**Eduard Cuadros 2017329**

**Edwin Garces 2017201**

**Thiago Almeida 2017201**

**Part A - Database analysis and design**

\_ As **a group**, provide a complete description of the application domain and any assumptions made;

**5%**

We as a team have decided to create a complete database specifically for a library in our school, which allows administrative staff and students to constantly monitor the books. Basically we have created a series of tables that allow us to have total control of each of the books. We have created the table called Books which contains: Id\_book, Title\_book and Type\_book. On the other hand we have the Students table: Id\_student, Name\_student, Gender and Course.

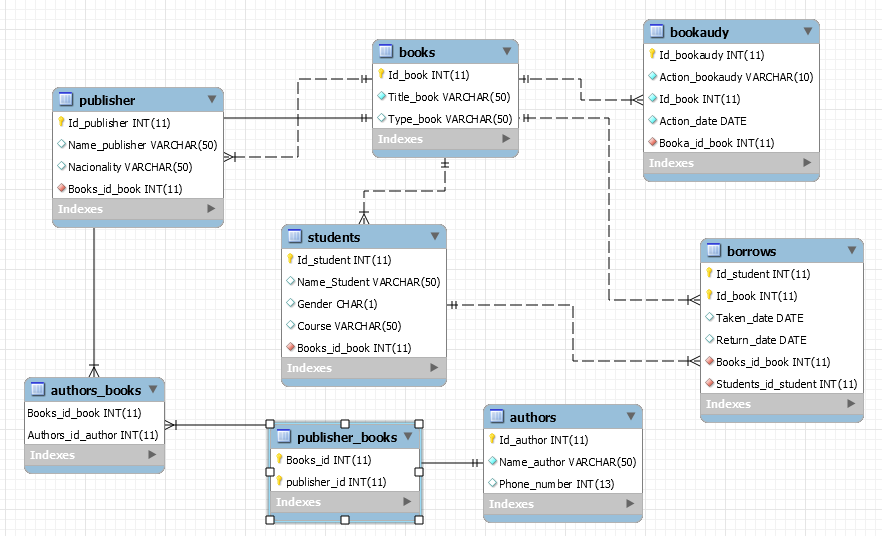
We have also created a table called Authors that contains: Id\_author, Name\_author, and Phone\_number. Without leaving aside Publisher (Id\_publisher, Name\_publisher, Nacionality). The table called Bookaudy (Id\_bookaudy, Action\_bookaudy, Id\_book, Action\_date). And the relations tables Authors\_books and Publisher\_books.

We are developing this database in great detail so that we can get the best out of it.

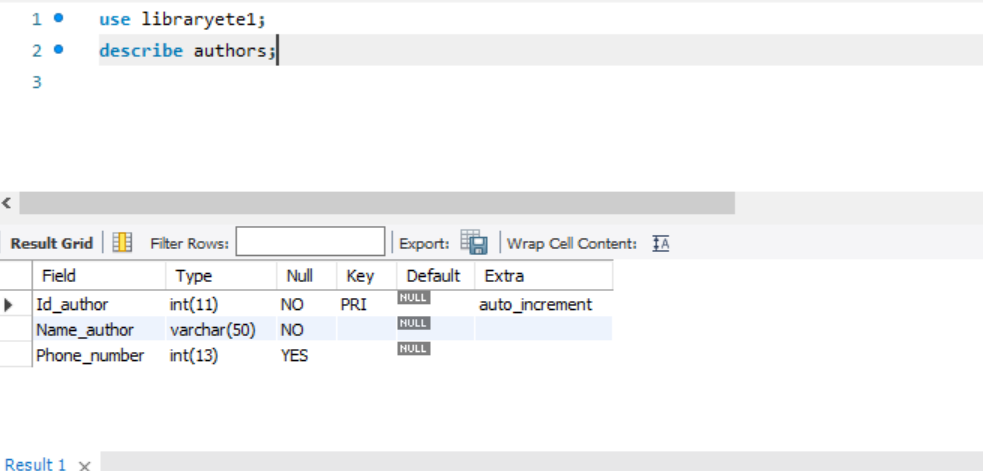
\_ As **a group**, provide complete **design documentation** in the form of complete table descriptions

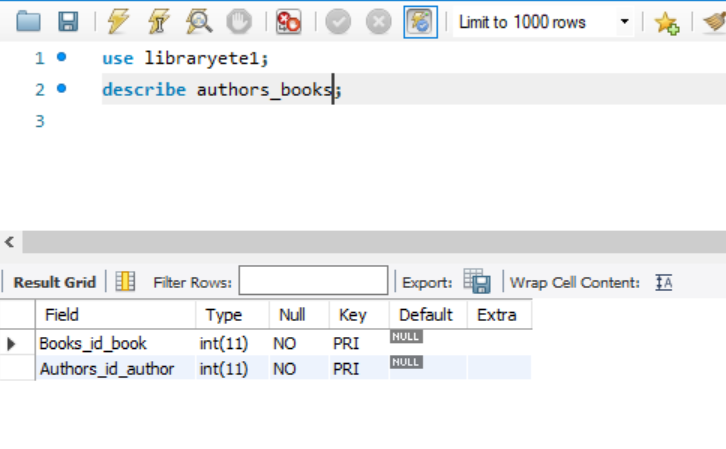
and an **enhanced entity-relationship diagram**;

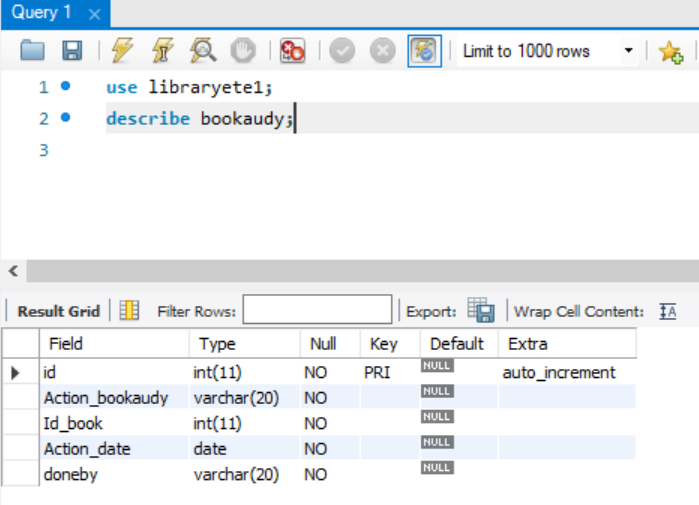
// we need to change this part with the arrows….

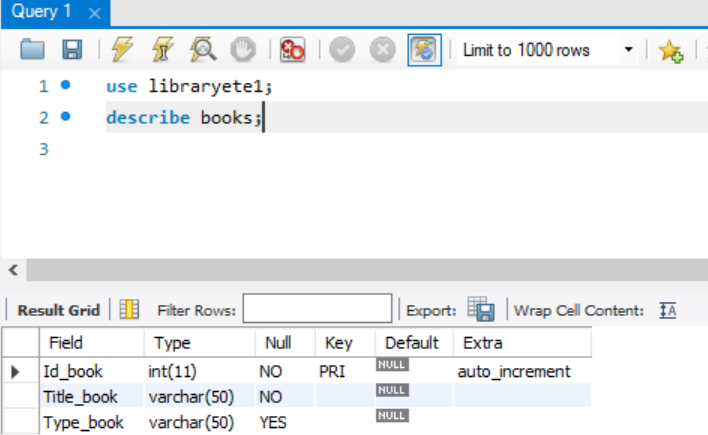


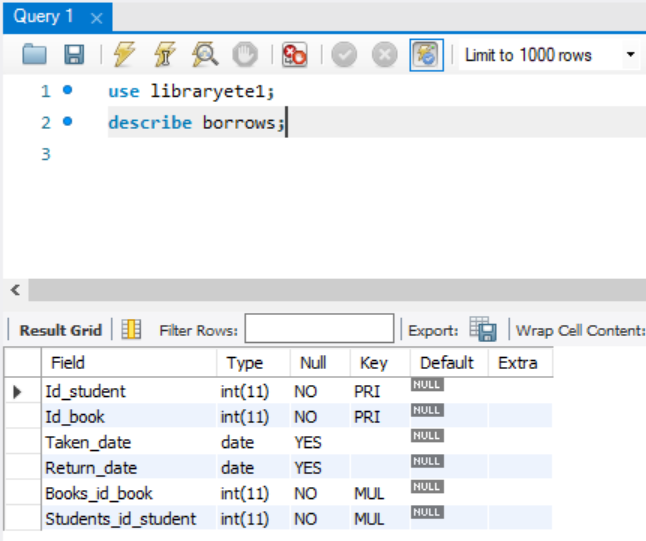
\_

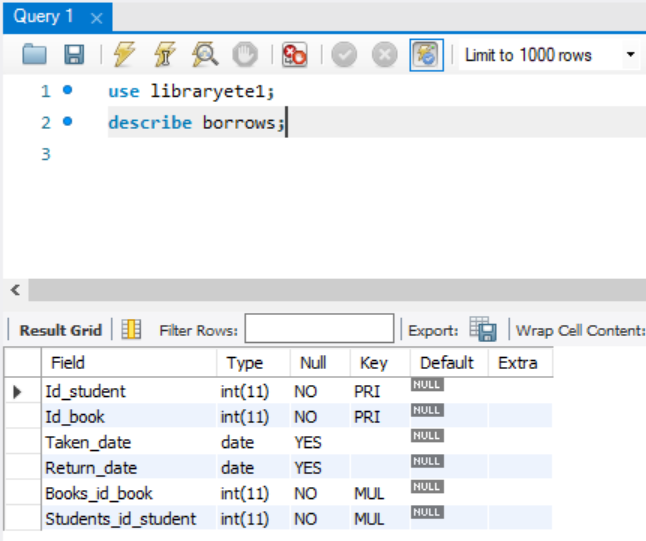


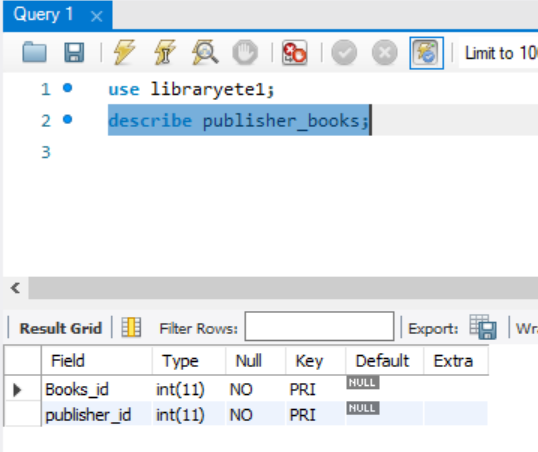


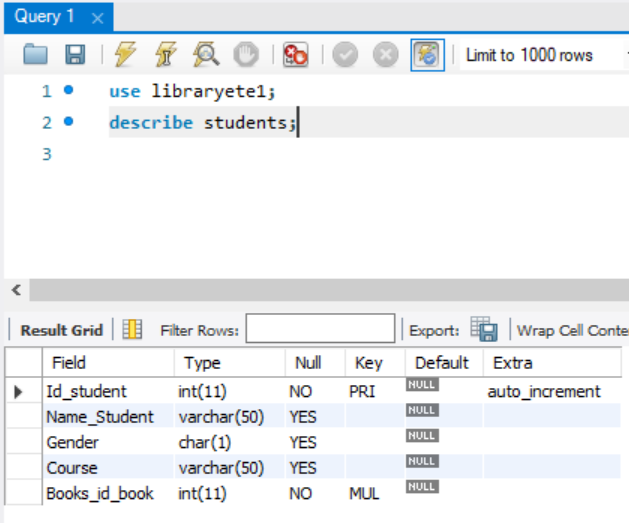








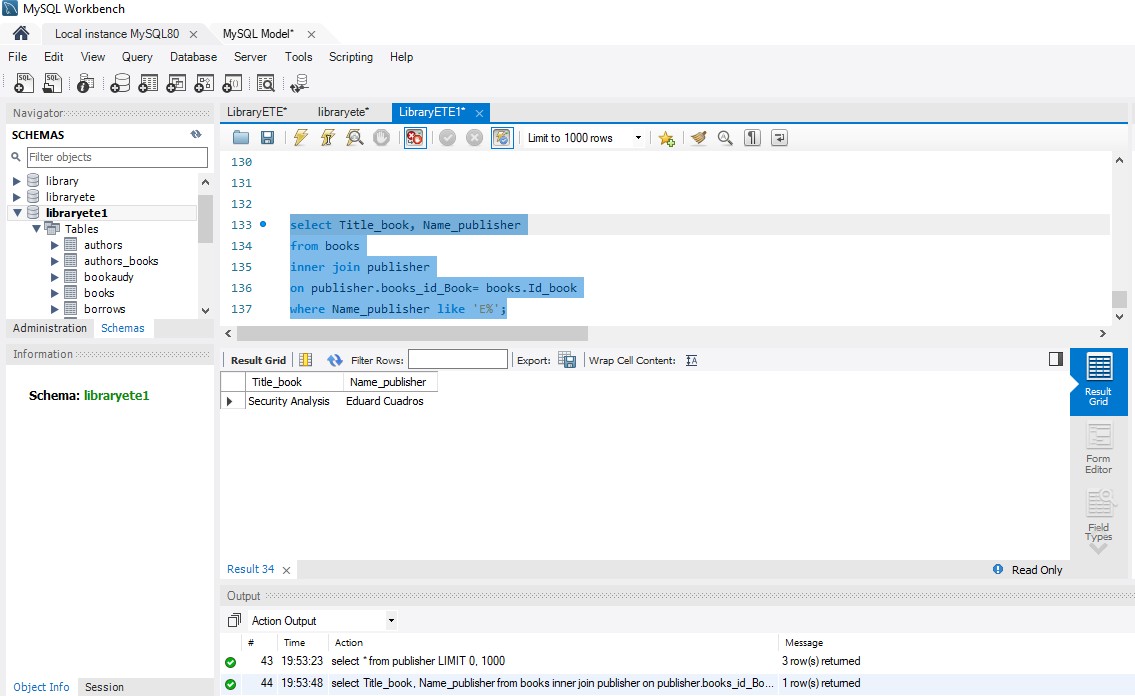




An example of each of the following:

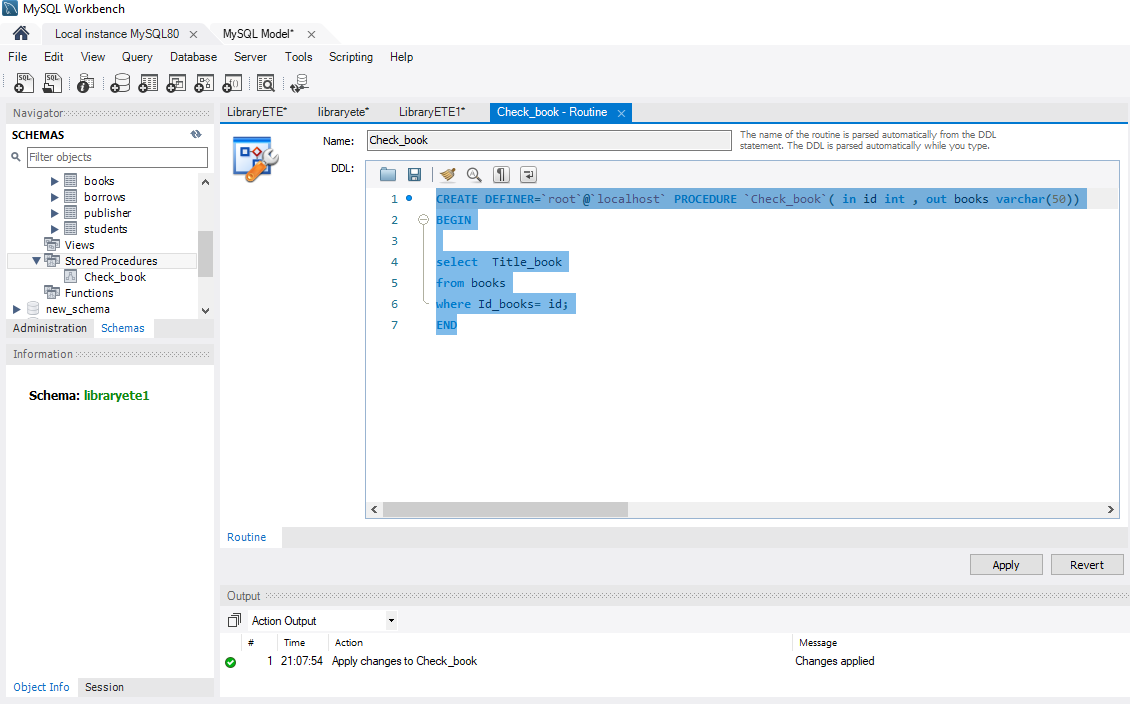
Eduard Cuadros 2017329

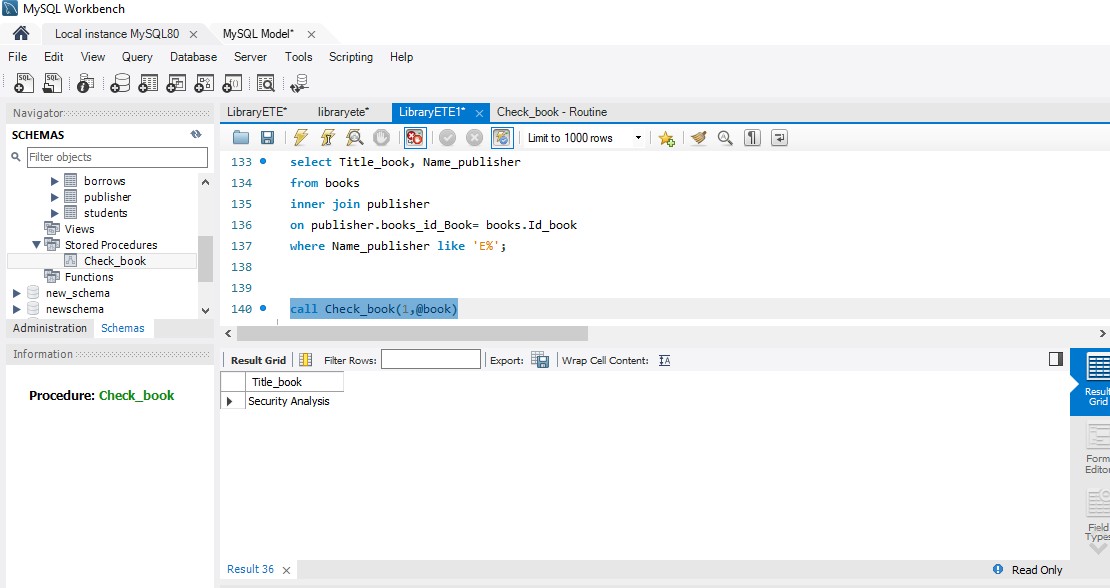
* One view **per student** incorporating at least two tables and a WHERE clause; **5%**



* One stored procedure **per student** that performs a multi-table SELECT query and includes at

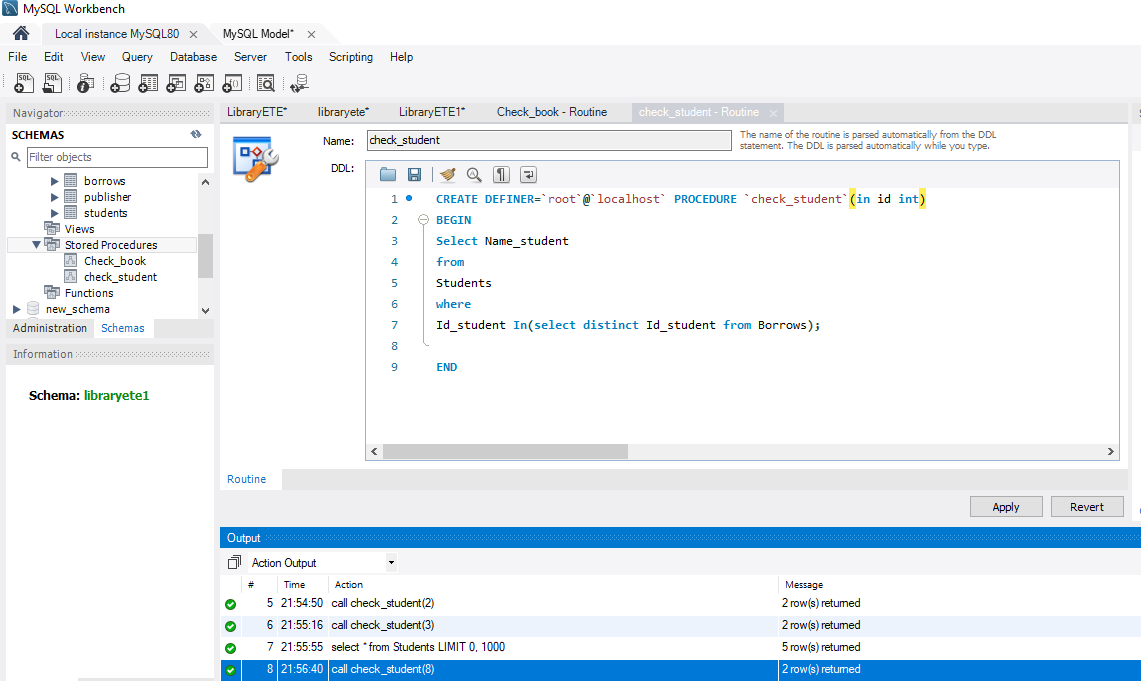
least one IN and one OUT parameter; **5%**





* One stored procedure **per student** that performs a correlated subquery and includes at least

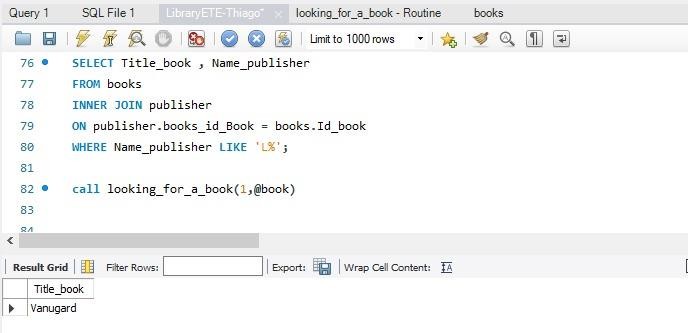
one IN parameter;



\_ An example of each of the following:

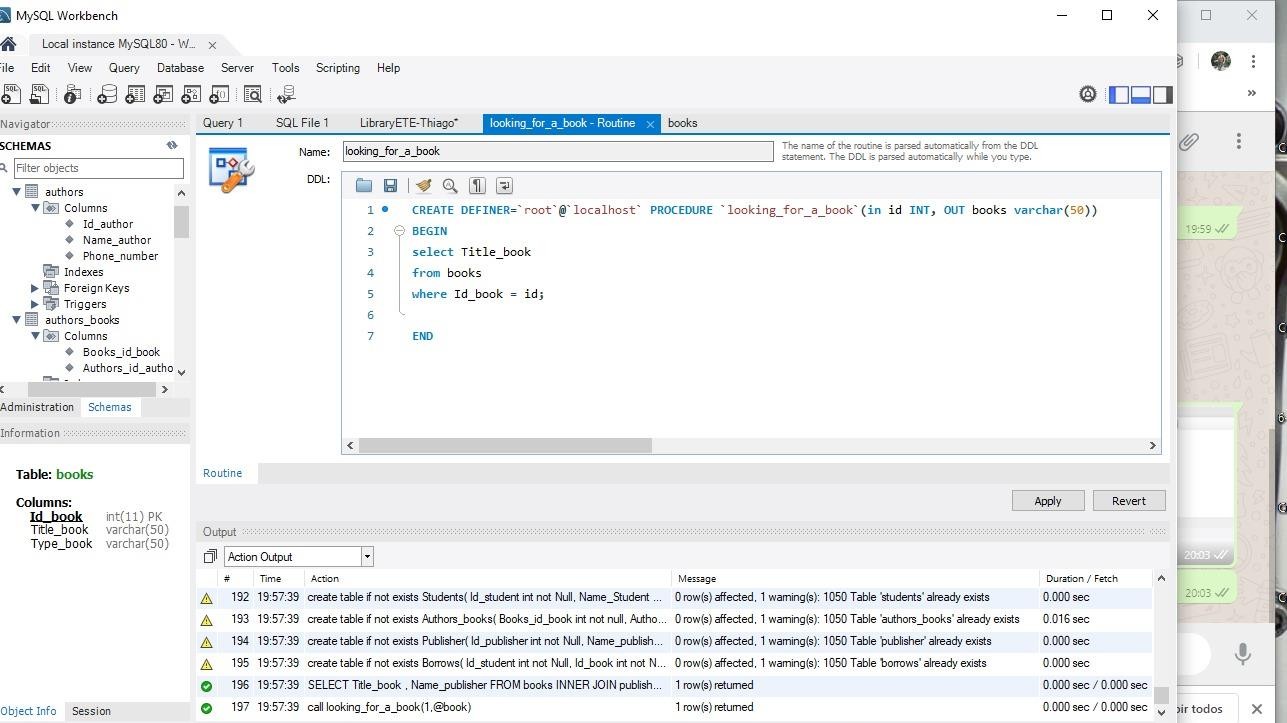
Thiago Almeida

* One view **per student** incorporating at least two tables and a WHERE clause; **5%**



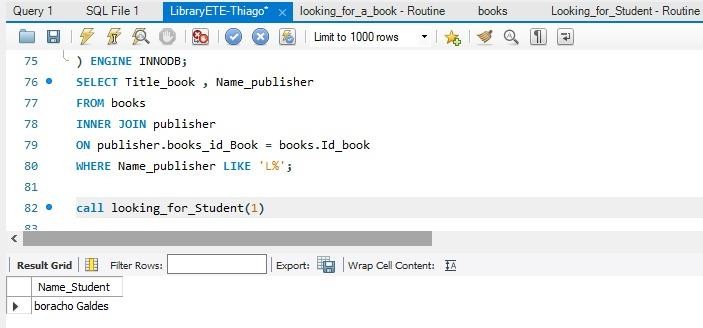
* One stored procedure **per student** that performs a multi-table SELECT query and includes at

least one IN and one OUT parameter; **5%**



* One stored procedure **per student** that performs a correlated subquery and includes at least

one IN parameter; **5%**



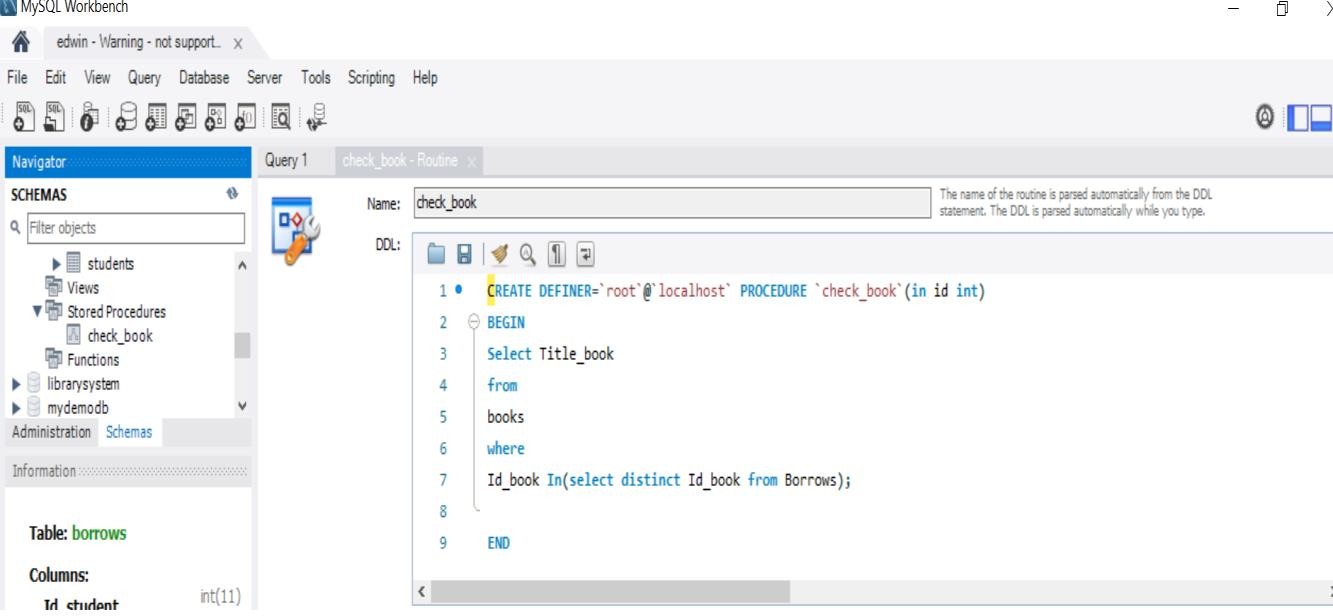
* One triggers **per group**, each of which must be a separate type (e.g. on UPDATE, on DELETE,

on INSERT);

\_ An example of each of the following:

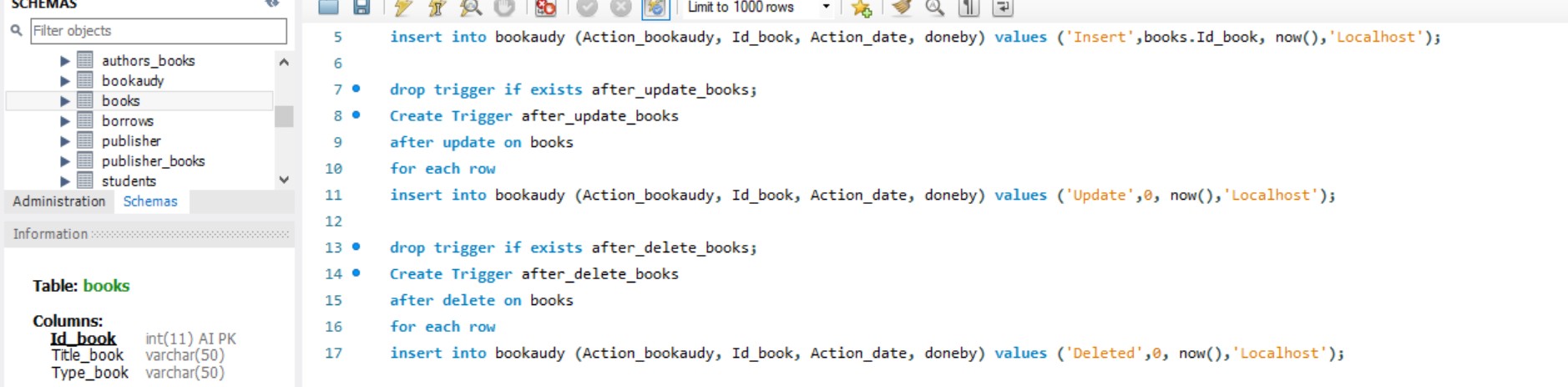
Edwin Garces

* One view **per student** incorporating at least two tables and a WHERE clause; **5%**



* One triggers **per group**, each of which must be a separate type (e.g. on UPDATE, on DELETE,

on INSERT);



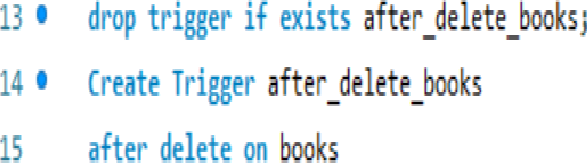
e u r r R u a e e sky •• •= I « 4 ua











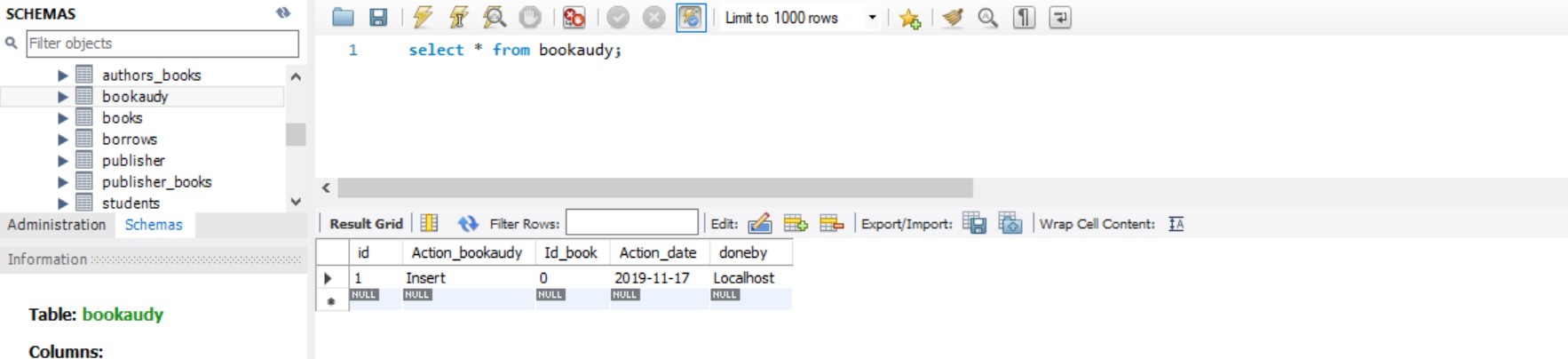




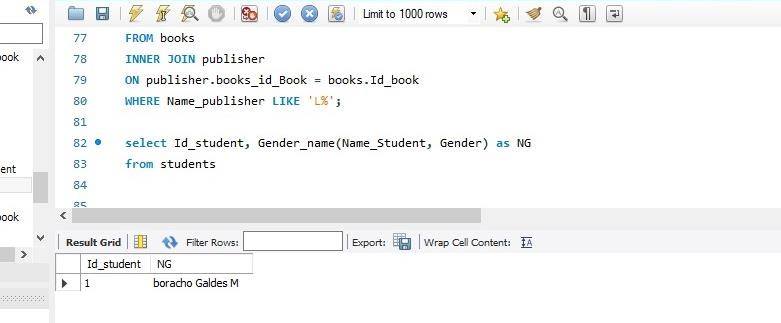


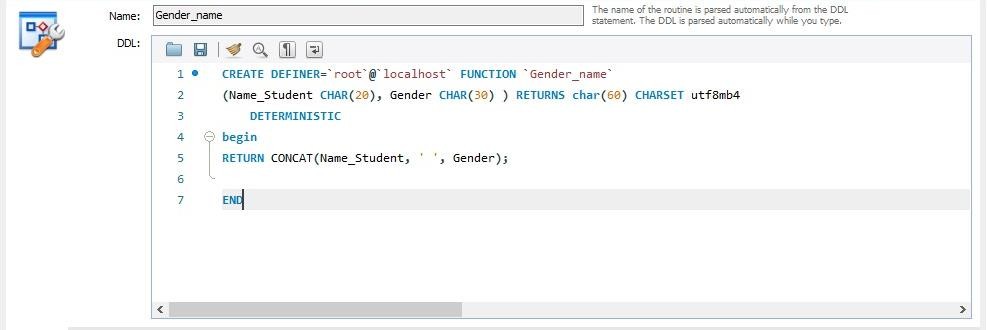


 fDlia



One stored function **per group**;



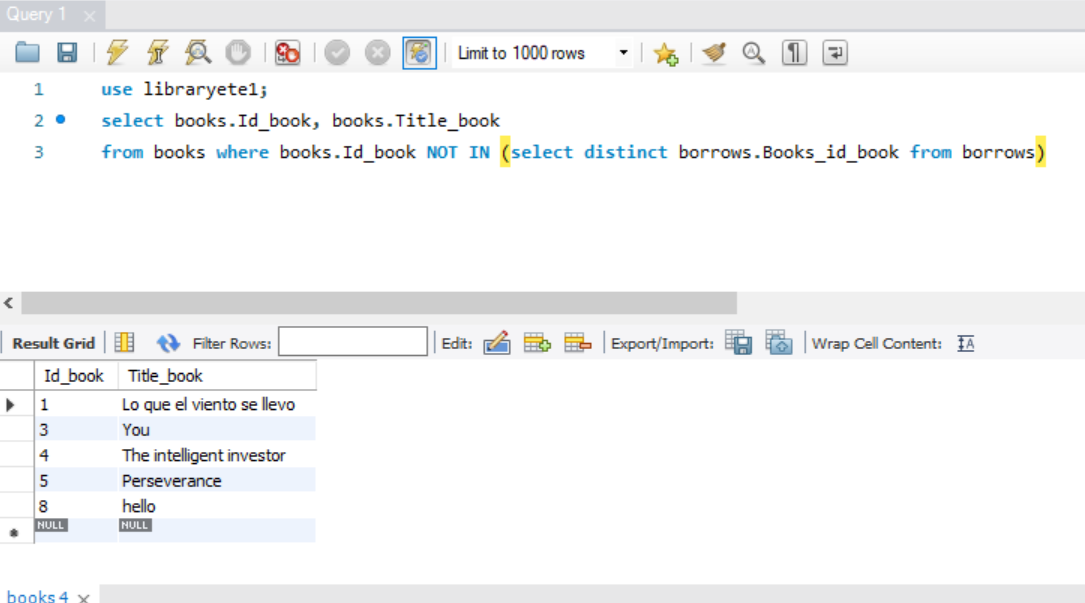


Part B

Not correlated subquery:

We are trying to find books that have not yet been borrowed by anyone.

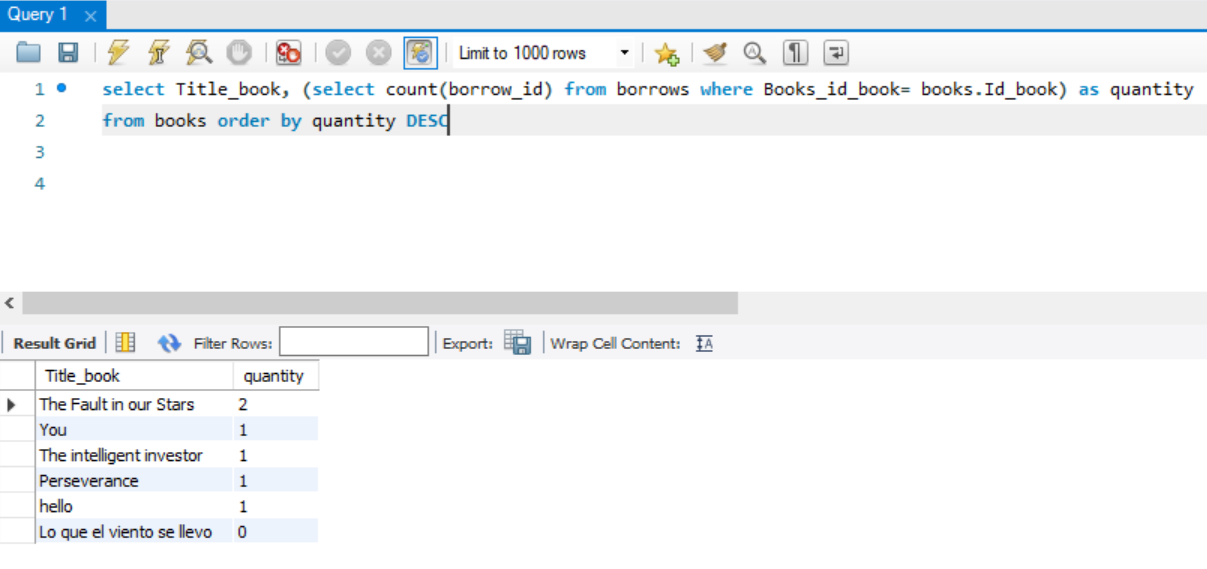
The subquery can still run without the main query which makes it a not correlated subquery



Correlated subquery:

In this query we are looking for the number of times every book has been borrowed. We have entered more data into the borrows table for testing purposes.

In this query, the subquery is correlated to the main query and it can not run on its own



|  |
| --- |
| Transaction analysis form  **Transaction** In this transaction we are trying to get the number of times a title from the books table has been borrowed  **Transaction volume**  **Average**  3 per hour  **Peek**  10 per hour (between 10:00 and 17:00 Monday to Friday ) |
| |  |  |  |  | | --- | --- | --- | --- | | select Title\_book, (select count(borrow\_id) from borrows where Books\_id\_book= books.Id\_book) as quantity  from books order by quantity DESC | |  |  | | --- | --- | | Predicate  Join attribute  Ordering attribute  Grouping attribute  Built-in functions  Attribute updated | Book\_id\_book  Books\_id\_book= books.Id\_book  quantity  none  none  none | | |
| Transaction usage map |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Access | Entity | Type of entity | |  |  |  | | --- | --- | --- | | Per transaction | Per Hour | Per peek |   No of references | | | | 1  2 | Books  Borrows | R  R | 7  6 | 3  10 | 10  30 | | Total references | | | 13 | 13 | 40 |  |  |  | | --- | --- | |  |  | |

Explain command:

