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**
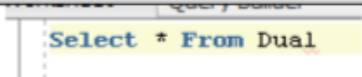
What to turn in:
1) You will turn in this word document only. I do not want any other files
2) Paste a printscreen of either the **SQLPlus session** or **SQL Developer** showing only the SQL command and the results from the database engine. Some of the SQL statements that you issue may cause an error and may actually be the expected result. 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
	Select * From Dual
	SQL> SELECT * FROM dual; D - X 1 row selected.　　　OR	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
);
```
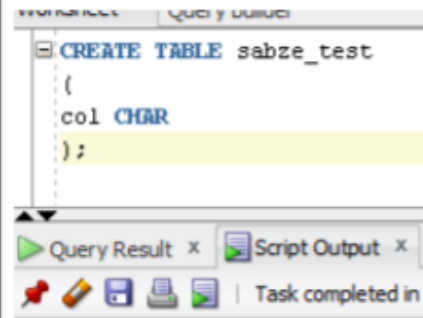
Query Result x | Script Output x
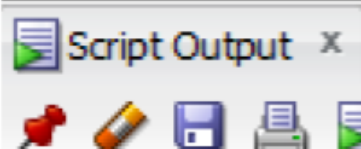
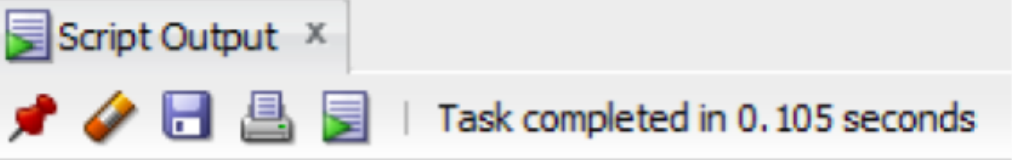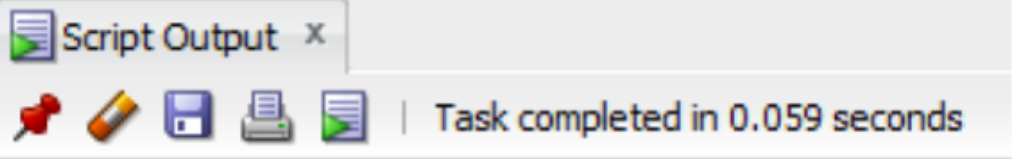Task completed in

```
table SABZE_TEST created.
```

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

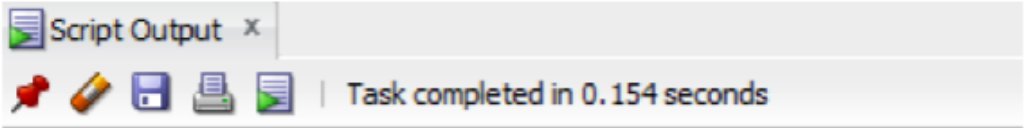| 1A	Create a **student** table that will hold the following data. Make sure you assign a **primary key**.
	SSN text
	lname text
	fname text
	age numeric
	salary numeric
	dob numeric

```
CREATE TABLE hawj_student
(
  SSN CHAR(9) PRIMARY KEY,
  lname VARCHAR(20),
  fname VARCHAR(20),
  age NUMBER,
  salary NUMBER,
  dob NUMBER
);
```

Script Output X

Task completed in 0.057 seconds

```
table HAWJ_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 |

```sql
ALTER TABLE hawj_student ADD UNIQUE (lname, fname);
```

Script Output X

Task completed in 0.053 seconds

```
table HAWJ_STUDENT created.
table HAWJ_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. |

```sql
ALTER TABLE hawj_student ADD CONSTRAINT hawj_student_age_ck CHECK (age>10 and age<50);
```

```
table HAWJ_STUDENT altered.
```

| 1D | **After** the table has been created **add a column** called address. |

```sql
ALTER TABLE hawj_student ADD address VARCHAR(20);
```

Script Output X

Task completed in 0.049 seconds

```
table HAWJ_STUDENT altered.
```

| 1E | **After** the table has been created, modify the dob column to be of datatype **date** and also **not null** |

```sql
ALTER TABLE hawj_student MODIFY dob DATE NOT NULL;
```

```
table HAWJ_STUDENT altered.
```

| 1F | Create a composite **index on** ssn and dob |

```sql
CREATE INDEX hawj_student_idx ON hawj_student (SSN, dob);
```

```
index HAWJ_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 |

```
ALTER TABLE hawj_student ADD transferable CHAR(1) NOT NULL;

table HAWJ_STUDENT altered.
```

1H	After the table has been created add a **check constraint** on the column **transferable** to allow only 'y','Y','n','N'. Give the constraint a name.

```
ALTER TABLE hawj_student
ADD CONSTRAINT hawj_student_transferable_ck CHECK (transferable IN ('y', 'Y', 'n', 'N'));
```

Script Output x

Task completed in 0.105 seconds

```
table HAWJ_STUDENT altered.
```

1I	Drop the age column

```
ALTER TABLE hawj_student DROP COLUMN age;
```

Script Output x

Task completed in 0.059 seconds

```
table HAWJ_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**

```
CREATE TABLE hawj_class
(
  code NUMBER,
  description VARCHAR(255) NOT NULL UNIQUE
);

table HAWJ_CLASS created.
```

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

```
ALTER TABLE hawj_class ADD CONSTRAINT hawj_class_pk PRIMARY KEY (code);

table HAWJ_CLASS altered.
```

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

```
CREATE INDEX hawj_class_idx ON hawj_class(description);
```

Script Output ×

Task completed in 0.154 seconds

```
Error starting at line : 36 in command -
CREATE INDEX hawj_class_idx ON hawj_class(description)
Error at Command Line : 36 Column : 43
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.

```
CREATE TABLE hawj_student_class
(
    SSN CHAR(9),
    code NUMBER
);
table HAWJ_STUDENT_CLASS created.
```

3B	**After** the table has been created add the **primary key** constraint (Name the constraint)

```
ALTER TABLE hawj_student_class ADD CONSTRAINT hawj_student_class_pk PRIMARY KEY (SSN);

table HAWJ_STUDENT_CLASS altered.
```

3C	**After** the table has been created add the **foreign key** constraint(s) (Name the constraint(s))

```sql
ALTER TABLE hawj_student_class
ADD CONSTRAINT hawj_student_class_ssn_fk
FOREIGN KEY (SSN) REFERENCES hawj_student;

ALTER TABLE hawj_student_class
ADD CONSTRAINT hawj_student_class_code_fk
FOREIGN KEY (code) REFERENCES hawj_class;
```

Script Output ×

Task completed in 0.051 seconds

table HAWJ_STUDENT_CLASS altered.
table HAWJ_STUDENT_CLASS altered.