ASSIGNMENT 7
SELECT

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

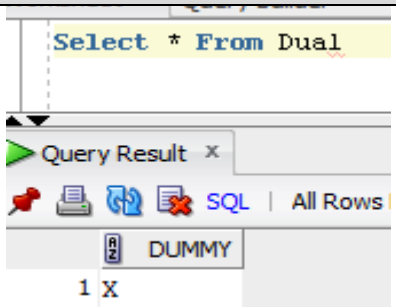
All the tables that you create should be prefixed with the first five letters of your lastname such as **sabze_patient**

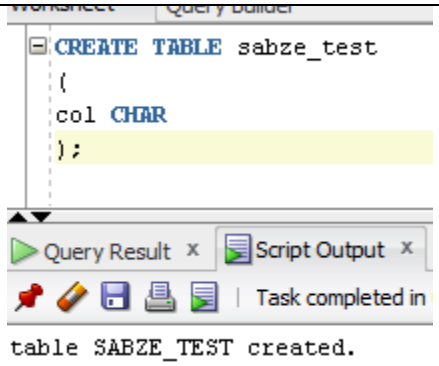
What to turn in:

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 
Next Example	Create a table called test

	SQL> CREATE TABLE sabze_test 2 (3 col CHAR 4); Table created. OR 

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

0	**Copy and paste the contents of student.txt into your SQLPlus or SQLDeveloper session. Rename the tables such that they are all prefixed with the first five letters of your lastname such as sabze_student. Make sure that the tables (student, class and student_class) are all renamed properly before you continue.**
1	Using a single SQL statement display fname, lname, dob, salary for all the **students** whose age is greater than 15. (Have to convert the dob to years)
	`SELECT fname,lname,dob,salary` `FROM student` `WHERE TRUNC(MONTHS_BETWEEN(SYSDATE,DOB)/12) > 15;`

	FNAME	LNAME	DOB	SALARY
1	Abraham	Bennet	26-FEB-88	10000
2	Marjorie	Green	25-FEB-89	20000
3	Albert	Greeenr	24-FEB-92	15000
4	Ann	Dull	23-FEB-93	30000
5	Akiko	Yokomoto	22-FEB-94	35000
6	Michael	O'Leary	12-FEB-95	32000
7	Burt	Gringlesby	12-FEB-96	34000
8	Morningstar	Greene	11-FEB-95	25000
9	Cal	Al	06-FEB-98	22000
10	Johnson	White	05-FEB-99	23000
11	Innes	del Castillo	04-FEB-82	23500
12	Sheryl	Hunter	03-FEB-79	18000
13	Chastity	Locksley	02-FEB-78	15500
14	Reginald	Blotchet-Halls	01-FEB-77	43000

2 Using a single SQL statement display the following from the **student** table.
ssn, lname and fname concatenated together with a comma and a space separating the two **(e.g sabzevary, IRAJ).** The last name should be all lower case. The first name should be all upper case. The heading on the column should be **Full_Name (Use the concat function or the || symbols)**

```
SELECT ssn||', '||LOWER(lname)||', '||UPPER(fname) AS Full_Name
FROM student;
```

```
1  999-00-0000, al, CAL
2  409-56-7008, bennet, ABRAHAM
3  648-92-1872, blotchet-halls, REGINALD
4  427-17-2319, dull, ANN
5  998-72-3567, greeenr, ALBERT
6  213-46-8915, green, MARJORIE
7  527-72-3246, greene, MORNINGSTAR
8  238-95-7766, gren, CHERYL
9  472-27-2349, gringlesby, BURT
10 846-92-7186, hunter, SHERYL
11 486-29-1786, locksley, CHASTITY
12 267-41-2394, o'leary, MICHAEL
13 172-32-1176, white, JOHNSON
14 672-71-3249, yokomoto, AKIKO
15 712-45-1867, del castillo, INNES
```

3 Using a single SQL statement display fname, lname, dob, salary from the **student** table where the lname contains the letters 'h' or 'a' regardless of case (Use the like clause)

```
SELECT fname,lname,dob,salary
FROM student
WHERE lname LIKE '%h%' OR lname LIKE '%a%';
```

1	Michael	O'Leary	12-FEB-95	32000
2	Johnson	White	05-FEB-99	23000
3	Innes	del Castillo	04-FEB-82	23500
4	Reginald	Blotchet-Halls	01-FEB-77	43000

4 Using a single SQL statement display fname, lname, dob, salary from the **student** table where age is between 15 and 25. (use the between clause) and fname starts with 'abr' regardless of case. If the dob is null, display 'not born yet' (USE NVL)

```
SELECT fname,lname,NVL(TO_CHAR(dob),'NOT BORN YET'),salary
FROM student
WHERE lname LIKE 'abr%' AND ROUND(MONTHS_BETWEEN(SYSDATE,DOB)/12) BETWEEN 15 AND 25;
```

5 Using a single SQL statement display fname, lname, dob, salary from the **student** table where the dob is not null. If the salary is <20000 display 'poor' otherwise display 'rich' (Use decode)

```
SELECT fname,lname,dob,salary, DECODE (SIGN(salary-20000),1,'RICH',-1,'POOR')
FROM student
WHERE DOB IS NOT NULL;
```

1	Abraham	Bennet	26-FEB-88	10000	POOR
2	Marjorie	Green	25-FEB-89	20000	(null)
3	Albert	Greeenr	24-FEB-92	15000	POOR
4	Ann	Dull	23-FEB-93	30000	RICH
5	Akiko	Yokomoto	22-FEB-94	35000	RICH
6	Michael	O'Leary	12-FEB-95	32000	RICH
7	Burt	Gringlesby	12-FEB-96	34000	RICH
8	Morningstar	Greene	11-FEB-95	25000	RICH
9	Cal	Al	06-FEB-98	22000	RICH
10	Johnson	White	05-FEB-99	23000	RICH
11	Innes	del Castillo	04-FEB-82	23500	RICH
12	Sheryl	Hunter	03-FEB-79	18000	POOR
13	Chastity	Locksley	02-FEB-78	15500	POOR
14	Reginald	Blotchet-Halls	01-FEB-77	43000	RICH

6 Using a single SQL statement display the square root of dob plus 20 divided by 5 from the **student** table(CAUTION: The order of precedence is as the question is read. Use paranthesis) (Have to convert dob to years first)

```
SELECT dob, (SQRT(ROUND(MONTHS_BETWEEN(SYSDATE,DOB)/12))+20)/5
FROM student;
```

	DOB	(SQRT(ROUND(MONTHS_BETWEEN(SYSDATE,DOB)/12))+20)/5
1	26-FEB-88	5.113552872566004384423894259783709904 1
2	25-FEB-89	5.095445115010332226913939565601604267 91
3	(null)	(null)
4	24-FEB-92	5.039230484541326376116467804903523420 17
5	23-FEB-93	5.019803902718556966005644821804556397 91
6	22-FEB-94	5
7	12-FEB-95	4.979795897113271239278913629882356556 79
8	12-FEB-96	4.959166304662543908319487612832538784
9	11-FEB-95	4.979795897113271239278913629882356556 79
10	06-FEB-98	4.916515138991168001317609438745601697 8
11	05-FEB-99	4.894427190999915878563669467492510494 18
12	04-FEB-82	5.216552506059643937799936849040413412 42
13	03-FEB-79	5.264911064067351732799557417773087413 49
14	02-FEB-78	5.280624847486569737297643534924362652 9
15	01-FEB-77	5.296148139681572046193193487217599331 54

7 Using a single SQL statement display fname, lname, dob, salary from the **student** table where the first name of the student can be **anything except** John, Jack or Bob. (Use the IN or NOT IN syntax)

```
SELECT fname,lname,dob,salary
FROM student
WHERE fname NOT IN ('John','Jack','Bob');
```

FNAME	LNAME	DOB	SALARY
Abraham	Bennet	26-FEB-88	10000
Marjorie	Green	25-FEB-89	20000
Cheryl	Gren		45000
Albert	Greeenr	24-FEB-92	15000
Ann	Dull	23-FEB-93	30000
Akiko	Yokomoto	22-FEB-94	35000
Michael	O'Leary	12-FEB-95	32000
Burt	Gringlesby	12-FEB-96	34000
Morningstar	Greene	11-FEB-95	25000
Cal	Al	06-FEB-98	22000
Johnson	White	05-FEB-99	23000
Innes	del Castillo	04-FEB-82	23500
Sheryl	Hunter	03-FEB-79	18000
Chastity	Locksley	02-FEB-78	15500
Reginald	Blotchet-Halls	01-FEB-77	43000

8 Using a single SQL statement display fname, lname, dob, salary from the **student** table where the fname is only three characters long; the first character and second characters can be anything, but the third character must be 'b' (e.g. bob, cib, lib, hub, mob). Also the salary must be greater than 10000 and the phone number must start with '527'

```
SELECT SUBSTR(fname,1,3),lname,dob,salary
FROM student
WHERE fname LIKE '  b%' AND salary>10000 AND phone like'527%';
```

9 Create a new table called student2 that contains the results from the following SQL statement: fname, lname, salary*2 from the **student** table where last name contains the letters 'nn' (e.g. Benny, Bonny, Sonny) and dob does not contain any

	data. (NOTE: Beware of salary*2 for the create table statement)
	`` CREATE TABLE student2 AS SELECT fname,lname,salary*2 new_salary FROM student WHERE lname LIKE '%nn%' AND DOB=NULL; ```