ASSIGNMENT 5
Create, Alter

In this lab you will use SQL statements that fall in both the DDL and DML category. In the previous labs you were retrieving information that was already stored in the database. In this lab you will be storing new information in the database.

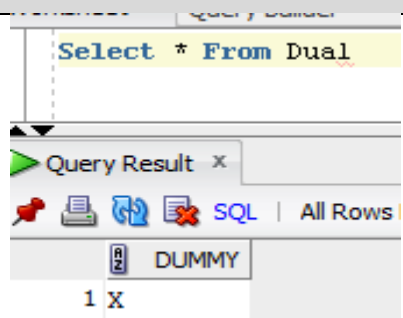**You must execute the statements in the order in which the questions are being asked.**

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
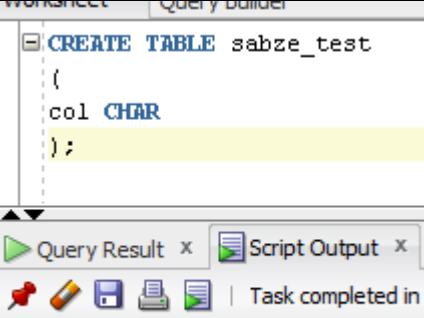
All the tables that you create should be prefixed with the first five letters of your lastname such as **sabze_patient**

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. <u>Do not assume that just because you are not getting an error message, everything is okay.</u>
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.
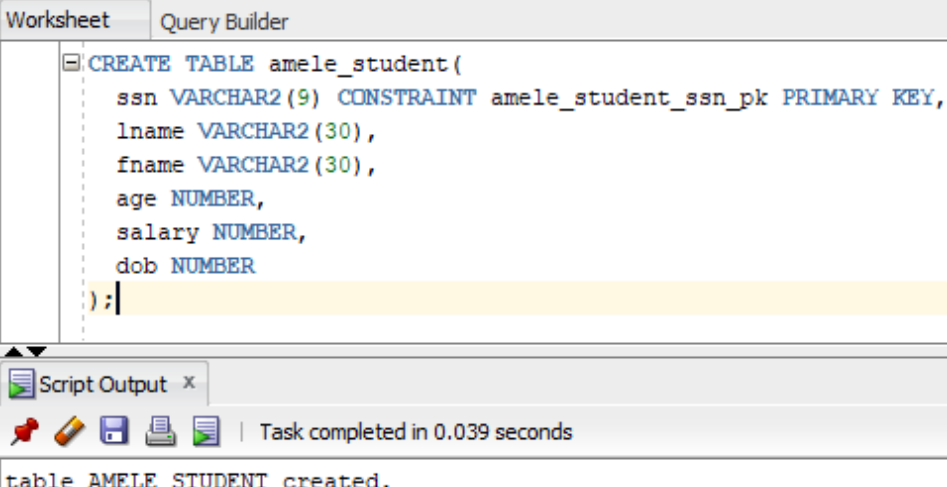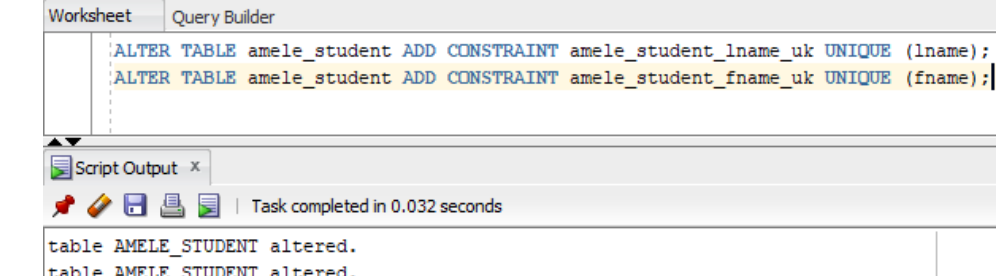5) Make sure that you only provide a printscreen of the snippet that pertains to the question (NOTHING MORE).

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.**

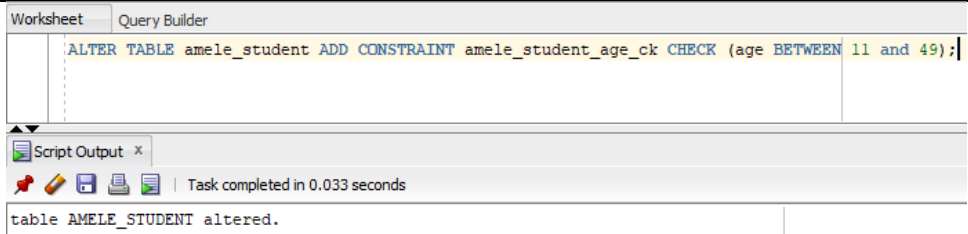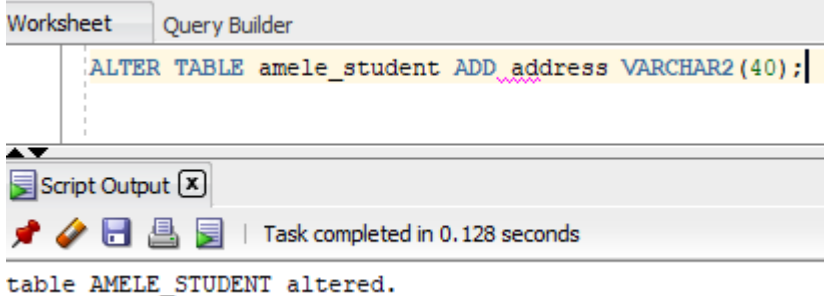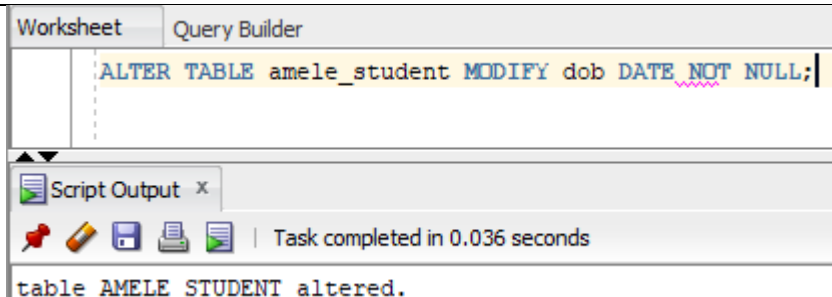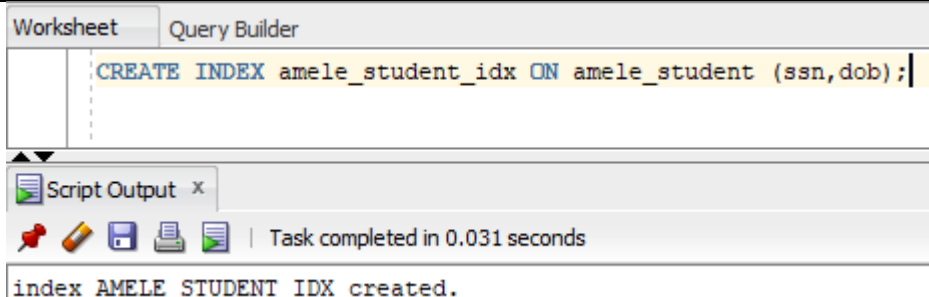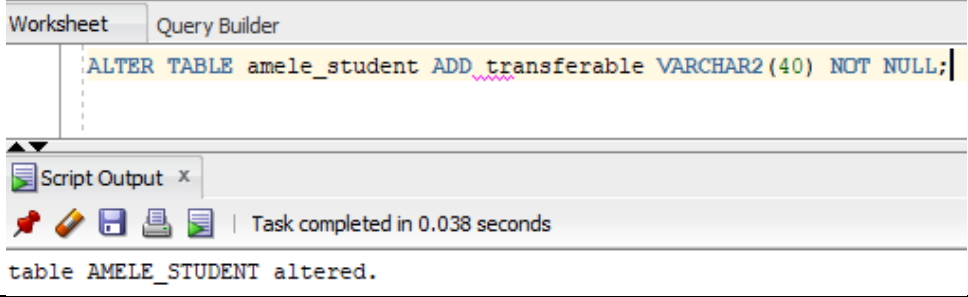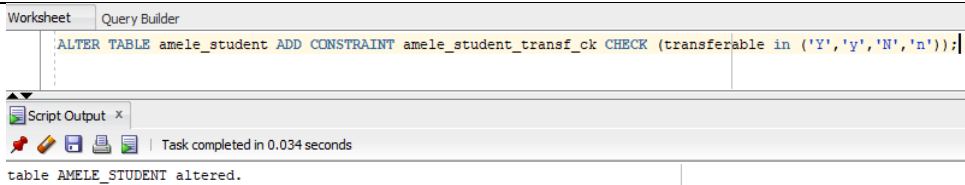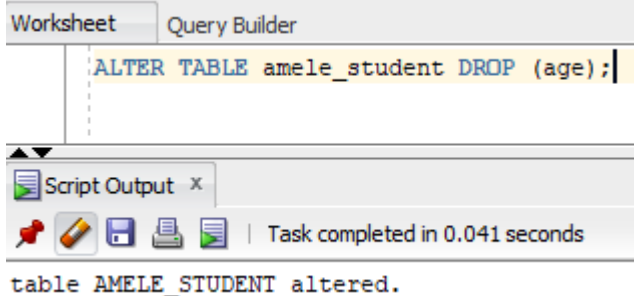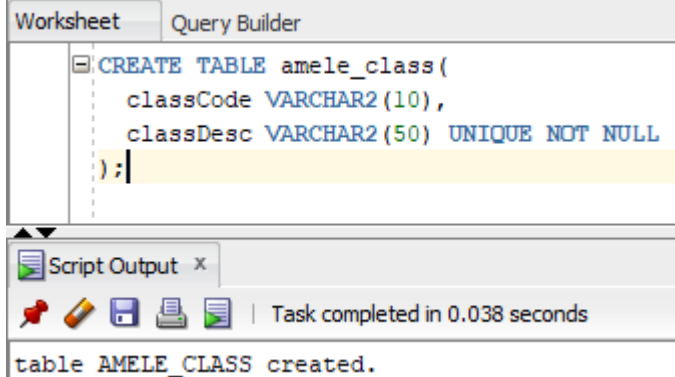|	SQLPlus or SQLDeveloper (Your choice)
Example	Display the contents of the dual table
	```SQL> SELECT * FROM dual; D - X 1 row selected.``` **OR**
Next Example	Create a table called test
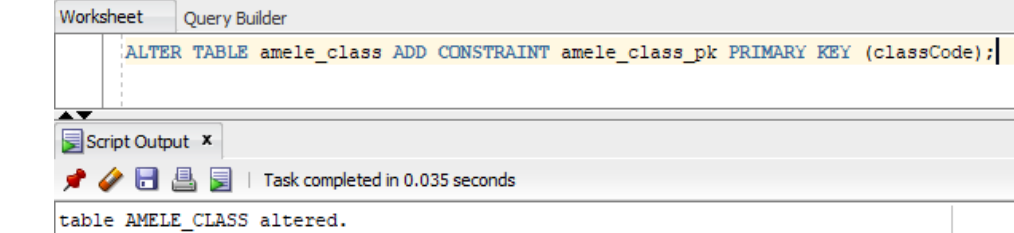
```
SQL> CREATE TABLE sabze_test
  2 (
  3 col    CHAR
  4 );

Table created.
```

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

Query Result ×   Script Output ×

📌 🖉 🖫 🖨 🗐 | Task completed in

table SABZE_TEST created.

OR

All the tables that you create must be prefixed with the first five letters of your last name such as sabze_student.

1A	Create a **student** table that will hold the following data. Make sure you assign a **primary key**. SSN text lname text fname text age numeric salary numeric dob numeric

	Worksheet Query Builder
	
	 ```CREATE TABLE amele_student(```
	 ``` ssn VARCHAR2(9) CONSTRAINT amele_student_ssn_pk PRIMARY KEY,```
	 ``` lname VARCHAR2(30),```
	 ``` fname VARCHAR2(30),```
	 ``` age NUMBER,```
	 ``` salary NUMBER,```
	 ``` dob NUMBER```
	 ```);```
	
	 Script Output ×
	 📌 🖉 🖫 🖨 🗐
		Task completed in 0.039 seconds
		
		 table AMELE_STUDENT created.

1B	**After** the table has been created add a **candidate key** based on lname and fname. **Note: Candidate and unique key are the same thing**

	Worksheet Query Builder
	
	 ```ALTER TABLE amele_student ADD CONSTRAINT amele_student_lname_uk UNIQUE (lname);```
	 ```ALTER TABLE amele_student ADD CONSTRAINT amele_student_fname_uk UNIQUE (fname);```
	
	 Script Output ×
	 📌 🖉 🖫 🖨 🗐
		Task completed in 0.032 seconds
		
		 table AMELE_STUDENT altered.
		 table AMELE_STUDENT altered.

1C	**After** the table has been created add a **check constraint** such that the age is greater than 10 but less than 50. Provide a name for the check constraint.

| | ```
Worksheet    Query Builder
    ALTER TABLE amele_student ADD CONSTRAINT amele_student_age_ck CHECK (age BETWEEN 11 and 49);

Script Output  x
Task completed in 0.033 seconds
table AMELE_STUDENT altered.
``` |
| **1D** | **After** the table has been created **add a column** called address. |
| | ```
Worksheet    Query Builder
    ALTER TABLE amele_student ADD address VARCHAR2(40);

Script Output  X
Task completed in 0.128 seconds
table AMELE_STUDENT altered.
``` |
| **1E** | **After** the table has been created, modify the dob column to be of datatype **date** and also **not null** |
| | ```
Worksheet    Query Builder
    ALTER TABLE amele_student MODIFY dob DATE NOT NULL;

Script Output  x
Task completed in 0.036 seconds
table AMELE_STUDENT altered.
``` |
| **1F** | Create a composite **index on** ssn and dob |
| | ```
Worksheet    Query Builder
    CREATE INDEX amele_student_idx ON amele_student (ssn,dob);

Script Output  x
Task completed in 0.031 seconds
index AMELE_STUDENT_IDX created.
``` |
| **1G** | After the table has been created **add a column** called **transferable** with a **not null** constraint. Do not assign a name to the constraint |
| | ```
Worksheet    Query Builder
    ALTER TABLE amele_student ADD transferable VARCHAR2(40) NOT NULL;

Script Output  x
Task completed in 0.038 seconds
table AMELE_STUDENT altered.
``` |
| **1H** | After the table has been created add a **check constraint** on the column t**ransferable** to allow only 'y','Y','n','N'. Give the constraint a name. |

```
ALTER TABLE amele_student ADD CONSTRAINT amele_student_transf_ck CHECK (transferable in ('Y','y','N','n'));
```

Script Output x

Task completed in 0.034 seconds

table AMELE_STUDENT altered.

1I	Drop the age column

Worksheet   Query Builder

```
ALTER TABLE amele_student DROP (age);
```

Script Output x

Task completed in 0.041 seconds

table AMELE_STUDENT altered.

2A	Create a second table called **class** that will hold the following data. You decide what the data types are going to be.
	Class code Class description
	In the create table statement make **Class description the candidate key** and also make it is **not null. NOTE: Candidate and unique keys are the same thing**

Worksheet   Query Builder

```
CREATE TABLE amele_class(
    classCode VARCHAR2(10),
    classDesc VARCHAR2(50) UNIQUE NOT NULL
);
```

Script Output x

Task completed in 0.038 seconds

table AMELE_CLASS created.

2B	**After** the table has been created add the **primary key**. Give the constraint a name

Worksheet   Query Builder

```
ALTER TABLE amele_class ADD CONSTRAINT amele_class_pk PRIMARY KEY (classCode);
```

Script Output x

Task completed in 0.035 seconds

table AMELE_CLASS altered.

2C	Create an **index** on **class description**

```
Worksheet    Query Builder
      CREATE INDEX amele_class_idx ON amele_class (classDesc);
```

```
Script Output ×   Query Result ×
📌 🖊 💾 🖨 📧 │ Task completed in 0.09 seconds

Error starting at line : 1 in command -
CREATE INDEX amele_class_idx ON amele_class (classDesc)
Error at Command Line : 1 Column : 46
Error report -
SQL Error: ORA-01408: such column list already indexed
01408. 00000 -  "such column list already indexed"
*Cause:
*Action:
```

3A	Create a third table called **student_class.** This table is an association table that contains information on the different class that the students are taking. You figure out what the columns should be. It should contain only two columns.

	``
	Worksheet Query Builder
	   CREATE TABLE amele_student_class(
	      ssn VARCHAR2(9),
	      classCode VARCHAR2(10)
	   );
	``
	
	``
	Script Output × Query Result ×
	📌 🖊 💾 🖨 📧 │ Task completed in 0.042 seconds
	
	table AMELE_STUDENT_CLASS created.
	``

3B	**After** the table has been created add the **primary key** constraint (Name the constraint)
	``` Worksheet Query Builder ALTER TABLE amele_student_class ADD CONSTRAINT amele_student_class_pk PRIMARY KEY (ssn,classcode); ``` ``` Script Output × Query Result × 📌 🖊 💾 🖨 📧 │ Task completed in 0.037 seconds table AMELE_STUDENT_CLASS altered. ```
3C	**After** the table has been created add the **foreign key** constraint(**s**) (Name the constraint(s))

	``
	Worksheet Query Builder
	   ALTER TABLE amele_student_class ADD CONSTRAINT amele_student_class_st_fk FOREIGN KEY (ssn) REFERENCES amele_student;
	   ALTER TABLE amele_student_class ADD CONSTRAINT amele_student_class_cl_fk FOREIGN KEY (classCode) REFERENCES amele_class;
	``
	
	``
	Script Output × Query Result ×
	📌 🖊 💾 🖨 📧 │ Task completed in 0.036 seconds
	
	table AMELE_STUDENT_CLASS altered.
	table AMELE_STUDENT_CLASS altered.
	``