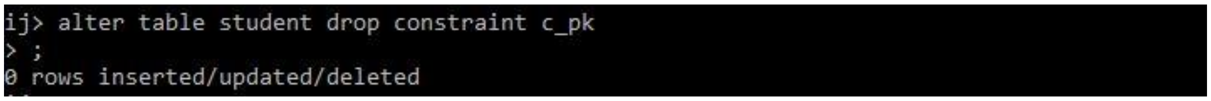
CSCE-310 – Database Systems
Homework 5
Raymond Zhu
923008555

Problem 1

The error is #23513, saying that "the check constraint '*<constraintName>*' was violated while performing an INSERT or UPDATE on table '*<tableName>*'." This error occurred because what was inserted violated the constraint that was set.

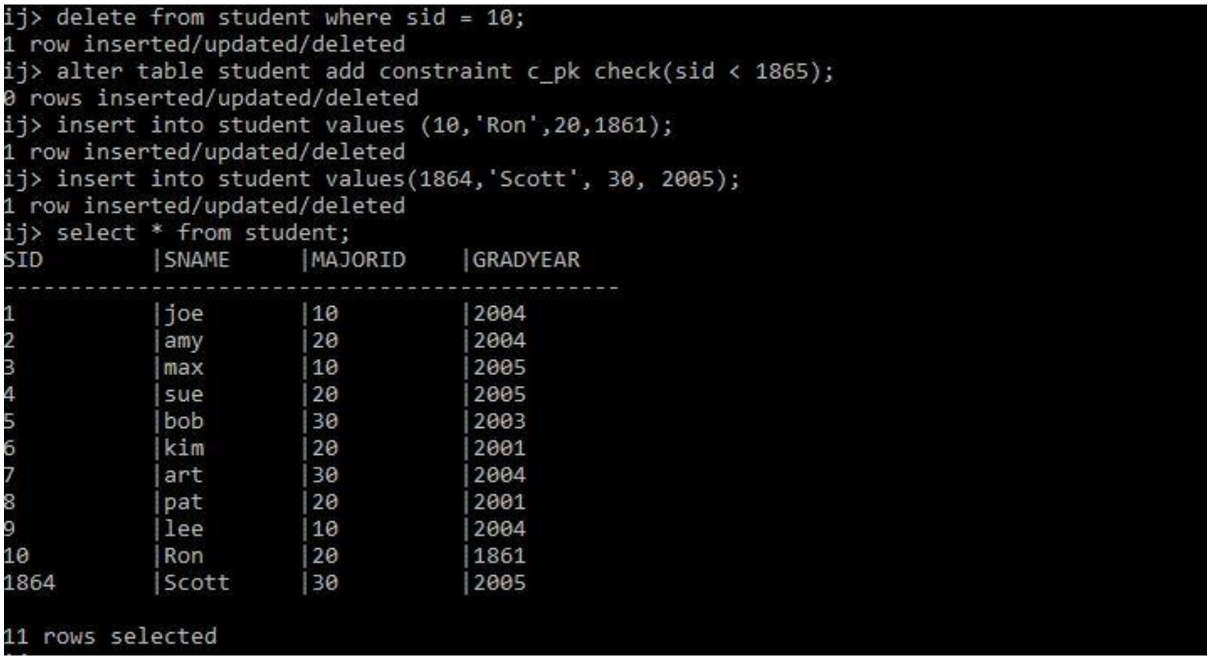
The SQL statement to drop the constraint is listed below:

alter table student drop constraint c_pk;



```
ij> alter table student drop constraint c_pk
> ;
0 rows inserted/updated/deleted
```

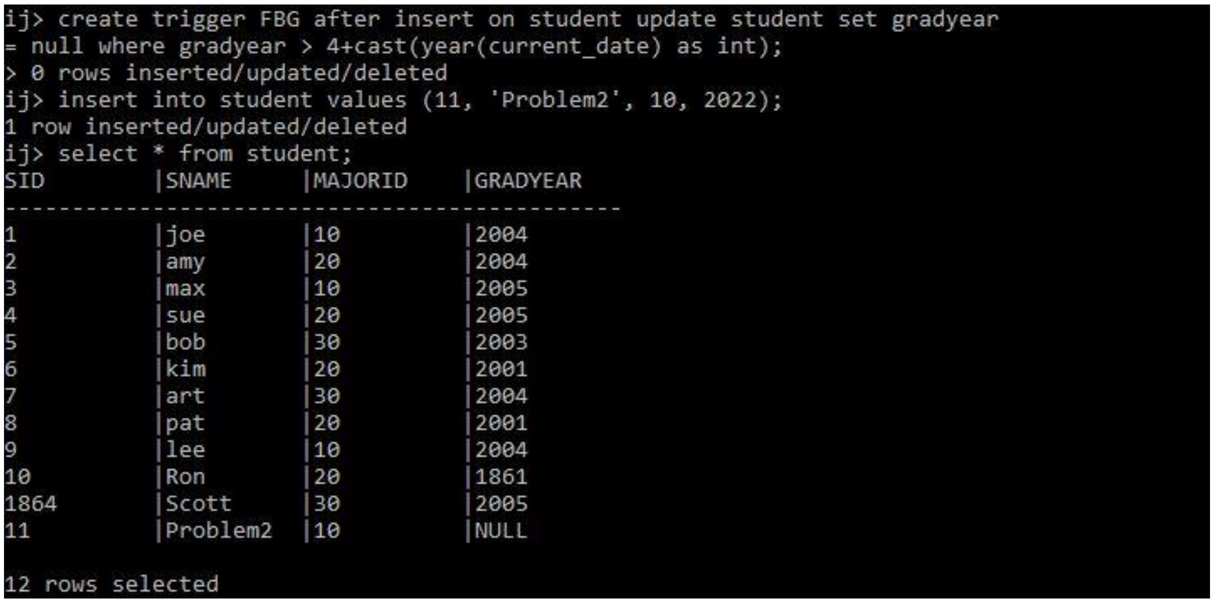
alter table student add constraint c_pk check(sid < 1863);



```
ij> delete from student where sid = 10;
1 row inserted/updated/deleted
ij> alter table student add constraint c_pk check(sid < 1865);
0 rows inserted/updated/deleted
ij> insert into student values (10,'Ron',20,1861);
1 row inserted/updated/deleted
ij> insert into student values(1864,'Scott', 30, 2005);
1 row inserted/updated/deleted
ij> select * from student;
SID        |SNAME     |MAJORID    |GRADYEAR
---------------------------------------------
1          |joe       |10         |2004
2          |amy       |20         |2004
3          |max       |10         |2005
4          |sue       |20         |2005
5          |bob       |30         |2003
6          |kim       |20         |2001
7          |art       |30         |2004
8          |pat       |20         |2001
9          |lee       |10         |2004
10         |Ron       |20         |1861
1864       |Scott     |30         |2005

11 rows selected
```

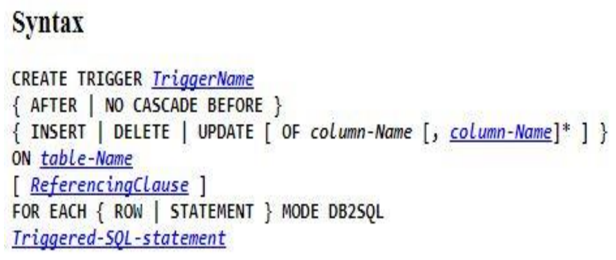
Problem 2:

The trigger will update the student's graduation year to null if upon inserting a student whose graduation is year is 4 years greater than the current year.



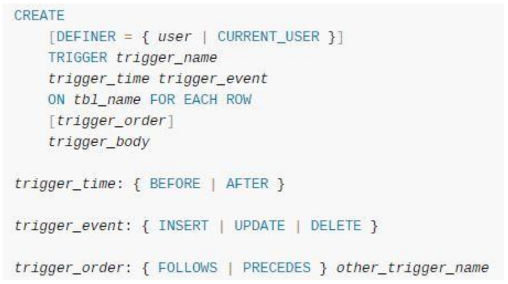
```
ij> create trigger FBG after insert on student update student set gradyear
= null where gradyear > 4+cast(year(current_date) as int);
> 0 rows inserted/updated/deleted
ij> insert into student values (11, 'Problem2', 10, 2022);
1 row inserted/updated/deleted
ij> select * from student;
SID        |SNAME     |MAJORID    |GRADYEAR
-----------------------------------------------
1          |joe       |10         |2004
2          |amy       |20         |2004
3          |max       |10         |2005
4          |sue       |20         |2005
5          |bob       |30         |2003
6          |kim       |20         |2001
7          |art       |30         |2004
8          |pat       |20         |2001
9          |lee       |10         |2004
10         |Ron       |20         |1861
1864       |Scott     |30         |2005
11         |Problem2  |10         |NULL

12 rows selected
```

Problem 3:



Syntax

```
CREATE TRIGGER TriggerName
{ AFTER | NO CASCADE BEFORE }
{ INSERT | DELETE | UPDATE [ OF column-Name [, column-Name]* ] }
ON table-Name
[ ReferencingClause ]
FOR EACH { ROW | STATEMENT } MODE DB2SQL
Triggered-SQL-statement
```



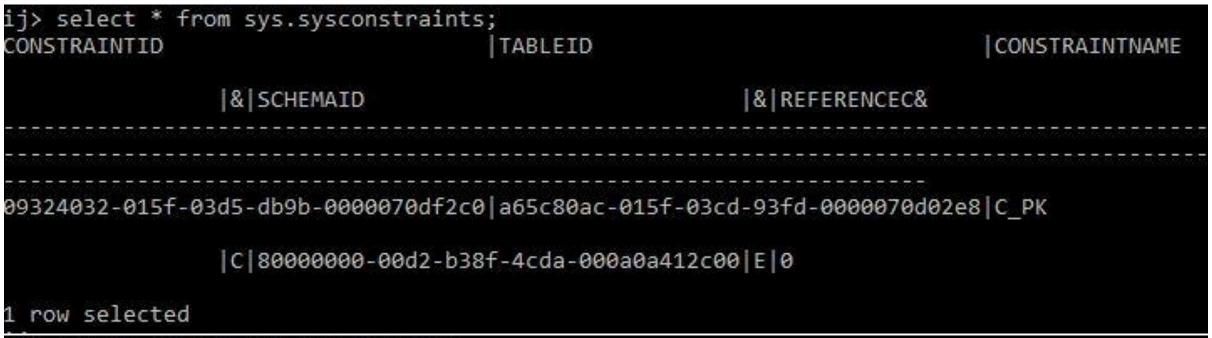
```
CREATE
    [DEFINER = { user | CURRENT_USER }]
    TRIGGER trigger_name
    trigger_time trigger_event
    ON tbl_name FOR EACH ROW
    [trigger_order]
    trigger_body

trigger_time: { BEFORE | AFTER }

trigger_event: { INSERT | UPDATE | DELETE }

trigger_order: { FOLLOWS | PRECEDES } other_trigger_name
```

As listed above, the syntax between derby ij and MySQL differ slightly. Unlike derby ij, MySQL requires a DEFINER and a trigger order. What trigger order means is that a trigger can trigger a trigger. What a DEFINER would do is set restrictions on specific users who are modifying the database. For example, a DBA can set triggers to prevent users who so happened to get access and prevent them from modifying data. Other than that, the rest are pretty much the same except syntax.

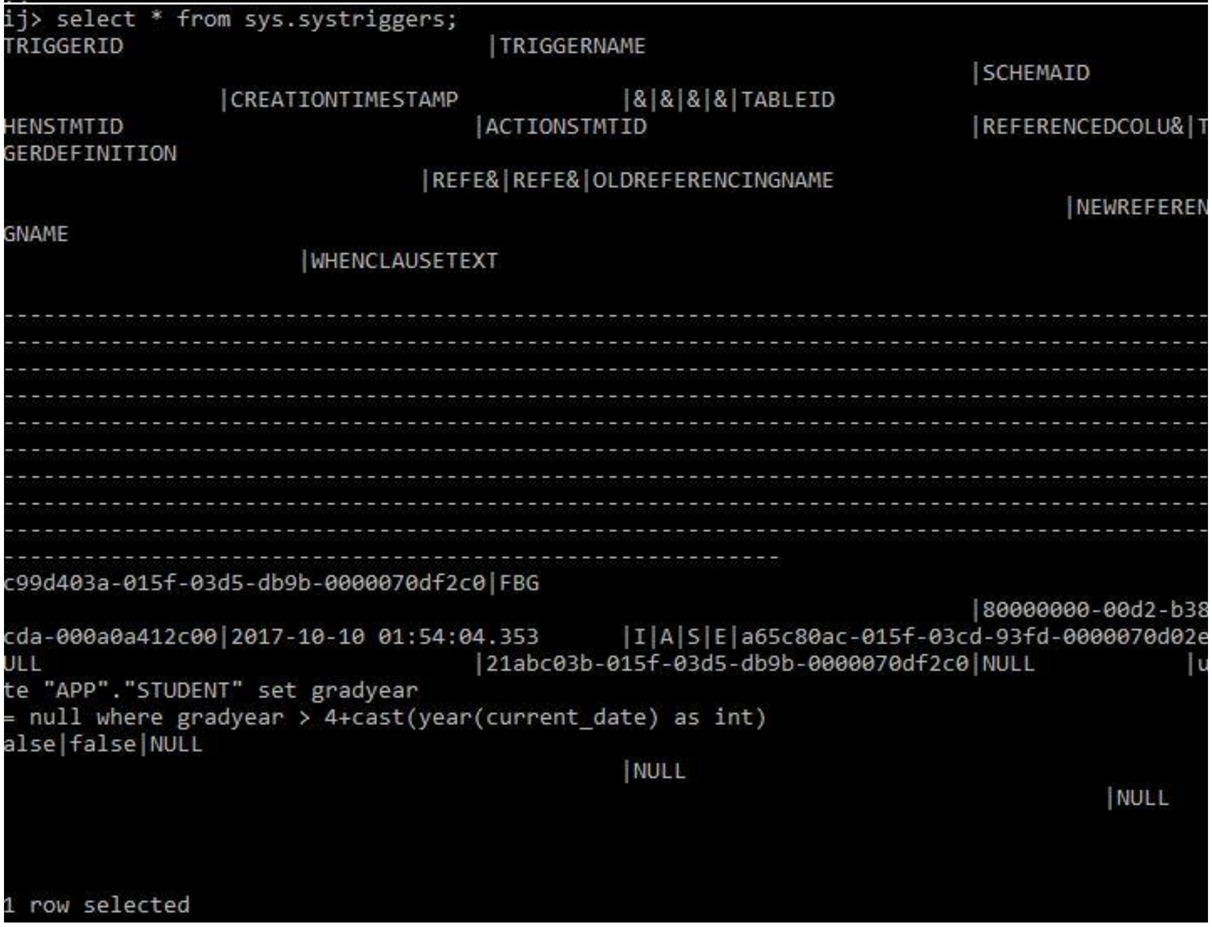
Derby (When a student with gradyear = null is inserted, delete the student):
Create Trigger deleteStudent After Insert on Student Delete From Student Where GradYear = NULL;

MySQL (Will delete the inserted row when readonly user attempts to insert):
Create Definer = 'readonly'@'localhost' Trigger deleteStudent After Insert on Student For Each Row Delete From Student;

Problem 4:



```
ij> select * from sys.sysconstraints;
CONSTRAINTID                        |TABLEID                             |CONSTRAINTNAME

                |&|SCHEMAID                            |&|REFERENCEC&
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
---------------------------------------------------------------------
09324032-015f-03d5-db9b-0000070df2c0|a65c80ac-015f-03cd-93fd-0000070d02e8|C_PK

                |C|80000000-00d2-b38f-4cda-000a0a412c00|E|0

1 row selected
```



```
ij> select * from sys.systriggers;
TRIGGERID                           |TRIGGERNAME
                                                                        |SCHEMAID
                |CREATIONTIMESTAMP            |&|&|&|&|TABLEID
HENSTMTID                          |ACTIONSTMTID                        |REFERENCEDCOLU&|T
GERDEFINITION
                               |REFE&|REFE&|OLDREFERENCINGNAME
                                                                               |NEWREFEREN
GNAME
                      |WHENCLAUSETEXT

------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------
----------------------------------------------------------
c99d403a-015f-03d5-db9b-0000070df2c0|FBG
                                                                        |80000000-00d2-b38
cda-000a0a412c00|2017-10-10 01:54:04.353      |I|A|S|E|a65c80ac-015f-03cd-93fd-0000070d02e
ULL                                |21abc03b-015f-03d5-db9b-0000070df2c0|NULL           |u
te "APP"."STUDENT" set gradyear
= null where gradyear > 4+cast(year(current_date) as int)
alse|false|NULL
                                              |NULL
                                                                                   |NULL



1 row selected
```

Each table consists of its own schema just like how we learned with how DBMS stores its tables. Both tables use an ID as an attribute which is used as the primary key. Parts of the schema contains ID which are foreign keys for other tables creating relations to which what we have learned.

Problem 5:

Returning a yes or no response depending on if there exist duplicate enrollment ids for any given student id and section id

Problem 6:

T/F: The two aspects for the awarding of privilege are:

(1) How they are created initially
(2) How they are passed from user to user

TRUE.

T/F: SQL provides a GRANT statement to allow one user to give a privilege to another

TRUE.

Any privilege that has been passed on by another privileged user that has been revoked does not need to be revoked

FALSE.

An aggie does not lie, cheat, steal, or tolerate those who do.