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

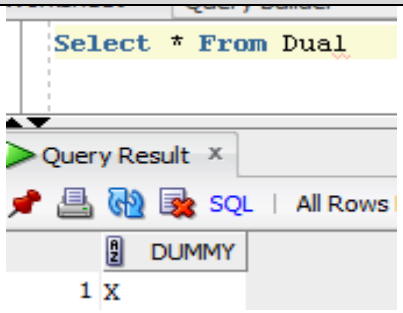
All the tables that you create should be prefixed with the first five letters of your lastname such as **sabze_patient**

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

```
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 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)

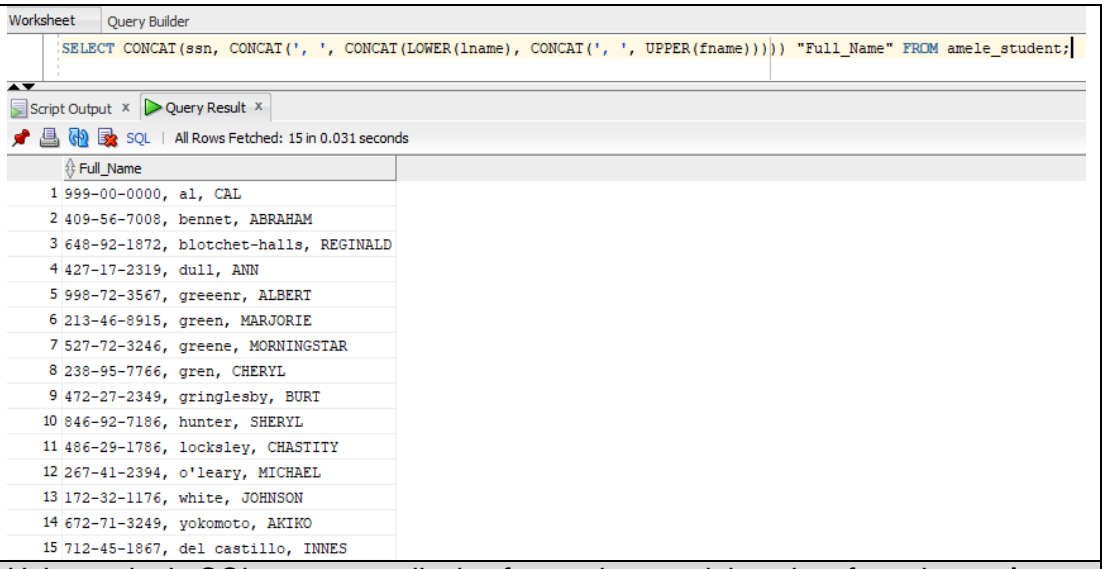
Worksheet | Query Builder

```
SELECT fname, lname, dob, salary FROM amele_student WHERE TRUNC(MONTHS_BETWEEN(sysdate,dob)/12) > 15;
```

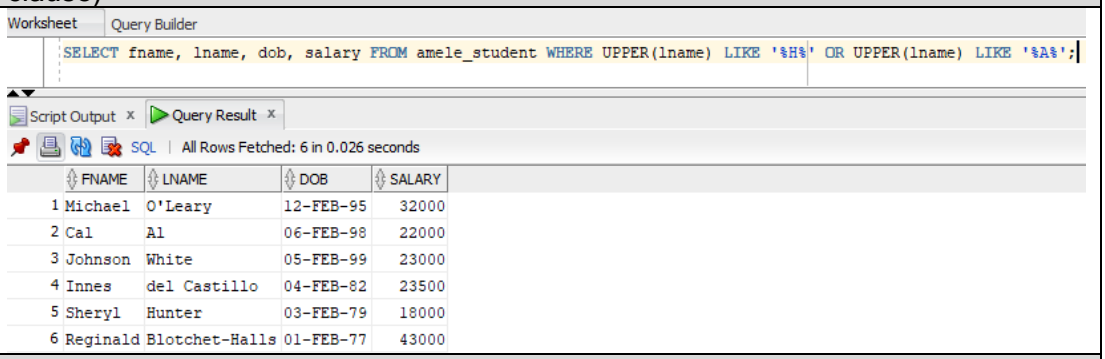
Script Output | Query Result | SQL | All Rows Fetched: 14 in 0.042 seconds

	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

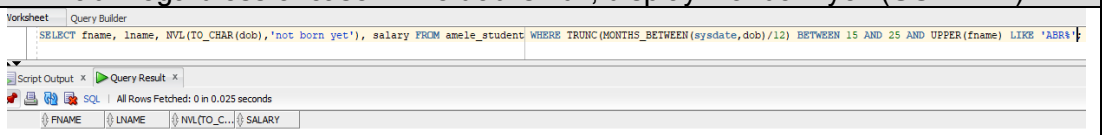
2	Using a single SQL statement display the following from the **student** table. *ssn, lname and fname* concatenated together with a comma and a space separating the two **(e.g sabzevary, IRAJ).** The last name should be all lower case. The first name should be all upper case. The heading on the column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



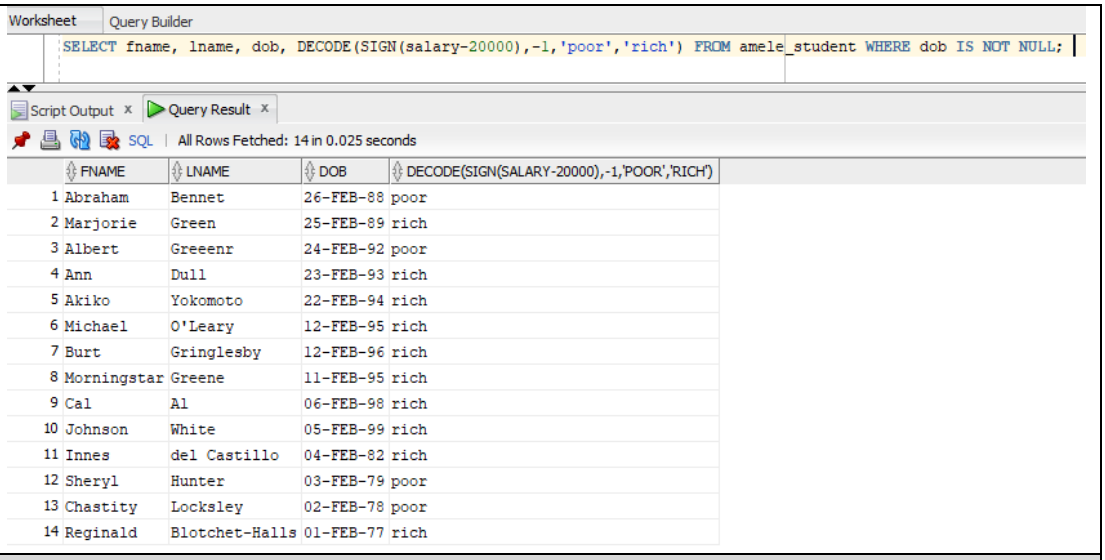
3	Using a single SQL statement display fname, lname, dob, salary from the **student** table where the lname contains the letters 'h' or 'a' regardless of case (Use the like clause)



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)

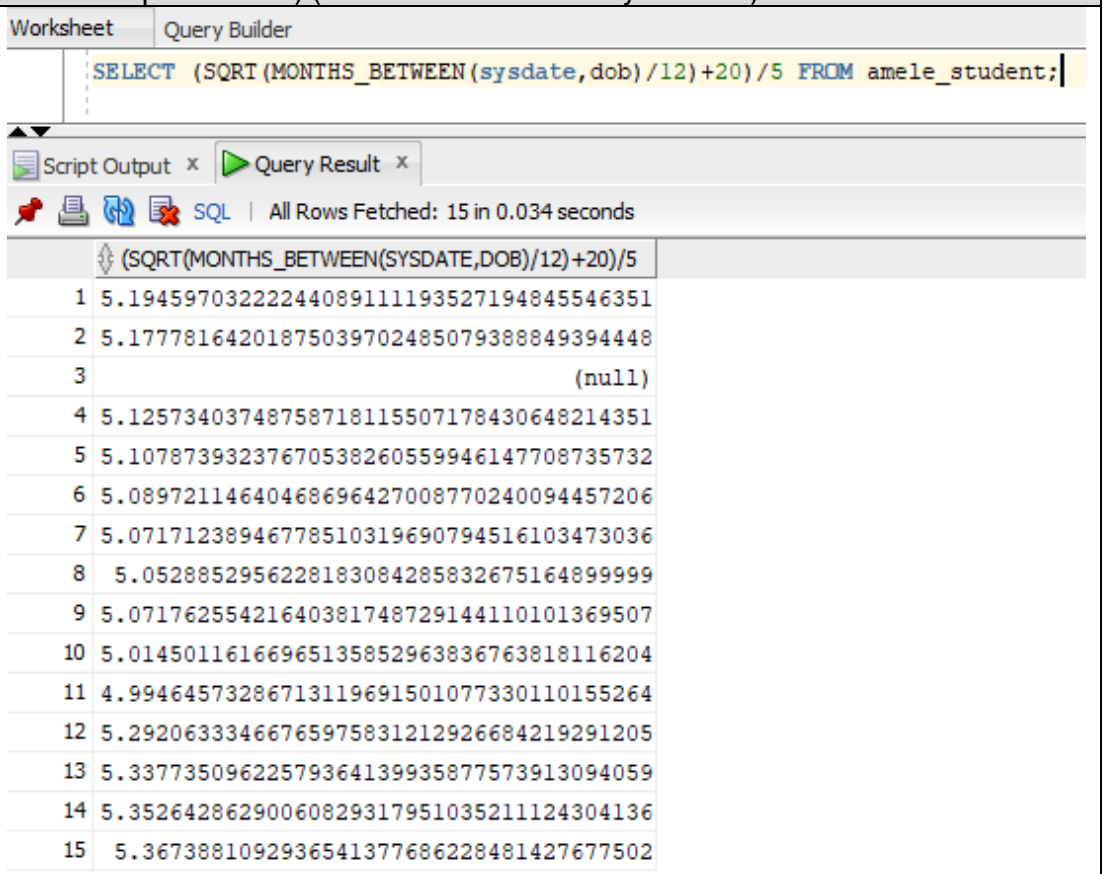


5	Using a single SQL statement display fname, lname, dob, salary from the **student** table where the dob is not null. If the salary is <20000 display 'poor' otherwise display 'rich' (Use decode)



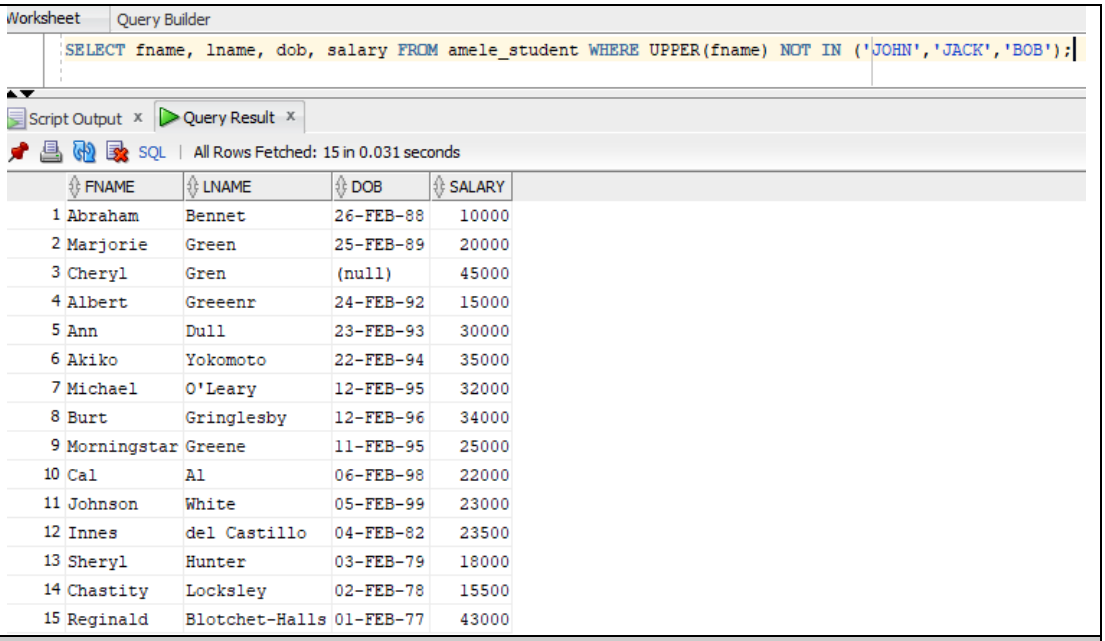
	FNAME	LNAME	DOB	DECODE(SIGN(SALARY-20000),-1,'POOR','RICH')
1	Abraham	Bennet	26-FEB-88	poor
2	Marjorie	Green	25-FEB-89	rich
3	Albert	Greeenr	24-FEB-92	poor
4	Ann	Dull	23-FEB-93	rich
5	Akiko	Yokomoto	22-FEB-94	rich
6	Michael	O'Leary	12-FEB-95	rich
7	Burt	Gringlesby	12-FEB-96	rich
8	Morningstar	Greene	11-FEB-95	rich
9	Cal	Al	06-FEB-98	rich
10	Johnson	White	05-FEB-99	rich
11	Innes	del Castillo	04-FEB-82	rich
12	Sheryl	Hunter	03-FEB-79	poor
13	Chastity	Locksley	02-FEB-78	poor
14	Reginald	Blotchet-Halls	01-FEB-77	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)



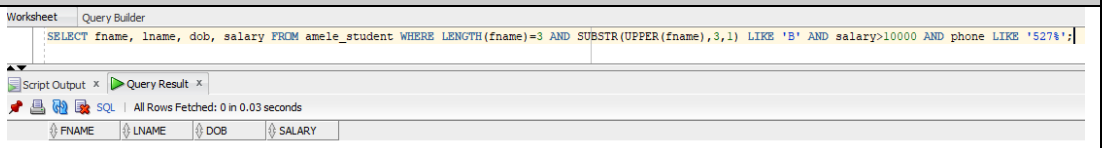
	(SQRT(MONTHS_BETWEEN(SYSDATE,DOB)/12)+20)/5
1	5.194597032222440891111935271948455463 51
2	5.177781642018750397024850793888493944 48
3	(null)
4	5.125734037487587181155071784306482143 51
5	5.107873932376705382605599461477087357 32
6	5.089721146404686964270087702400944572 06
7	5.071712389467785103196907945161034730 36
8	5.052885295622818308428583267516489999 9
9	5.071762554216403817487291441101013695 07
10	5.014501161669651358529638367638181162 04
11	4.994645732867131196915010773301101552 64
12	5.292063334667659758312129266842192912 05
13	5.337735096225793641399358775739130940 59
14	5.352642862900608293179510352111243041 36
15	5.367388109293654137768622848142767750 2

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)



	FNAME	LNAME	DOB	SALARY
1	Abraham	Bennet	26-FEB-88	10000
2	Marjorie	Green	25-FEB-89	20000
3	Cheryl	Gren	(null)	45000
4	Albert	Greeenr	24-FEB-92	15000
5	Ann	Dull	23-FEB-93	30000
6	Akiko	Yokomoto	22-FEB-94	35000
7	Michael	O'Leary	12-FEB-95	32000
8	Burt	Gringlesby	12-FEB-96	34000
9	Morningstar	Greene	11-FEB-95	25000
10	Cal	Al	06-FEB-98	22000
11	Johnson	White	05-FEB-99	23000
12	Innes	del Castillo	04-FEB-82	23500
13	Sheryl	Hunter	03-FEB-79	18000
14	Chastity	Locksley	02-FEB-78	15500
15	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'



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)