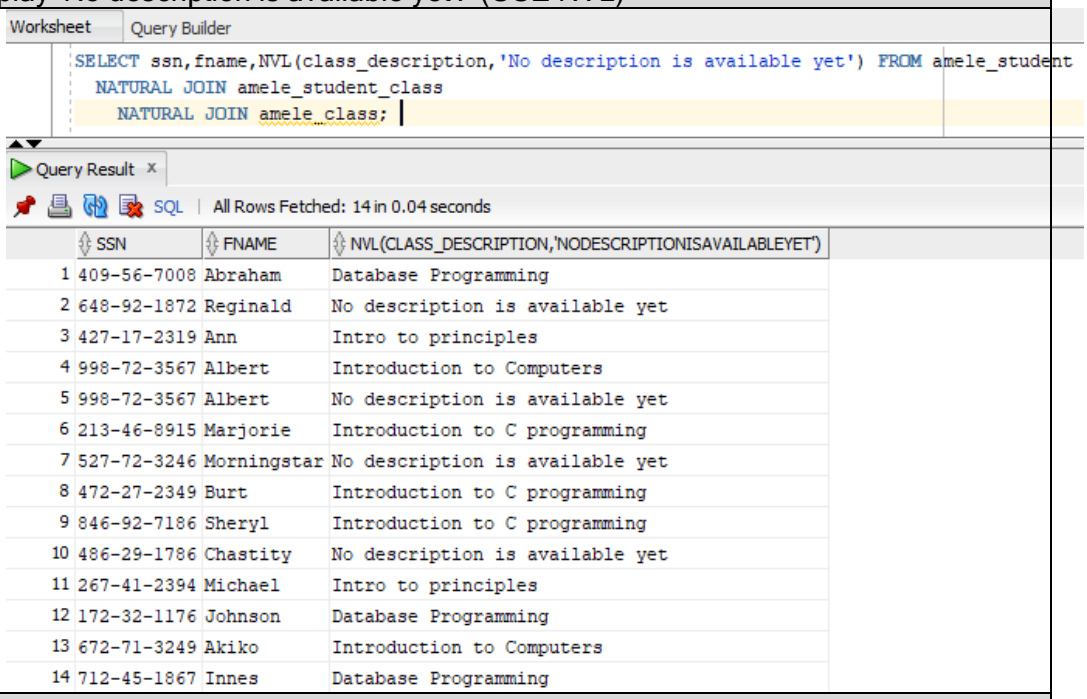
Follow the same formatting guidelines as the previous homework assignment.

Copy and paste the contents of student.txt (Same as the previous lab) into your SQLPlus 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, classes and student_class) are all renamed properly before you continue.

Use only a single SQL statement for each of the following questions

1 Give a listing of all the ssns,first names and the class descriptions of all the classes the students are taking. If there are no class _descriptions display 'No description is available yet'. (USE NVL)



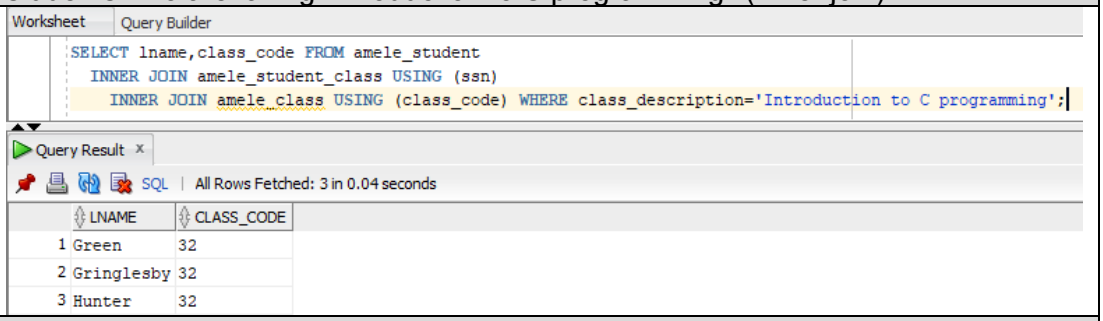
Worksheet | Query Builder

```
SELECT ssn,fname,NVL(class_description,'No description is available yet') FROM amele_student
  NATURAL JOIN amele_student_class
    NATURAL JOIN amele_class;
```

Query Result | SQL | All Rows Fetched: 14 in 0.04 seconds

	SSN	FNAME	NVL(CLASS_DESCRIPTION,'NODESCRIPTIONISAVAILABLEYET')
1	409-56-7008	Abraham	Database Programming
2	648-92-1872	Reginald	No description is available yet
3	427-17-2319	Ann	Intro to principles
4	998-72-3567	Albert	Introduction to Computers
5	998-72-3567	Albert	No description is available yet
6	213-46-8915	Marjorie	Introduction to C programming
7	527-72-3246	Morningstar	No description is available yet
8	472-27-2349	Burt	Introduction to C programming
9	846-92-7186	Sheryl	Introduction to C programming
10	486-29-1786	Chastity	No description is available yet
11	267-41-2394	Michael	Intro to principles
12	172-32-1176	Johnson	Database Programming
13	672-71-3249	Akiko	Introduction to Computers
14	712-45-1867	Innes	Database Programming

2 Give a listing of only the lname and the class_code for students who are taking 'Introduction to C programming'. (Inner join)



Worksheet | Query Builder

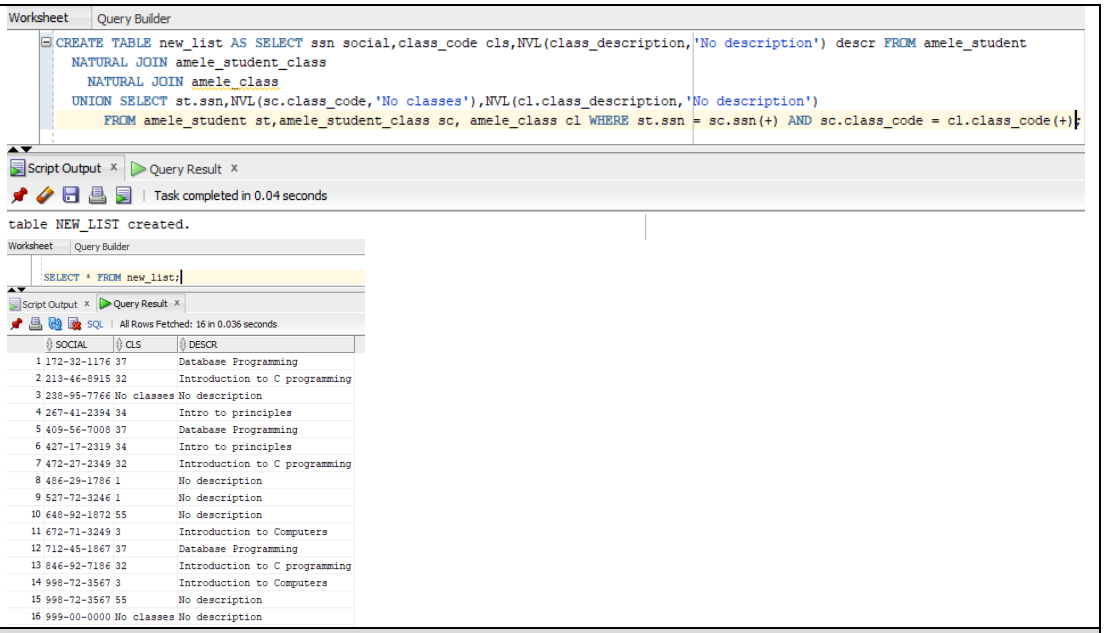
```
SELECT lname,class_code FROM amele_student
  INNER JOIN amele_student_class USING (ssn)
    INNER JOIN amele_class USING (class_code) WHERE class_description='Introduction to C programming';
```

Query Result | SQL | All Rows Fetched: 3 in 0.04 seconds

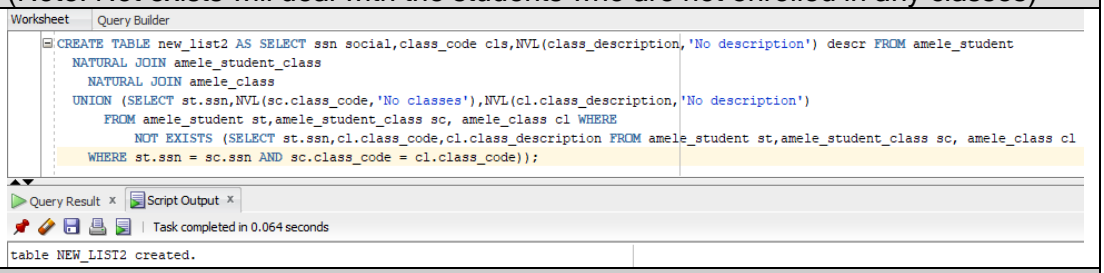
	LNAME	CLASS_CODE
1	Green	32
2	Gringlesby	32
3	Hunter	32

3 Give a lising of all the class_descriptions and the number of students enrolled in each class for all students who are older than the average age where the total number of students for the class is more than 1 student. Order by the number of students. If there

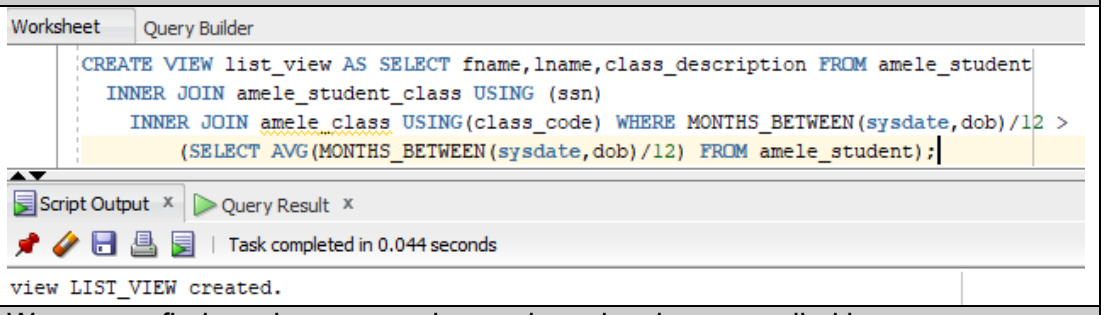
	is no class description replace it with 'Other Classes' (Note: Take it in steps. First do all those who are older than the average age, then do the group by, then add the having clause and then the order and then combine everything together)
	Worksheet Query Builder `SELECT NVL(class_description,'Other Classes'),COUNT(class_code) FROM amele_student_class INNER JOIN amele_class USING (class_code) INNER JOIN amele_student USING (ssn) WHERE MONTHS_BETWEEN(sysdate,dob)/12 > (SELECT AVG(MONTHS_BETWEEN(sysdate,dob)/12) FROM amele_student) GROUP BY class_description HAVING COUNT(class_code) > 1 ORDER BY COUNT(class_code);` Query Result SQL | All Rows Fetched: 3 in 0.036 seconds NVL(CLASS_DESCRIPTION,'OTHERCLASSES') | COUNT(CLASS_CODE) 1 Introduction to C programming 2 2 Other Classes 2 3 Database Programming 2
4	Give a listing of all the classes for which no students are enrolled in (use in or not in clause) (subquery)
	Worksheet Query Builder `SELECT class_code,class_description FROM amele_class WHERE class_code NOT IN (SELECT class_code FROM amele_student_class);` Query Result SQL | All Rows Fetched: 1 in 0.048 seconds CLASS_CODE | CLASS_DESCRIPTION 1 14A Operating systems
5	Give a listing of all the students who are not enrolled in any classes (Note: Use Exists or not Exists)
	Worksheet Query Builder `SELECT ssn,fname,lname FROM amele_student MINUS (SELECT ssn,fname,lname FROM amele_student st WHERE EXISTS (SELECT ssn FROM amele_student_class sc WHERE st.ssn=sc.ssn));` Query Result SQL | All Rows Fetched: 2 in 0.044 seconds SSN | FNAME | LNAME 1 238-95-7766 Cheryl Gren 2 999-00-0000 Cal Al
6	create a new table that contains the list of all the students and class_descriptions. Include In this table the list of all students who are not enrolled in any classes (display no classes). If there are no class descriptions then display 'no description' (Use combination of inner join, union and minus) (Note: minus will deal with the students who are not enrolled in any classes)



7	repeat question 6 using a combination of inner join, union and not exists (Note: Not exists will deal with the students who are not enrolled in any classes)



8	**create a view.** We want to find out which courses are being taken by the different students for all those whose age is greater than the average age. Give a listing of the course descriptions and student names (Inner join)



9	We want to find out the courses that each student is not enrolled in. Give a listing of the course descriptions, and the students (lname) who are not taking that specific course (Use a cartesian product and union it with a minus)

```
SELECT lname,class_description FROM amele_student,amele_student_class,amele_class
MINUS SELECT lname,class_description FROM amele_student
  NATURAL JOIN amele_student_class
    NATURAL JOIN amele_class;
```

Script Output | Query Result | SQL | Fetched 50 rows in 0.052 seconds

	LNAME	CLASS_DESCRIPTION
1	Al	Database Programming
2	Al	Intro to principles
3	Al	Introduction to C programming
4	Al	Introduction to Computers
5	Al	Operating systems
6	Al	(null)
7	Bennet	Intro to principles
8	Bennet	Introduction to C programming
9	Bennet	Introduction to Computers
10	Bennet	Operating systems
11	Bennet	(null)
12	Blotchet-Halls	Database Programming
13	Blotchet-Halls	Intro to principles
14	Blotchet-Halls	Introduction to C programming
15	Blotchet-Halls	Introduction to Computers
16	Blotchet-Halls	Operating systems
17	Dull	Database Programming
18	Dull	Introduction to C programming
19	Dull	Introduction to Computers
20	Dull	Operating systems
21	Dull	(null)
22	Greeenr	Database Programming
23	Greeenr	Intro to principles
24	Greeenr	Introduction to C programming
25	Greeenr	Operating systems
26	Green	Database Programming
27	Green	Intro to principles
28	Green	Introduction to Computers

10 Use the system catalog tables to display the results to find out the following:(Note show me the SQL syntax along with your results) Only a single SQL statement for each question.

a) Primary key name and the columns that make up the primary key for **student** table

```
SELECT constraint_name,column_name FROM user_cons_columns WHERE constraint_name IN
  (SELECT constraint_name FROM user_constraints WHERE table_name='AMELE_STUDENT' AND constraint_type='P');
```

Script Output | Query Result | SQL | All Rows Fetched: 1 in 0.952 seconds

	CONSTRAINT_NAME	COLUMN_NAME
1	AMELE_STUDENT_PK	SSN

b) Unique key name and the columns that make up the unique key for the **student** table

```
SELECT constraint_name,column_name FROM user_cons_columns WHERE constraint_name IN
  (SELECT constraint_name FROM user_constraints WHERE table_name='AMELE_STUDENT' AND constraint_type='U');
```

Script Output | Query Result | SQL | All Rows Fetched: 2 in 0.936 seconds

	CONSTRAINT_NAME	COLUMN_NAME
1	AMELE_STUDENT_UK	LNAME
2	AMELE_STUDENT_UK	FNAME

c) Foreign key name, the columns that make up the foreign key and the columns it references in the parent table for **student_class** table

d) Name of all the check constraints and their conditions for the **student** table

Worksheet | Query Builder

```
SELECT constraint_name,search_condition FROM user_constraints WHERE table_name='AMELE_STUDENT' AND constraint_type='C';
```

Script Output | Query Result

SQL | All Rows Fetched: 3 in 0.447 seconds

	CONSTRAINT_NAME	SEARCH_CONDITION
1	SYS_C00232568	"SSN" IS NOT NULL
2	SYS_C00232569	"LNAME" IS NOT NULL
3	SYS_C00232570	"FNAME" IS NOT NULL