taipalus and Piia Perälä. 2019. What to Expect and What to Focus on in SQL Query Teaching. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), February 27-March 2, 2019, Minneapolis, MN, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3287324.3287359

1 INTRODUCTION

SQL has been taught at university level courses for decades, yet compared to programming languages, SQL has received relatively little attention in educational research. A number of new teaching methods have been presented to facilitate learning [5, 7–9], but scientific evidence on which parts of SQL students struggle with leaves room for interpretation.

This study is an attempt to address the issue of difficult, i.e., persistent errors in SQL. We consider an error persistent if it is present in a student's final answer to an exercise, i.e., the student did not fix the error during the query writing process. The aim of our study is to find out which errors are persistent, and then consider which query concepts invite which persistent errors. To that end, we analyzed the final answers of three student cohorts with the total of 744 students and over 8,700 SQL queries.

In Section 2, we discuss previous studies on SQL error categorization, and briefly explain the frameworks we chose for this study. In Section 3, we describe how we collected and analyzed the data, and in Section 4 we report our findings. In Section 5 we compare our findings to previous studies, and consider the practical implications and limitations of our research, as well as further research opportunities. Finally, in Section 6, we present conclusions.

2 BACKGROUND

2.1 Related Work

Previous SQL research in educational contexts mainly focuses on one of two perspectives. First, on the development and analysis of a particular tool for facilitating SQL learning, and second, on the study of student errors in SQL. The former class of SQL research is out of the scope of our study, but the latter class presents a number of error categorizations, which are needed to quantifiably measure and analyze student errors.

Brass and Goldberg [4] presented an extensive list of semantic errors to be used in database management system (DBMS) compilers, and a set of studies [1–3], which inspired us to this research, explored the frequencies of syntax and semantic errors students made when learning SQL. Ahadi et al. [1] used PostgreSQL to categorize syntax errors from over 160,000 SQL queries. Ahadi et al. [2] studied student errors in exercises with seven different query concepts, and, as the authors point out, theirs is the first published quantitative study of the relative student difficulties in regard to different SQL query concepts. Additionally, Ahadi et al. [3] closely investigated 551 queries that contained a semantic error which students were unable to correct.

Taipalus et al. [11] composed an unified error categorization and a framework of query concepts using earlier research [1, 2, 4, 10, 12]. These findings were validated and complemented by an analysis of over 33,000 SQL queries, and the categorization was based on the SQL standard, and is DBMS independent. The categorization mapped 105 errors and complications into four classes; 1) complications, which do not affect the result table; 2) logical errors, which affect the result table, and for which there exists a valid data demand (i.e., a natural language representation of what the query

SIGCSE '19, February 27-March 2, 2019, Minneapolis, MN, USA

ACM ISBN 978-1-4503-5890-3/19/02…$15.00
https://doi.org/10.1145/3287324.3287359

needs to return from the database); 3) semantic errors, which affect the result table, and for which there exists no valid data demand; and 4) syntax errors, which result in an error message instead of a result table.

2.2 Errors and Complications

We use the error categorization, and the query concept framework composed by Taipalus et al. [11], because both of them are wider than any of those presented in the previous studies. According to the categorization, there are three classes of errors in addition to complications. The 38 syntax errors are categorized into six categories; ambiguous database object (SYN-1), undefined database object (SYN-2), data type mismatch (SYN-3), illegal aggregate function placement (SYN-4), illegal or insufficient grouping (SYN-5), and common syntax error (SYN-6).

The 13 semantic errors are categorized into five categories; inconsistent expression (SEM-1), inconsistent join (SEM-2), missing join (SEM-3), duplicate rows (SEM-4), and redundant column output (SEM-5).

The 30 logical errors are categorized into six categories; operator error (LOG-1), join error (LOG-2), nesting error (LOG-3), expression error (LOG-4), projection error (LOG-5), and function error (LOG-6). The categorization points out that, while a semantic error is evident just by reading the query without knowing the data demand, a logical error can be identified only if the data demand is known.

Finally, in addition to errors, the framework contains 24 complications, which are, e.g., unnecessary joins and other structural problems which could be formulated in a simpler fashion. For brevity, we refer to all of these four classes simply as *errors*, when it is not necessary to differentiate errors from complications.

2.3 Query Concepts

As the base for our exercises, we use the the framework [11] which lists 18 query concepts, all of which are present at least once in the 15 exercises. Notably, all of the exercises test skill with more than one query concept, and, what's more, some query concepts invite others by design. All the query concepts per exercise are presented in Table 1.

While some of the concepts are basic, e.g., single-table queries, expressions, aggregate functions, and ordering, others, e.g., multiple source tables (i.e., data in the result table is projected or calculated from more than one source table), and parameter distinct (i.e., DISTINCT is required as an aggregate function parameter) are relatively difficult for an introductory database course. See to Taipalus et al. [11] for more detailed information about the query concepts, exercises, and error categories, as well as example data demands and queries for each of the 15 exercises.

Table 1: Query Concepts per Exercise [11]

Exercise	Concepts
A1	single-table; expressions
A2	single-table; expressions; ordering
A3	single-table; wildcard; expressions with nesting
B4	multi-table; expressions; facing foreign keys
B5	multi-table; expressions with nesting; ordering
B6	multi-table; expressions; does not exist
B7	multi-table; expressions; does not exist
B8	multi-table; expressions; does not exist; equal subqueries
B9	single-table; expressions; aggregate functions
B10	multi-table; expressions; multiple source tables
B11	multi-table; expressions; self-join; aggregate function evaluated against a column value; correlated subquery
B12	multi-table; expressions; aggregate function evaluated against a constant; uncorrelated subquery; parameter distinct
B13	multi-table; expressions; self-join
C14	multi-table; multiple source tables; aggregate functions; grouping
C15	multi-table; multiple source tables; aggregate functions; grouping; grouping restrictions; ordering

3 METHOD

For this study, we had over 123,000 SQL queries which we collected from three student cohorts in a mandatory database course. The course consisted of lectures, voluntary exercises, exercise discussion sessions, and an exam. The students majored in computer science or information systems with no prior knowledge on using SQL. Each cohort answered to 15 SQL retrieval exercises. We designed the exercises using the query concept framework presented by Taipalus et al. [11], with the same query concepts, as well as the number of source and subject tables. In order to mitigate the polarization effect of the database business domain on the number of errors, we designed different database structures and business domains for each cohort.

We also replicated the learning environment as presented by Taipalus et al. [11]. The students answered the 15 exercises during the course. The exercises were divided into three sets (A, B, and C in Table 1). For each set, the students had approximately one week to complete the exercises in the given set, in whatever order, and with unlimited tries. The learning environment was minimally controlled, and the students could use whatever material or forms of communication at their disposal, which more accurately mimics their future work environments, as argued for by Taipalus et al. [11]. The exercise discussion sessions were held after the weekly deadline had passed, and the sessions had no impact on the previous week's queries. The learning environment was effectively an interactive SQL prompt (SQLite) embedded into a web page. The correct result table was visible during the whole query writing process, which, in turn, constitutes as making the environment unnatural when compared to work environments.

In order to pinpoint persistent errors, we analyzed only final answers from each student, which left us with over 8,700 queries for further analysis. Out of these final queries, 2,765 were incorrect. We manually marked these errors according to the error categorization by Taipalus et al. [11]. Based on this, we approached error persistence from two perspectives; relative frequencies for each error category per exercise, and estimated means for each error class per exercise.

Table 2: Relative Error Frequencies by Error Category in Incorrect Final Answers, and Complications in All Final Answers

	A1	A2	A3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	C14	C15
Correct	0.67	0.77	0.60	0.79	0.77	0.78	0.76	0.82	0.91	0.63	0.41	0.43	0.62	0.75	0.56
SYN-1 ambiguous database object	0.00	0.00	0.04	0.02	0.09	0.01	0.03	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00
SYN-2 undefined database object	0.00	0.05	0.00	0.05	0.01	0.03	0.01	0.00	0.00	0.07	0.03	0.03	0.05	0.02	0.02
SYN-3 data type mismatch	0.07	0.00	0.32	0.09	0.23	0.11	0.03	0.03	0.00	0.02	0.01	0.01	0.09	0.06	0.00
SYN-4 ill. aggr. function placement	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.00	0.02
SYN-5 ill. or insufficient grouping	0.03	0.00	0.00	0.01	0.00	0.04	0.00	0.03	0.02	0.06	0.38	0.43	0.06	0.37	0.67
SYN-6 common syntax error	0.11	0.27	0.09	0.25	0.15	0.09	0.16	0.22	0.05	0.14	0.04	0.09	0.10	0.11	0.06
SEM-1 inconsistent expression	0.08	0.00	0.04	0.00	0.01	0.01	0.01	0.14	0.05	0.02	0.01	0.13	0.15	0.04	0.03
SEM-2 inconsistent join	0.00	0.00	0.00	0.04	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.01	0.02	0.01	0.00
SEM-3 missing join	0.00	0.00	0.00	0.01	0.05	0.06	0.05	0.04	0.00	0.34	0.09	0.03	0.08	0.13	0.03
SEM-4 duplicate rows	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.20	0.05	0.00	0.00	0.00	0.00
SEM-5 redundant column output	0.00	0.01	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
LOG-1 operator error	0.00	0.06	0.12	0.03	0.44	0.11	0.65	0.47	0.22	0.05	0.04	0.06	0.07	0.00	0.02
LOG-2 join error	0.00	0.00	0.00	0.34	0.15	0.23	0.19	0.35	0.00	0.40	0.07	0.20	0.39	0.06	0.02
LOG-3 nesting error	0.00	0.00	0.47	0.00	0.01	0.01	0.00	0.09	0.00	0.00	0.00	0.00	0.08	0.00	0.00
LOG-4 expression error	0.13	0.27	0.11	0.47	0.09	0.14	0.20	0.13	0.09	0.25	0.47	0.36	0.63	0.06	0.14
LOG-5 projection error	0.75	0.46	0.11	0.10	0.25	0.62	0.07	0.05	0.20	0.45	0.14	0.08	0.12	0.59	0.30
LOG-6 function error	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.75	0.00	0.01	0.60	0.00	0.25	0.25
Complication	0.05	0.07	0.14	0.37	0.08	0.33	0.37	0.12	0.06	0.46	0.41	0.21	0.40	0.31	0.17

We calculated the relative frequencies of different errors among incorrect final answers (N=2,765) in the 15 exercises, as we wanted to identify which query concepts invite which classes of errors. An answer was considered incorrect if it contained at least one syntax, semantic or logical error. Importantly, a complication by itself did not constitute in making an answer incorrect. In addition to errors, we calculated relative frequencies of complications among all final answers (N=8,773).

In order to identify persistent errors among exercises, we analyzed the data by counting the number of different classes of errors for each exercise. The homogeneity of variance was tested before analysis, and it was found that the data was overdispersed. Therefore, we modeled the data with negative binomial regression that is also common method for count data. Because our interest was in comparing the error rates under the different exercises, we added fixed effects of task to the model. The non-independence of the observations due to the fact that each student completed multiple exercises was modeled by including random effect of student in the model. We estimated the model using the SPSS (version 24) Mixed command, and interpreted the results by calculating the predicted number of errors and their confidence intervals (CI) for each exercise. We estimated the statistical significance using an overlap rule of 95% for the CI bars, as proposed by Cumming et al. [6].

4 RESULTS

4.1 Error Persistence for Error Categories

We collected success rates, relative error frequencies by error category in incorrect final answers, and complications in all final answers to Table 2. We ignored non-attempts, as not all students attempted to solve all exercises. Some errors were absent in some exercises altogether. Again, we observed high numbers of relative frequencies among logical errors when compared to other categories. In fact, for each of the logical error categories, at least one exercise yielded a relative frequency of at least 0.40. By contrast, e.g., illegal aggregate function placement (SYN-4), inconsistent join (SEM-2), and redundant column output (SEM-5) error categories had a relatively low highest relative frequency of at most 0.02. Complications were present in all exercises.

4.2 Error Persistence for Error Classes

We collected estimated means and CIs of syntax, semantic, and logical errors as well as complications in final answers per exercise to Table 3. Collectively, logical errors were more prominent than other error classes in all exercises. The results show that on average, students are more likely to commit persistent syntax errors in exercise C15 (estimate 0.828 with 95% CI [0.721, 0.951), whereas persistent semantic errors were most prominent in exercise B10 (estimate 0.855 with 95% CI [0.735, 0.995]).

Investigation concerning the class of logical errors showed that students committed fewer persistent logical errors in exercises A1, A2, A3, B11 and C15 when compared to exercises B6, B7, B9, B10, B12, and B13. The overall trend showed that, with a few exceptions, the means of logical errors increase over time, but in the final exercises C14 and C15, the means decreased.

Inspection of complications showed that students wrote more complications in exercises B10, B13, and C14 when compared to exercises A1, A3 and B5. Additionally, exercise B9 showed the widest CIs among all error classes.

Table 3: Estimated Means and CIs for Each Errors Class per Exercise

Exercise	Syntax errors			Semantic errors			Logical errors			Complications		
	Mean	95% CI		Mean	95% CI		Mean	95% CI		Mean	95% CI	
		LL	UL		LL	UL		LL	UL		LL	UL
A1	0.556	0.464	0.666	0.402	0.325	0.498	0.939	0.817	1.079	0.422	0.343	0.519
A2	0.533	0.426	0.666	0.370	0.283	0.485	0.845	0.708	1.009	0.491	0.390	0.620
A3	0.578	0.491	0.682	0.385	0.314	0.471	0.885	0.775	1.011	0.431	0.356	0.521
B4	0.566	0.455	0.704	0.392	0.301	0.510	1.098	0.939	1.285	0.539	0.431	0.675
B5	0.637	0.521	0.777	0.390	0.302	0.505	1.026	0.877	1.202	0.396	0.307	0.510
B6	0.555	0.444	0.693	0.395	0.303	0.515	1.266	1.092	1.467	0.532	0.424	0.667
B7	0.522	0.419	0.651	0.442	0.348	0.563	1.213	1.049	1.402	0.547	0.441	0.679
B8	0.505	0.386	0.660	0.602	0.470	0.770	1.171	0.982	1.397	0.555	0.430	0.716
B9	0.396	0.260	0.602	0.389	0.254	0.595	1.282	1.015	1.619	0.382	0.249	0.585
B10	0.639	0.537	0.759	0.855	0.735	0.995	1.325	1.175	1.494	0.652	0.550	0.774
B11	0.585	0.514	0.667	0.372	0.315	0.438	0.819	0.734	0.915	0.528	0.460	0.606
B12	0.687	0.598	0.791	0.467	0.394	0.554	1.409	1.278	1.553	0.540	0.461	0.632
B13	0.571	0.464	0.703	0.526	0.423	0.655	1.420	1.244	1.621	0.625	0.512	0.763
C14	0.692	0.569	0.841	0.440	0.344	0.562	1.062	0.907	1.243	0.630	0.514	0.773
C15	0.828	0.721	0.951	0.425	0.351	0.516	0.949	0.834	1.080	0.482	0.402	0.578

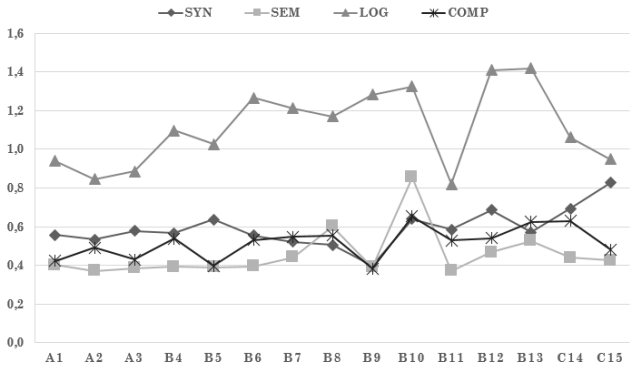

Figure 1: Estimated Means for Each Error Class

5 DISCUSSION

5.1 Persistent Errors in Previous Studies

Although Ahadi et al. [2] used slightly different query concepts, and did not quantitatively study which types of errors occur, we can compare the success rates of our results to theirs to gain understanding on which concepts students struggle with. Their study listed seven query concepts, five of which map to the framework [11] we used. *Simple, one table* [2] corresponds to *single-table* (exercises A1, A2 and A3). Their study reports success rate of 90%, while our rates vary from 60% to 77%. The success rate of a query with *group by* [2], which corresponds to *grouping* (exercise C14), is 75%, which is the same as in our results.

Their query for *group by with having* [2], which corresponds to our *grouping restrictions* (exercise C15), has a success rate of 61%, while ours is 56%. *Self-join* [2] corresponds to *self-join* (exercises B11 and B13), but the success rate of their study (24%) differs from ours (41% and 62%). Finally, *correlated subquery* [2] corresponds to *correlated subquery* (exercise B11), and the success rates are fairly similar with their 46% and our 41%. It is worth noting that most of our exercises tested more than one concept instead of just one. What differs between these two studies, is that the most difficult query concept in their results was self-join, but in our results the most difficult concepts were correlated and uncorrelated subqueries, with success rates of 41% and 43%, respectively.

An earlier study on syntax errors in the whole query writing process [1] (as opposed to our study, in which we only analyzed final answers) used PostgreSQL to categorize syntax errors, which makes the results uncomparable to ours. However, some similarities can be found, as the study points out *syntax error* (corresponds to SYN-6 common syntax error), *undefined column* and *undefined table* (SYN-2), and *grouping error* (SYN-5 illegal or insufficient grouping) among the most frequent syntax errors. Additionally, by examining the final incorrect answers, they discovered that 51% of students abandoned the exercise when they were not able to fix a syntactic error. Ahadi et al. [3] explored persistent semantic errors related to query concepts. The study lists several common errors, most of which correspond to the error categorization we used; *missing/wrong condition* (corresponds to LOG-4, also possibly SEM-1), *self-join not used* (SEM-3 missing join, also possibly SEM-2 inconsistent join), *missing group by or having clause* (SYN-5 illegal or insufficient grouping, and LOG-4 expression error), *use of wrong column* (SYN-5 illegal or insufficient grouping, and LOG-4 expression error), *missing order by clause* (LOG-5 projection error), *incorrect/incomplete column* (LOG-5 projection error), and *missing/extra column in select* (LOG-5 projection error). When compared to our findings in Table 2, these error categories are among the most prominent ones, and our findings seem to support theirs. Additionally, based on our results, data type mismatch (SYN-3), common syntax (SYN-6), operator (LOG-1), join (LOG-2), nesting (LOG-3) and function errors (LOG-6) were relatively frequent in the exercises.

In summary, the available evidence seems to suggest that students struggle with similar query concepts, and some errors are more persistent than others. The consensus view among our results and the results by Ahadi et al. [3] seems to be that the most persistent errors are illegal or insufficient grouping (SYN-5), common syntax error (SYN-6), inconsistent expression (SEM-1), inconsistent join (SEM-2), missing join (SEM-3), expression error (LOG-4), and projection error (LOG-5). Additionally, the results by Ahadi et al. [1] and Taipalus et al. [11] seem to agree that, among all queries, the most frequent syntax errors are common syntax errors (SYN-6), undefined database object errors (SYN-2), and, in queries involving aggregate functions, illegal or insufficient grouping errors (SYN-5).

5.2 Persistent Errors versus All Errors

Taipalus et al. [11] counted relative error frequencies in not just final answers, but during the whole query writing process, i.e., in all queries submitted to the DBMS. By comparing their results to those presented in Table 2, we can acquire some insight on which errors are common, but usually corrected by the students. First, rough comparison reveals that ambiguous database object (SYN-1) and illegal aggregate function placement errors (SYN-4) were uncommon in all and in final answers, while undefined database object errors (SYN-2) were common in all answers, but uncommon in final answers, i.e., usually corrected. The occurrence of data type mismatch (SYN-3) and illegal or insufficient grouping (SYN-5) appears to be closely related to the query concepts, as these errors were frequent in all and in final answers, but only in certain exercises. Common syntax errors (SYN-6) were common in all and in final answers.

The most frequent semantic errors among all and final answers were inconsistent expressions (SEM-1), and missing joins (SEM-3). In general, semantic errors were less frequent in both all and final answers than syntax errors, and the least frequent among all four error classes.

In general, logical errors were most frequent in all and in final answers. Interestingly, in 9 of the 15 exercises, operator errors (LOG-1) were more frequent in the final answers than in all answers. This suggests that operator errors are both common, and they have a high tendency to stay uncorrected. Join errors (LOG-2) were common in almost all multi-table queries, both all and final. Nesting errors (LOG-3) were common in queries which required nesting expressions (A3), or designing the subqueries by using previously uncommon nesting (B8). These errors, however, were closely related to the query concepts, and most of the exercises invited no such errors. Expression (LOG-4) and projection (LOG-5) errors were most common in all and in final answers. These types of errors seem relatively common in the query writing process, and are relatively difficult for students to fix. Smelcer [10] studied SQL query writing process in regard to short term memory, and the occurrence of these types of errors might be related to the thought process of a student translating the natural language data demand into SQL. Intuitively, function errors (LOG-6) occurred only in exercises involving aggregate functions (B9, B11, B12, C14, C15). With the exception of exercise B11, function errors were more common in final than in all answers. This suggests that function errors are difficult for students to fix.

Finally, while complications were relatively common in both all and in final answers, the number of final answers with complications was usually considerably lower than the number of all answers with complications. This suggests that, even though complications are still common, many of them are corrected during the query writing process.

In summary, different query concepts invite different errors by design, e.g., a query with aggregate functions invites the possibility of function errors. By examining the maximum error frequencies for each error category in all answers, we can determine the commonness of each error category, and, consequently, by examining the maximum error frequencies for each error category in all answers, we can determine the persistence of each error category. These things considered, it seems reasonable to assume that we as teachers and researchers should focus on errors which are both common and persistent. Our data indicates the same results as Taipalus et al. [11], i.e., such error categories with a maximum (and arbitrary) relative frequency of at least 0.40 are logical (LOG-1 through LOG-6), and illegal or insufficient grouping errors (SYN-5), as well as complications.

5.3 Persistent Errors and Query Concepts

Among syntax errors, the most persistent were data type mismatch (SYN-3), illegal or insufficient grouping (SYN-5), and common syntax errors (SYN-6). Common syntax errors were relatively frequent in all exercises. With the exception of exercise B13, data type mismatch errors seemed to decrease while the students completed more and more exercises, which might suggest that students learned to formulate queries in which the expressions were Boolean type. Illegal or insufficient grouping errors were common, but only in exercises involving aggregate functions (B11, B12, C14, C15), with the exception of exercise B9, which might be explained by the fact that B9 is the only single-table query with aggregate functions, and thus easier to solve. In terms of persistent syntax errors, answers to exercise C15 showed the most syntax errors. This might be explained by the query requiring the use of all six SQL clauses, as the second highest mean among syntax errors is in exercise C14, which, in turn, required the use of five SQL clauses, while all other exercises required the use of three to four clauses.

As seen in Fig. 1, the trend was that semantic errors were the least persistent among the four error classes. Among semantic errors, the most persistent were inconsistent expressions (SEM-1), inconsistent joins (SEM-2), and missing joins (SEM-3). Inconsistent expressions were relatively prominent compared to other semantic errors in almost all exercises. This is rather unsurprising, because expressions are required in almost all exercises (A1 through B13). Also, intuitively, missing and inconsistent joins were relatively common in all multi-table exercises (B4 through B8, and B10 through C15). Persistent semantic errors were most prominent in exercise B10. One explanation might be that this exercise is the first in which the result table must contain data from multiple tables, which in turn invites joins without subqueries. This might be the first time a student needs to use this approach, but it might not be evident that this approach is preferred.

Finally, and most importantly, logical errors were most persistent among the four error classes. Expression (LOG-4) and projection

Table 4: Which Persistent Errors to Expect

Concept	Expect
multi-table	SEM-2 inconsistent join, SEM-3 missing join, LOG-2 join error
equal subqueries	LOG-2 join error
self-join	LOG-2 join error
multiple source tables	SEM-3 missing join
aggregate functions	SYN-5 illegal or insufficient grouping, LOG-6 function error
(all)	SYN-6 common syntax error, LOG-4 expression error, LOG-5 projection error, complication

errors (LOG-5) were common across all exercises, and operator (LOG-1) and join errors (LOG-2) across almost all exercises. Among logical errors, nesting errors (LOG-3) were relatively uncommon, and the persistence of function errors (LOG-6) was closely related to query concepts involving the use of aggregate functions. Based on our results, we collected the different errors that some of the concepts invite to Table 4. For most of the concepts, however, the data showed no distinguishable patterns.

In conclusion, our results provide needed insight on which SQL errors students struggle with, and which aspects we as teachers and researchers should focus on when utilizing SQL. Although Taipalus et al. [11] proposed an operational model for designing SQL exercises and exercise database data, taking into account their framework's 105 different errors for each exercise is arguably an onerous task. We propose that teachers should start the exercise design by focusing on the most expected persistent errors first, and then work down to less common errors depending on their personal time and resource constraints. What's more, our study propounds the view that students are able to correct some types of errors by themselves, and therefore teaching should focus on errors which students struggle with.

5.4 Limitations and Further Research

We compared our results with certain previous studies [1–3], which did not use the same query concepts or error categorization. With this in mind, there is a chance of misinterpretation of the listed concepts and error categories. Furthermore, these concepts and categories might include or exclude aspects of the framework [11] we used. Finally, the framework lists multiple query concepts per exercise, while Ahadi et al. [2, 3] only list one. In some cases, this might be the result of different levels of specificity between the concept listings, but nonetheless makes the cause and effect analysis in our results more difficult, as it not clear whether frequent errors in any one exercise are caused by a particular query concept, or by a combination of two specific query concepts.

For further research, we propose a tool for automatically categorizing errors in student answers according to the error categorization by Taipalus et al. [11]. The categorization is extensive, and complications, semantic errors, and syntax errors must be analyzed by hand. This is further emphasized by the fact that the categorization is DBMS independent, which makes reliable syntax error discovery by a single DBMS unreliable. By automation, the categorization is open to larger datasets, and by making the automation of error categorization real-time, learning environments may provide students feedback as the query is being written.

6 CONCLUSION

In this study, we set out to investigate which errors are persistent, i.e., more difficult for students to fix, and which types of persistent errors different query concepts, such as joins or aggregate functions invite. The results show that logical errors and complications are more persistent than syntax or semantic errors. While function errors were common in exercises with certain query concepts, expression and projection errors were persistent in all exercises, regardless of the query concepts. We propose that while designing SQL exercises, teachers and researchers design the exercise data to take account most persistent errors, starting from the most common ones.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Mikko Rönkkö for his invaluable advice and support regarding the methodology, and the anonymous reviewers for their comments and suggestions.

REFERENCES

[1] Alireza Ahadi, Vahid Behbood, Arto Vihavainen, Julia Prior, and Raymond Lister. 2016. Students' Syntactic Mistakes in Writing Seven Different Types of SQL Queries and its Application to Predicting Students' Success. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16)*. ACM Press, New York, New York, USA, 401–406. https://doi.org/10.1145/2839509.2844640

[2] Alireza Ahadi, Julia Prior, Vahid Behbood, and Raymond Lister. 2015. A Quantitative Study of the Relative Difficulty for Novices of Writing Seven Different Types of SQL Queries. In *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '15)*. ACM Press, New York, New York, USA, 201–206. https://doi.org/10.1145/2729094.2742620

[3] Alireza Ahadi, Julia Prior, Vahid Behbood, and Raymond Lister. 2016. Students' Semantic Mistakes in Writing Seven Different Types of SQL Queries. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education (SIGCSE '16)*. 272–277. https://doi.org/10.1145/2899415.2899464

[4] Stefan Brass and Christian Goldberg. 2005. Semantic errors in SQL queries: A quite complete list. *Journal of Systems and Software* 79, 5 (2005), 630–644. https://doi.org/10.1016/j.jss.2005.06.028

[5] Luca Cagliero, Luigi De Russis, Laura Farinetti, and Teodoro Montanaro. 2018. Improving the Effectiveness of SQL Learning Practice: A Data-Driven Approach. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, Vol. 01. 980–989. https://doi.org/10.1109/COMPSAC.2018.00174

[6] Geoff Cumming, Fiona Fidler, and David L. Vaux. 2007. Error bars in experimental biology. *The Journal of Cell Biology* 177, 1 (2007), 7–11. https://doi.org/10.1083/jcb.200611141 arXiv:http://jcb.rupress.org/content/177/1/7.full.pdf

[7] Mohammad Dadashzadeh. 2003. A Simpler Approach to Set Comparison Queries in SQL. *Journal of Information Systems Education* 14, 4 (2003), 345–348.

[8] Victor M. Matos and Rebecca Grasser. 2002. Teaching Tip A Simpler (and Better) SQL Approach to Relational Division. *Journal of Information Systems Education* 13, 2 (2002), 85–88. http://jise.org/Volume13/Pdf/085.pdf

[9] Gang Qian. 2018. Teaching SQL: A Divide-and-conquer Method for Writing Queries. *Journal of Computing Sciences in Colleges* 33, 4 (April 2018), 37–44. http://dl.acm.org/citation.cfm?id=3199572.3199577

[10] John B Smelcer. 1995. User errors in database query composition. *International Journal of Human-Computer Studies* 42, 4 (Apr 1995), 353–381. https://doi.org/10.1006/ijhc.1995.1017

[11] Toni Taipalus, Mikko Siponen, and Tero Vartiainen. 2018. Errors and Complications in SQL Query Formulation. *ACM Transactions on Computing Education* 18, 3, Article 15 (Aug. 2018), 29 pages. https://doi.org/10.1145/3231712

[12] Charles Welty. 1985. Correcting user errors in SQL. *International Journal of Man-Machine Studies* 22, 4 (1985), 463–477. https://doi.org/10.1016/S0020-7373(85)80051-1