Semantic Errors in SQL Queries: Exam Evaluation 2005-01

Christian Goldberg
Martin-Luther-Universität Halle-Wittenberg
goldberg@informatik.uni-halle.de

Abstract

We investigate classes of SQL queries which are syntactically correct, but certainly not intended, no matter for which task the query was written. For instance, queries that are contradictory, i.e. always return the empty set, are obviously not intended. Current database management systems, e.g. Oracle, execute such queries without any warning.

In this evaluation, we give a statistic of such errors for one special exam and list the concerning SQL exercises and their possible solutions. Section 1 contains important data of the analyzed exam. In section 2 we explain the database scheme(s) that is/are used in the listed exercises together with their possible solutions in section 3. Section 4 conducts a survey on the number and sorts of occurred semantic errors.

1 Exam Data

Lecture Title	:	Database Systems I
Term	:	Winter term 2005/2006
Lecturer	:	Prof. Dr. Stefan Brass
University	:	Martin-Luther-University Halle, Germany
Analysis	:	Christian Goldberg
Date of Analysis	:	November 2008
Error Code Reference	:	[1]

2 Underlying Database Scheme

In the following exercises, we use a database scheme for a digital photo archive which stores information about photos, main topics and subtopics:

```
PHOTO(PNO, FILE, DATE, DESCRIPTION)
MAINTOPIC(MID, NAME)
SUBTOPIC(MID, SID, NAME)
BELONGSTO(PNO, MID, SID)
```

3 Analyzed Exercises and Possible Solutions

The exam "Database Systems I" in winter term 2005/2006 contained 15 exercises about logic, relational calculus, SQL, database modelling, functional dependencies and normalization, transactions and locking. The 6 analyzed SQL queries resulted in 18 out of 37 points. The 40 participating students had 120 minutes to solve the exercises and were allowed to use the lecture script or other notes but no electronic resources.

It was pointed out that unnecessary complications, unnecessary `DISTINCT` and many duplicates may result in a deduction of points.

3.1 Exercise 3a)

Request the number (`PNO`) and description of all photos that belong to the main topic "people".

```
SELECT DISTINCT P.PNO, P.DESCRIPTION
FROM   PHOTO P, BELONGSTO B, MAINTOPIC M
WHERE  P.PNO=B.PNO
AND    B.MID=M.MID
AND    M.DESCRIPTION='people'
```

3.2 Exercise 3b)

Request the number, file and description of all photos whose file or description contains "Christmas". Please use the function `UPPER` (convert to upper case) to make the comparison with file case-insensitive, so that e.g. "christmas" and "CHRISTMAS" would be accepted, too.

```
SELECT PNO, FILE, DESCRIPTION
FROM   PHOTO
WHERE  UPPER(FILE) LIKE '%CHRISTMAS%'
OR     DESCRIPTION LIKE '%Christmas%
```

3.3 Exercise 3c)

Which subtopics do not contain any newer photos? That means, they don't contain any photo since January 1st, 2001. Subtopics that do not contain any photo at all, shall also be listed. Specify `MID`, `SID` and the name of these subtopics.

```
SELECT S.MID, S.SID, S.NAME
FROM   SUBTOPIC S
WHERE  NOT EXISTS(SELECT *
                  FROM   PHOTO P, BELONGSTO B
                  WHERE  P.PNO=B.PNO
                  AND    B.MID=S.MID AND B.SID=S.SID
                  AND    P.DATE>='01.01.2001')
```

3.4 Exercise 3d)

Which photos show both "Nina" and "Lisa" (both are subtopics). Specify `PNO` and `DESCRIPTION` of these photos. Photos that belong to only one of these subtopics shall not be listed.

```
SELECT P.PNO, P.DESCRIPTION
FROM   PHOTO P
WHERE  EXISTS(SELECT *
              FROM   BELONGSTO B1, SUBTOPIC S1
              WHERE  B1.MID=S1.MID AND B1.SID=S1.SID
              AND    B1.PNO=P.PNO AND S1.NAME='Nina')
AND    EXISTS(SELECT *
              FROM   BELONGSTO B2, SUBTOPIC S2
              WHERE  B2.MID=S2.MID AND B2.SID=S2.SID
              AND    B2.PNO=P.PNO AND S2.NAME='Lisa')
```

Here, a solution with only inner joins (and of course the combination of both) is possible, too:

```
SELECT P.PNO, P.DESCRIPTION
FROM   PHOTO P, BELONGSTO B1, SUBTOPIC S1,
       BELONGSTO B2, SUBTOPIC S2
WHERE  B1.MID=S1.MID AND B1.SID=S1.SID
AND    B2.MID=S2.MID AND B2.SID=S2.SID
AND    B1.PNO=P.PNO AND S1.NAME='Nina'
AND    B2.PNO=P.PNO AND S2.NAME='Lisa'
```

3.5 Exercise 3e)

Generate a list of all main topics and subtopics by specifying `MID`, `SID` and the according `NAME` entitled as `TEXT`. For main topics the value 0 shall be used as `SID`. For subtopics `TEXT` shall be the name indented with four leading blanks (use the concatenation '`||`'). Rank the output first according to `MID` and then to `SID`.

```
SELECT    MID, 0 AS SID, NAME AS TEXT
FROM      MAINTOPIC
UNION ALL
SELECT    MID, SID, '  '||NAME AS TEXT
FROM      SUBTOPIC
ORDER BY  MID, SID
```

3.6 Exercise 3f)

Write a query that requests `MID`, `SID`, `NAME` for every subtopic and the number of photos that belong to that subtopic. Include only subtopics with at least two photos and rank the output according to the number of photos, with the highest number first.

```
SELECT   S.MID, S.SID, S.NAME, COUNT(B.PNO)
FROM     SUBTOPIC S, BELONGSTO B
WHERE    S.MID=B.MID AND S.SID=B.SID
GROUP BY S.MID, S.SID, S.NAME
HAVING   COUNT(B.PNO)>=2
ORDER BY COUNT(B.PNO) DESC
```

4 Statistics

The list of error types mentioned in [1] is based on our experience from grading a large number of exams and homework. (Error 1a is new and not mentioned in [1], it means: Unnecessary outer query.) After this error taxonomy was finished, we analyzed the solutions of the SQL exercises

in several exams of the course "Databases I" at the University of Halle The results for the final exam in winter term 2005/2006 are shown in Figure 1. The exercises are numbered with the numbers and letters from section 3, Further course material and exam exercises are available from the project web page ([4]).

Error	3a	3b	3c	3d	3e	3f	$\sum$
1	-	1	3	13	-	-	17
1a	-	-	2	-	2	1	5
2	3	5	-	1	-	-	9
5	1	-	2	4	1	3	11
6	4	1	2	5	7	10	29
7	-	-	-	5	-	1	6
8	-	6	6	-	-	1	13
12	-	3	-	-	-	-	3
13	-	-	2	1	-	1	4
14	-	-	4	1	-	-	5
17	-	-	-	-	-	24	24
19	-	-	-	-	-	1	1
22	-	-	2	-	1	-	3
25	-	-	-	-	-	2	2
26	-	-	1	-	11	-	12
27	1	-	12	11	1	11	36
28	-	2	4	1	-	1	8
30	-	-	-	2	-	-	2
31	-	-	2	1	1	3	7
34	1	5	-	-	-	-	6
36	-	-	2	-	-	-	2
37	25	1	19	-	3	2	50
39	-	-	2	1	-	-	3
Only Semantic	27	11	19	24	10	13	43.3%
Syntax and Semantic	3	9	14	5	9	21	25.4%
Correct	6	10	3	7	5	3	14.2%
Wrong Task	1	7	3	1	5	-	7.1%
Only Syntax	3	3	1	3	4	1	6.3%
Not Counted	-	-	-	-	7	2	3.8%

Figure 1: Error statistics for final exam, winter term 2005/2006

In some solutions we did count several unrelated semantic errors in the same exercise, but in most cases they did not interact, thus almost all semantic errors (that did not occur simultaneously with syntactic errors) could have been found by our methods, described in [2]. Otherwise, if an error occurred more than once in the same exercise it was counted only once in our statistics. Error 27 (Missing join condition) almost always involves Error 37 (Many duplicates). Thus, in our statistics Error 37 is only counted if it occurs independent from Error 27.

The number of exams that contained at least one semantic error is the sum of the entries "Only semantics" and "Syntax and Semantic". Of course we counted only semantic errors from our list in [1], i.e. that are detectable without knowing the task of the query. "Wrong task" lists the number of exams that can only be detected as incorrect if the goal of the query is known. "Not counted" lists exams that did not try the particular exercise, or that contained so severe syntax errors that looking at semantic errors in detail was not possible. In this exam that we analyzed with this error taxonomy, the occurred semantic errors are (percentages are relative to all detected semantic errors):

1.	19.4 %	Error 37: Many duplicates
2.	14.0 %	Error 27: Missing join condition
3.	11.2 %	Error 6: Unnecessary join
4.	9.3 %	Error 17: Unnecessary argument of `COUNT`
5.	6.6 %	Error 1: Inconsistent condition
6.	5.0 %	Error 8: Implied, tautological or inconsistent subcondition

References

[1] Stefan Brass and Christian Goldberg. Semantic Errors in SQL Queries: A Quite Complete List. In: *Elsevier's Journal of Systems and Software 79(5)*, 2006.

[2] Stefan Brass and Christian Goldberg. Proving the Safety of SQL Queries. In: *Fifth International Conference on Quality Software (QSIC'05)*, IEEE Computer Society Press, 2005.

[3] Stefan Brass and Christian Goldberg. Detecting Logical Errors in SQL Queries. In: *16th Workshop on Foundations of Databases (GvD'04)*, 2004.

[4] Stefan Brass and Christian Goldberg. SQLLint: Detecting Logical Errors in SQL Queries. Project website: http://dbs.informatik.uni-halle.de/sqllint/