ASSIGNMENT 6
Insert/Delete/Truncate/Drop

In this lab you will use SQL statements that fall in both the DDL and DML category. In this lab you will be storing new information in the database. You will be using the tables from your previous assignment as such: (Make sure that your tables contain the following columns along with the appropriate constraints

Student
>   SSN   primary key
>   lname
>   fname
>   dob
>   salary   check>10000
>   (lname and fname are a composite candidate key)

Class
>   Class code   primary key
>   Class description   (Create an index on this column using the create index command)

Student_class
>   SSN   Foreign key
>   Class Code   Foreign key
>   (SSN and class code are a composite primary key)

You must execute the statements in the order in which the questions are being asked.

Suggestions:
1)   Do not create a spool file. This lab will probably take several days. Since you cannot guarantee that the work that you did on my home computer or the lab computers on campus will be there the next time you open up the SQLPlus session, I would make the following suggestion: Store all your SQL statements in a text file. Then you can just copy and paste your SQL statements into the SQLPlus session and get back to where you left off.

2)   I would also suggest that you drop all your tables in the beginning of the text file just in case the tables are still there so that you don't get any error messages
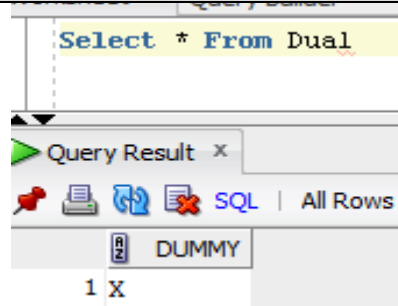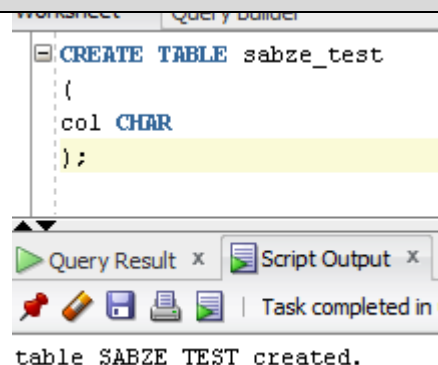
All the tables that you create should be prefixed with the first five letters of your lastname such as **sabze_patient**

What to turn in:
1)   You will turn in this word document only. I do not want any other files
2)   Paste a printscreen of either the **SQLPlus session** or **SQL Developer** showing only the SQL command and the results from the database engine. Some of the SQL statements that you issue may cause an error and may actually be the expected result. Do not assume that just because you are not getting an error message, everything is okay.
3)   When typing in your SQL statements, make sure that the keywords are all in uppercase. The identifiers that you come up with such as table names, column names or constraint names should all be in lower case.
4)   Make sure that you prefix your table names with the first five letters of your last name.
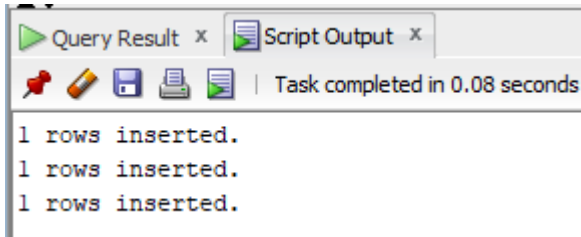5)   Make sure that you only provide a printscreen of the snippet that pertains to the question (NOTHING MORE).

Suggestion: you can use the snipping tool in windows 7 or you can download this open source program http://getgreenshot.org/ for printscreens. Provide only the printscreen
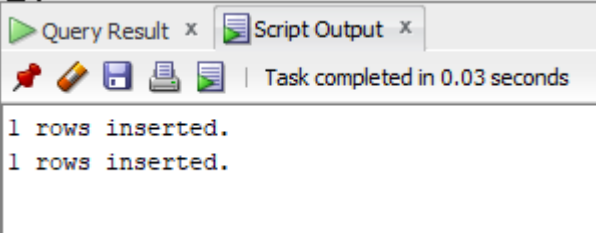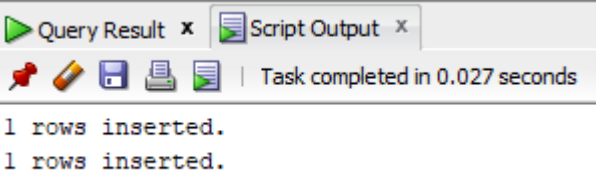
that pertains to the question. **I do not want to see your trial and errors or things that pertain to other questions.**

SQLPlus   or   SQLDeveloper   (Your choice)

Example	Display the contents of the dual table

```
SQL> SELECT * FROM dual;
D
-
X

1 row selected.
```

OR

```
Select * From Dual
```

Query Result ✕

📌 🖨 🔁 ❌ SQL | All Rows

☰ DUMMY

1 X

Next Example	Create a table called test

```
SQL> CREATE TABLE sabze_test
  2  (
  3  col    CHAR
  4  );

Table created.
```

OR

```
CREATE TABLE sabze_test
(
col CHAR
);
```

Query Result ✕   Script Output ✕

📌 ✏ 💾 🖨 ☰ | Task completed in

```
table SABZE_TEST created.
```
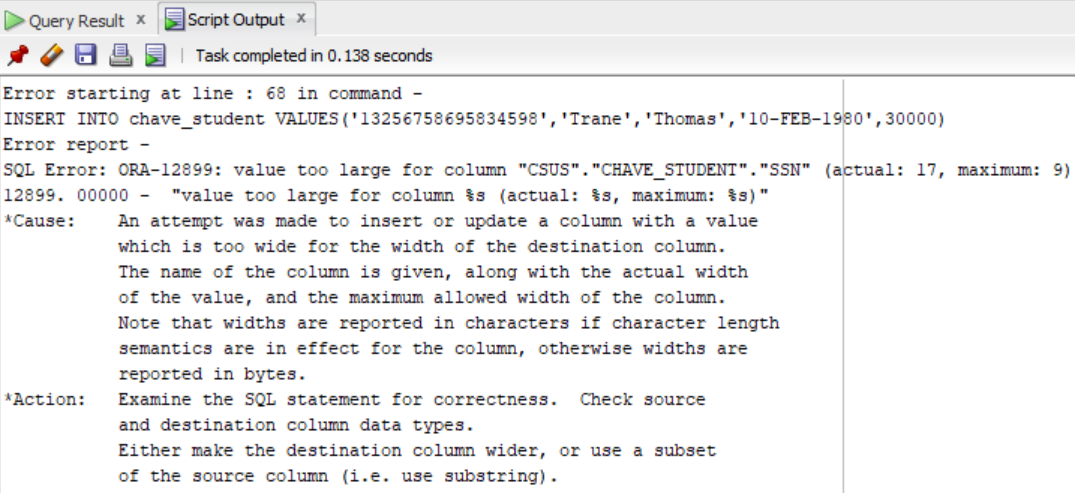
All the tables that you create must be prefixed with the first five letters of your <u>last name</u> such as sabze_student.
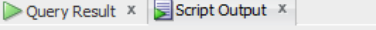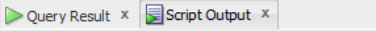
The order in which you insert data into your tables is different from the order in which the questions have been asked. **Questions 1a, 1b and 1c should not give you any error messages**
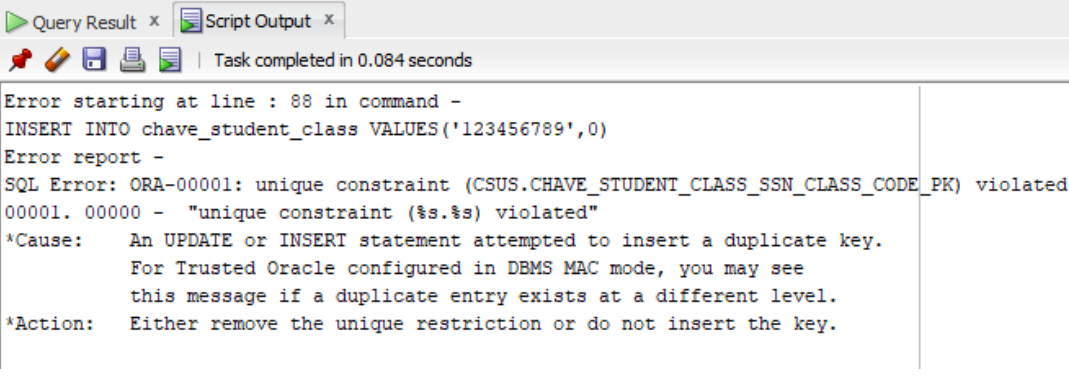
1A	**Insert** three rows of valid data into the **student_class** table

```
INSERT INTO chave_student_class VALUES ('123456789',0);
INSERT INTO chave_student_class VALUES ('123456789',1);
INSERT INTO chave_student_class VALUES ('132567586',1);
```

Query Result ✕   Script Output ✕

📌 ✏ 💾 🖨 ☰ | Task completed in 0.08 seconds

```
1 rows inserted.
1 rows inserted.
1 rows inserted.
```

1B	**Insert** two rows of valid data into the **class** table according to the following. Make sure that you provide a value for every column.

```
INSERT INTO chave_class VALUES (0,'MATH 100');
INSERT INTO chave_class VALUES (1,'CSC 134');
```

Task completed in 0.03 seconds

```
1 rows inserted.
1 rows inserted.
```

1C	**Insert** two rows of valid data into the **student** table according to the following. Provide a value for every column.
	  For the 1st row:
	    Use the to_date function to insert into the DOB column in the format (yyddmm)
	  For the 2nd row
	    Use the default date format (Do not use to_date function) (use default)

```
INSERT INTO chave_student VALUES('123456789','Smith','John',(TO_DATE(602506,'YYDDMM')),15000);
INSERT INTO chave_student VALUES('132567586','Ross','Boss','01-FEB-1989',30000);
```


Task completed in 0.027 seconds

```
1 rows inserted.
1 rows inserted.
```

Do the questions in the order in which they appear. You may get error messages which of course is valid

2A	**Insert a** row of invalid data into student table such that it violates the primary key

```
INSERT INTO chave_student VALUES('13256758695834598','Trane','Thomas','10-FEB-1980',30000);
```


Task completed in 0.138 seconds

```
Error starting at line : 68 in command -
INSERT INTO chave_student VALUES('13256758695834598','Trane','Thomas','10-FEB-1980',30000)
Error report -
SQL Error: ORA-12899: value too large for column "CSUS"."CHAVE_STUDENT"."SSN" (actual: 17, maximum: 9)
12899. 00000 -  "value too large for column %s (actual: %s, maximum: %s)"
*Cause:    An attempt was made to insert or update a column with a value
           which is too wide for the width of the destination column.
           The name of the column is given, along with the actual width
           of the value, and the maximum allowed width of the column.
           Note that widths are reported in characters if character length
           semantics are in effect for the column, otherwise widths are
           reported in bytes.
*Action:   Examine the SQL statement for correctness.  Check source
           and destination column data types.
           Either make the destination column wider, or use a subset
           of the source column (i.e. use substring).
```

2B	**Insert a** row of invalid data into student table such that it violates the candidate key

```
INSERT INTO chave_student VALUES('135304389','Ross','Boss','10-FEB-1980',20000);
```
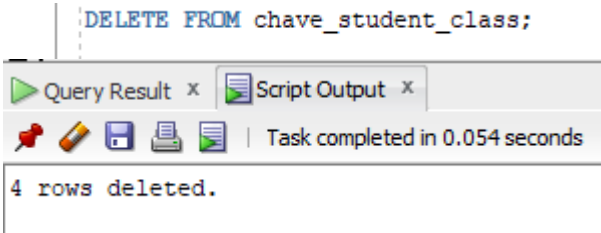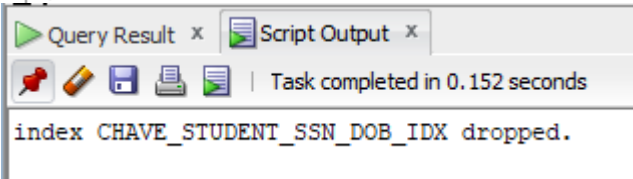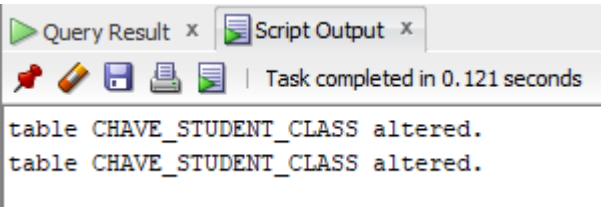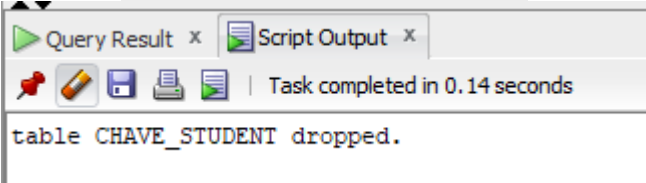
```
Query Result ×   Script Output ×
📌 ✏ 💾 🖨 📋 | Task completed in 0.081 seconds
Error starting at line : 72 in command -
INSERT INTO chave_student VALUES('135304389','Ross','Boss','10-FEB-1980',20000)
Error report -
SQL Error: ORA-00001: unique constraint (CSUS.CHAVE_STUDENT_LNAME_FNAME_UK) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:     An UPDATE or INSERT statement attempted to insert a duplicate key.
            For Trusted Oracle configured in DBMS MAC mode, you may see
            this message if a duplicate entry exists at a different level.
*Action:    Either remove the unique restriction or do not insert the key.
```

2C	**Insert a** row of invalid data into student table such that it violates a check constraint

```
INSERT INTO chave_student VALUES('132567582','Trane','Thomas','10-FEB-1980',9000);
```
```
Query Result ×   Script Output ×
📌 ✏ 💾 🖨 📋 | Task completed in 0.141 seconds
Error starting at line : 76 in command -
INSERT INTO chave_student VALUES('132567582','Trane','Thomas','10-FEB-1980',9000)
Error report -
SQL Error: ORA-02290: check constraint (CSUS.CHAVE_STUDENT_SALARY_CK) violated
02290. 00000 -  "check constraint (%s.%s) violated"
*Cause:     The values being inserted do not satisfy the named check

*Action:    do not insert values that violate the constraint.
```

2D	**Insert a** row of invalid data into the **student_class** table such that it violates the foreign key to the class table

```
INSERT INTO chave_student_class VALUES('123495334',0);
```
```
Query Result ×   Script Output ×
📌 ✏ 💾 🖨 📋 | Task completed in 0.13 seconds
Error starting at line : 80 in command -
INSERT INTO chave_student_class VALUES('123495334',0)
Error report -
SQL Error: ORA-02291: integrity constraint (CSUS.CHAVE_STUDENT_CLASS_SSN_FK) violated - parent key not found
02291. 00000 - "integrity constraint (%s.%s) violated - parent key not found"
*Cause:    A foreign key value has no matching primary key value.
*Action:   Delete the foreign key or add a matching primary key.
```

2E	**Insert a** row of invalid data into the **student_class** table such that it violates the foreign key to the student table

```
INSERT INTO chave_student_class VALUES('245306649',5);
```
```
Query Result ×   Script Output ×
📌 ✏ 💾 🖨 📋 | Task completed in 0.096 seconds
Error starting at line : 84 in command -
INSERT INTO chave_student_class VALUES('245306649',5)
Error report -
SQL Error: ORA-02291: integrity constraint (CSUS.CHAVE_STUDENT_CLASS_CLASS_CODE_FK) violated - parent key not found
02291. 00000 - "integrity constraint (%s.%s) violated - parent key not found"
*Cause:    A foreign key value has no matching primary key value.
*Action:   Delete the foreign key or add a matching primary key.
```

2F	**Insert a** row of invalid data into the **student_class** table such that it violates the primary key in the student_class table.

```
                         INSERT INTO chave_student_class VALUES('123456789',0);
```

Query Result × | Script Output ×

📌 🖊 💾 🖨 📇 | Task completed in 0.084 seconds

```
Error starting at line : 88 in command -
INSERT INTO chave_student_class VALUES('123456789',0)
Error report -
SQL Error: ORA-00001: unique constraint (CSUS.CHAVE_STUDENT_CLASS_SSN_CLASS_CODE_PK) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:     An UPDATE or INSERT statement attempted to insert a duplicate key.
            For Trusted Oracle configured in DBMS MAC mode, you may see
            this message if a duplicate entry exists at a different level.
*Action:    Either remove the unique restriction or do not insert the key.
```

Do the questions in the order in which they appear.You may get error messages which of course is valid

3A	**Disable** the foreign key constraint to the **student** table

	```ALTER TABLE chave_student_class DISABLE CONSTRAINT chave_student_class_ssn_fk;```
	
	Query Result × | Script Output ×
	
	📌 🖊 💾 🖨 📇 | Task completed in 0.031 seconds
	
	```table CHAVE_STUDENT_CLASS altered.```

3B	**Insert** a rows of data into the **student_class** table such that it would violate the foreign key constraint if the above constraint to the student table was enabled

	```INSERT INTO chave_student_class VALUES('987654321',0);```
	
	Query Result × | Script Output ×
	
	📌 🖊 💾 🖨 📇 | Task completed in 0.027 seconds
	
	```1 rows inserted.```

3C	**Enable** the foreign key constraint

	```ALTER TABLE chave_student_class ENABLE CONSTRAINT chave_student_class_ssn_fk;```
	
	Query Result × | Script Output ×
	
	📌 🖊 💾 🖨 📇 | Task completed in 0.061 seconds
	
	```Error starting at line : 99 in command -```
	```ALTER TABLE chave_student_class ENABLE CONSTRAINT chave_student_class_ssn_fk```
	```Error report -```
	```SQL Error: ORA-02298: cannot validate (CSUS.CHAVE_STUDENT_CLASS_SSN_FK) - parent keys not found```
	```02298. 00000 - "cannot validate (%s.%s) - parent keys not found"```
	```*Cause: an alter table validating constraint failed because the table has```
	``` child records.```
	```*Action: Obvious```

Do the questions in the order in which they appear. You may get error messages which of course
is valid

4A	**Delete** the data from the student_class table
	DELETE FROM chave_student_class; Query Result ✕ Script Output ✕ Task completed in 0.054 seconds 4 rows deleted.
4B	**Truncate** the student table
	TRUNCATE TABLE chave_student; table CHAVE_STUDENT truncated.
4C	**drop** the index that you created on the student table
	DROP INDEX chave_student_ssn_dob_idx; Query Result ✕ Script Output ✕ Task completed in 0.152 seconds index CHAVE_STUDENT_SSN_DOB_IDX dropped.
4D	**Drop** the foreign key constraints
	ALTER TABLE chave_student_class DROP CONSTRAINT chave_student_class_ssn_fk; ALTER TABLE chave_student_class DROP CONSTRAINT chave_student_class_class_code_fk; Query Result ✕ Script Output ✕ Task completed in 0.121 seconds table CHAVE_STUDENT_CLASS altered. table CHAVE_STUDENT_CLASS altered.
4E	**Drop** the student table.
	DROP TABLE chave_student; Query Result ✕ Script Output ✕ Task completed in 0.14 seconds table CHAVE_STUDENT dropped.