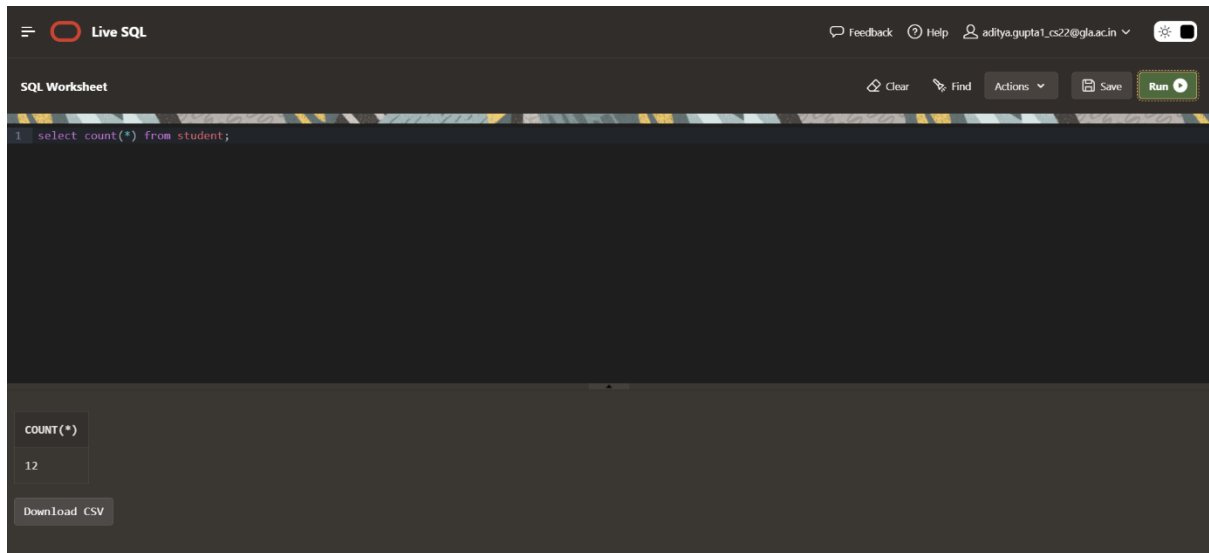


# DBMS ASSIGNMENT-4

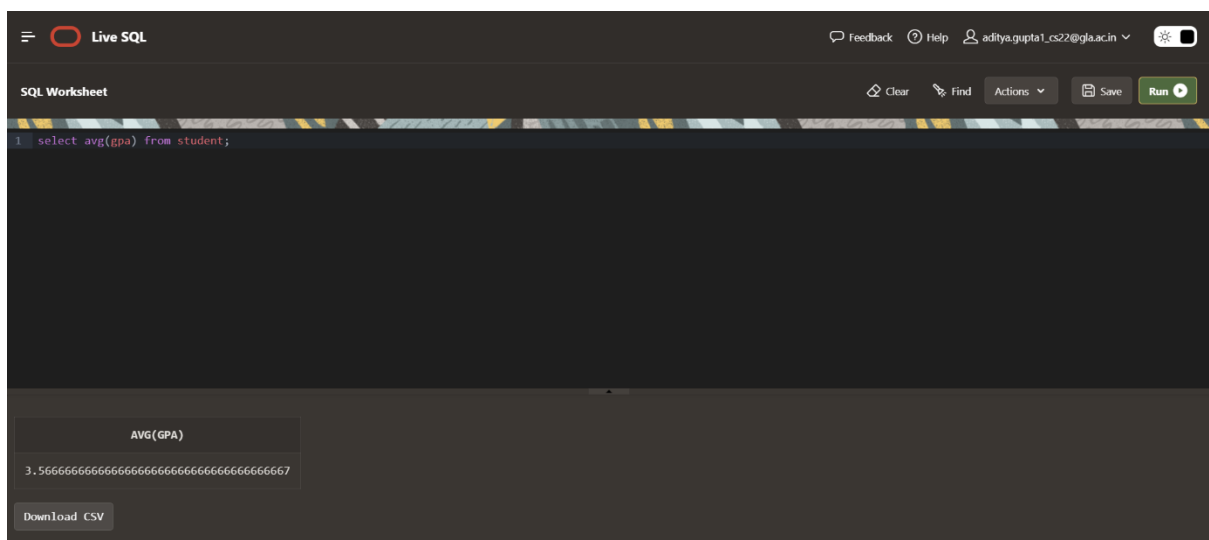
Q1. Count the total number of Students.



The screenshot shows the Live SQL interface. The SQL Worksheet contains the query: `1 select count(*) from student;`. The results panel displays a single row with the column header `COUNT(*)` and the value `12`. A `Download CSV` button is visible below the results.

COUNT(*)
12

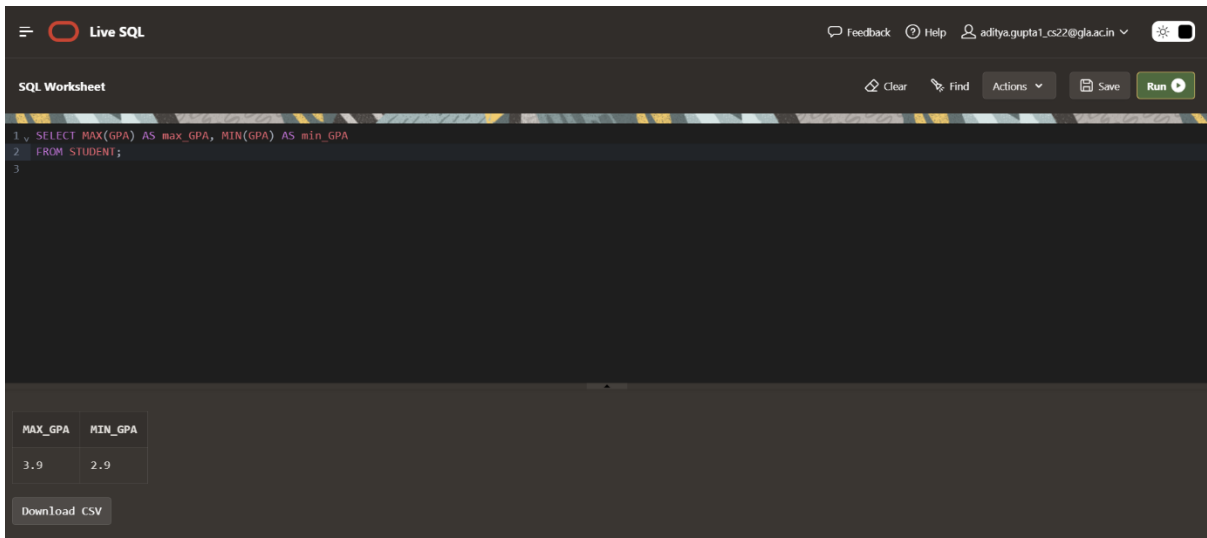
Q2. Calculate the average GPA of all Student.



The screenshot shows the Live SQL interface. The SQL Worksheet contains the query: `1 select avg(gpa) from student;`. The results panel displays a single row with the column header `AVG(GPA)` and the value `3.56666666666666666666666666666667`. A `Download CSV` button is visible below the results.

AVG(GPA)
3.56666666666666666666666666666667

Q3. Determine the minimum and maximum GPA. Rename the titles as 'max\_GPA' and 'min\_GPA' respectively.



The screenshot shows the Live SQL interface with a SQL Worksheet. The query entered is:

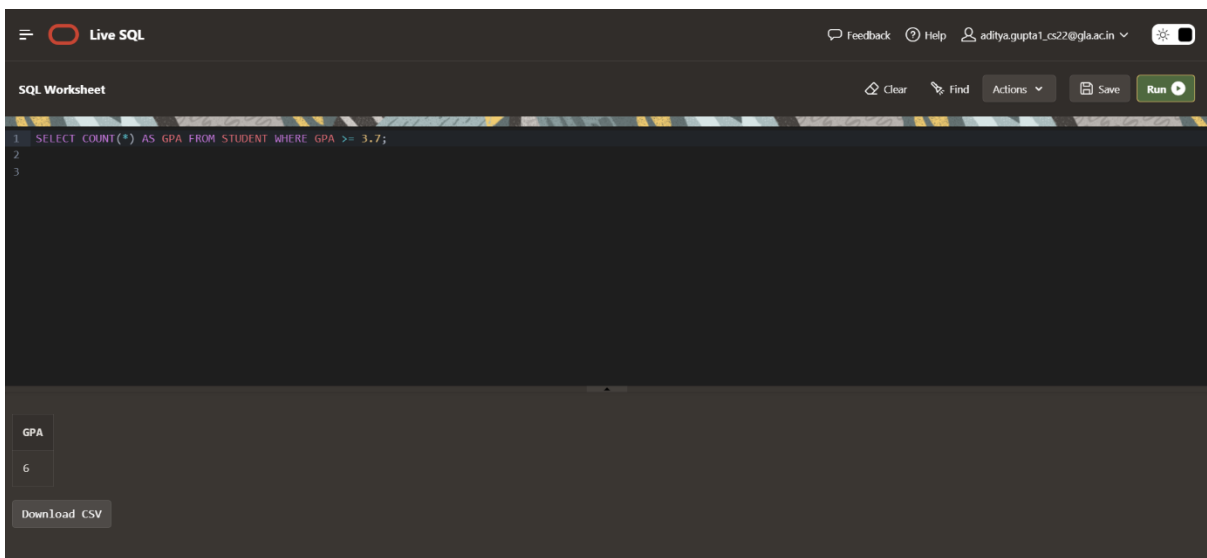
```
1 SELECT MAX(GPA) AS max_GPA, MIN(GPA) AS min_GPA
2 FROM STUDENT;
3
```

The results are displayed in a table:

MAX_GPA	MIN_GPA
3.9	2.9

There is a 'Download CSV' button below the results table.

Q4. Count the number of students having GPA greater than or equal to 3.7.



The screenshot shows the Live SQL interface with a SQL Worksheet. The query entered is:

```
1 SELECT COUNT(*) AS GPA FROM STUDENT WHERE GPA >= 3.7;
2
3
```

The results are displayed in a table:

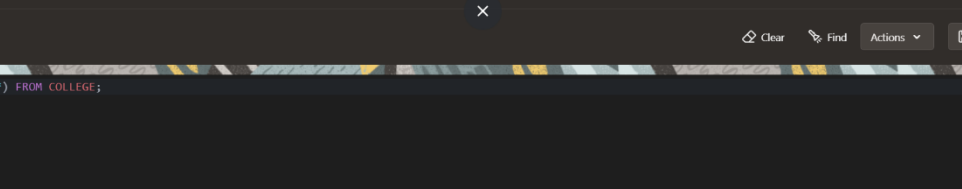
GPA
6

There is a 'Download CSV' button below the results table.

Q5. Find Maximum, Average, Minimum, total GPA of all student.

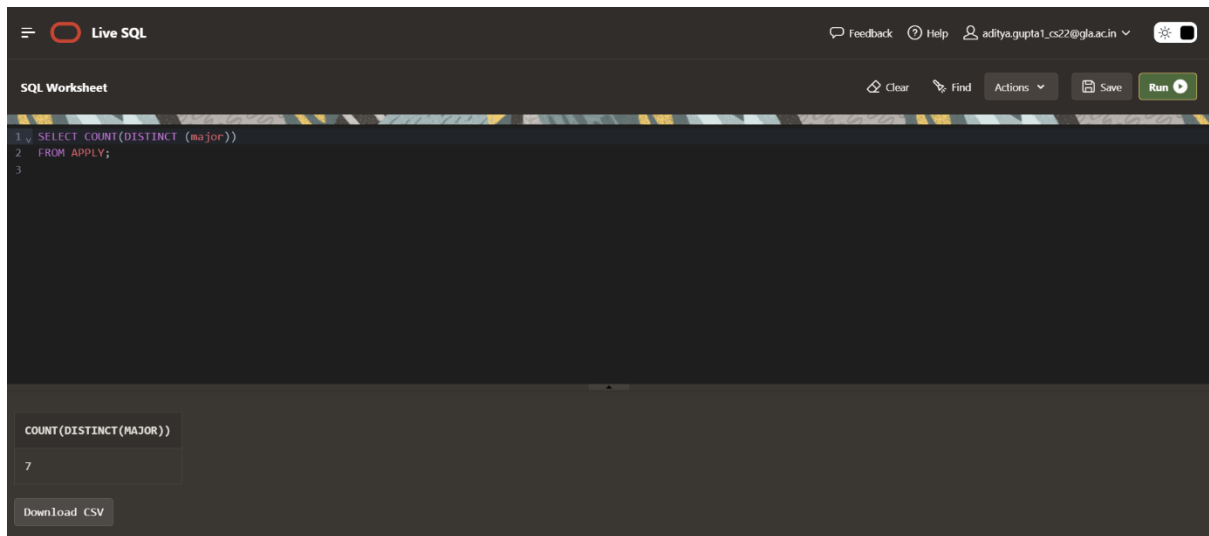
[illegible]

Q6. Find total number of colleges in our Application Database.



The screenshot shows the SQL Live SQL interface. At the top, there's a header with a hamburger menu, the "Live SQL" logo, and user information: "Feedback", "Help", and "aditya.gupta1\_cs22@glia.ac.in". Below the header, the title "SQL Worksheet" is displayed. The main area contains a SQL query: "SELECT COUNT(\*) FROM COLLEGE;". Below the query, the execution result is shown as a table with one row: "COUNT(\*)" and the value "5". At the bottom left, there is a "Download CSV" button.

Q7. Find how many different majors student had applied in.

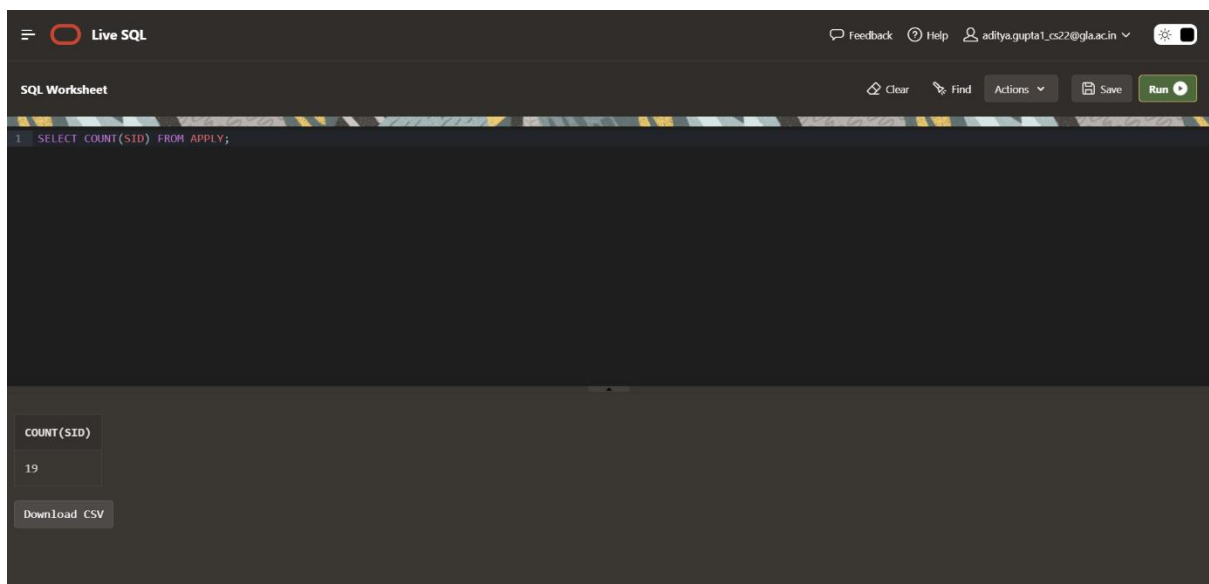


The screenshot shows the Live SQL interface. The top bar includes the 'Live SQL' logo, a user profile 'aditya.gupta1\_cs22@glia.ac.in', and icons for feedback, help, and settings. Below the top bar, the 'SQL Worksheet' section contains a query editor with the following SQL code:

```
1 SELECT COUNT(DISTINCT (major))
2 FROM APPLY;
3
```

Below the query editor, the results section displays the query 'COUNT(DISTINCT(MAJOR))' and the result '7'. A 'Download CSV' button is located at the bottom left of the results section.

Q8. Find total no. of Applications in our Application System's Database.



The screenshot shows the Live SQL interface. The top bar includes the 'Live SQL' logo, a user profile 'aditya.gupta1\_cs22@glia.ac.in', and icons for feedback, help, and settings. Below the top bar, the 'SQL Worksheet' section contains a query editor with the following SQL code:

```
1 SELECT COUNT(SID) FROM APPLY;
```

Below the query editor, the results section displays the query 'COUNT(SID)' and the result '19'. A 'Download CSV' button is located at the bottom left of the results section.

Q9. Find average of all distinct GPA.

The screenshot shows the Live SQL interface. The SQL Worksheet contains the query: `1 SELECT AVG(DISTINCT(GPA)) FROM STUDENT;`. The results table displays the query output:

AVG(DISTINCT(GPA))
3.54285714285714285714285714285714

Below the results table is a button labeled "Download CSV".

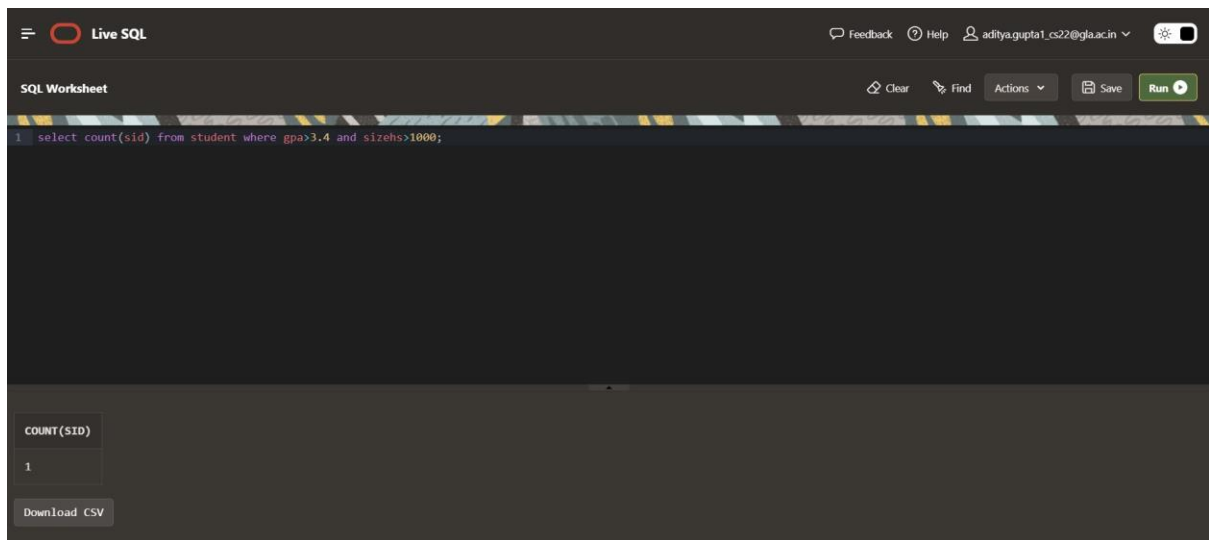
Q10. Display the total number of application accepted.

The screenshot shows the Live SQL interface. The SQL Worksheet contains the query: `1 select count(sid) from apply where decision = 'Y';`. The results table displays the query output:

COUNT(SID)
11

Below the results table is a button labeled "Download CSV".

Q11. Find number of students having GPA>3.4 and coming from high school having size>1000.

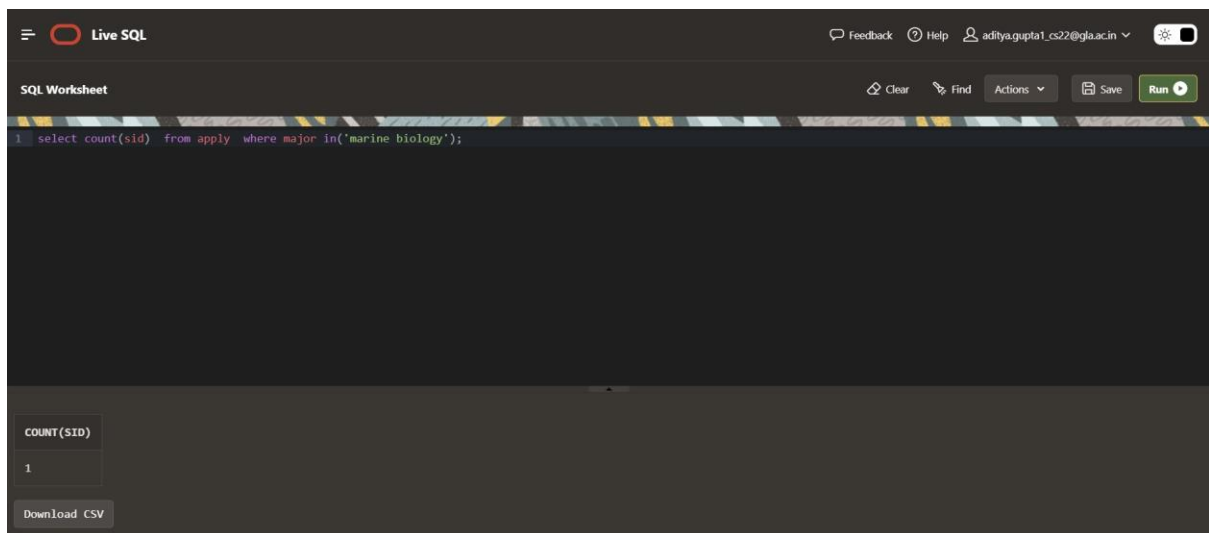


The screenshot shows the Live SQL interface. At the top, there's a header with the Live SQL logo, a feedback icon, a help icon, and a user profile. Below the header, there's a toolbar with 'Clear', 'Find', 'Actions', 'Save', and 'Run' buttons. The main area contains a SQL query: `1 select count(sid) from student where gpa>3.4 and size>1000;`. Below the query, there's a result table with one row: 

COUNT(SID)
1

. At the bottom left, there's a 'Download CSV' button.

Q12. Find how many student applied to 'marine biology'.

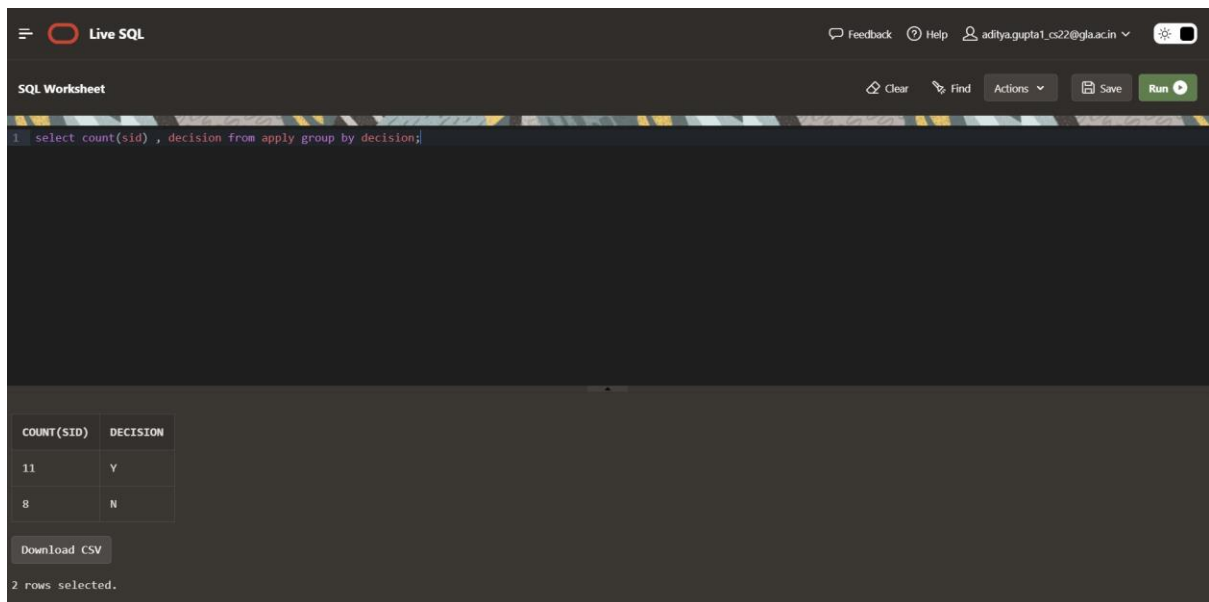


The screenshot shows the Live SQL interface. At the top, there's a header with the Live SQL logo, a feedback icon, a help icon, and a user profile. Below the header, there's a toolbar with 'Clear', 'Find', 'Actions', 'Save', and 'Run' buttons. The main area contains a SQL query: `1 select count(sid) from apply where major in('marine biology');`. Below the query, there's a result table with one row: 

COUNT(SID)
1

. At the bottom left, there's a 'Download CSV' button.

Q13. Find how many applications were rejected and accepted by the colleges.

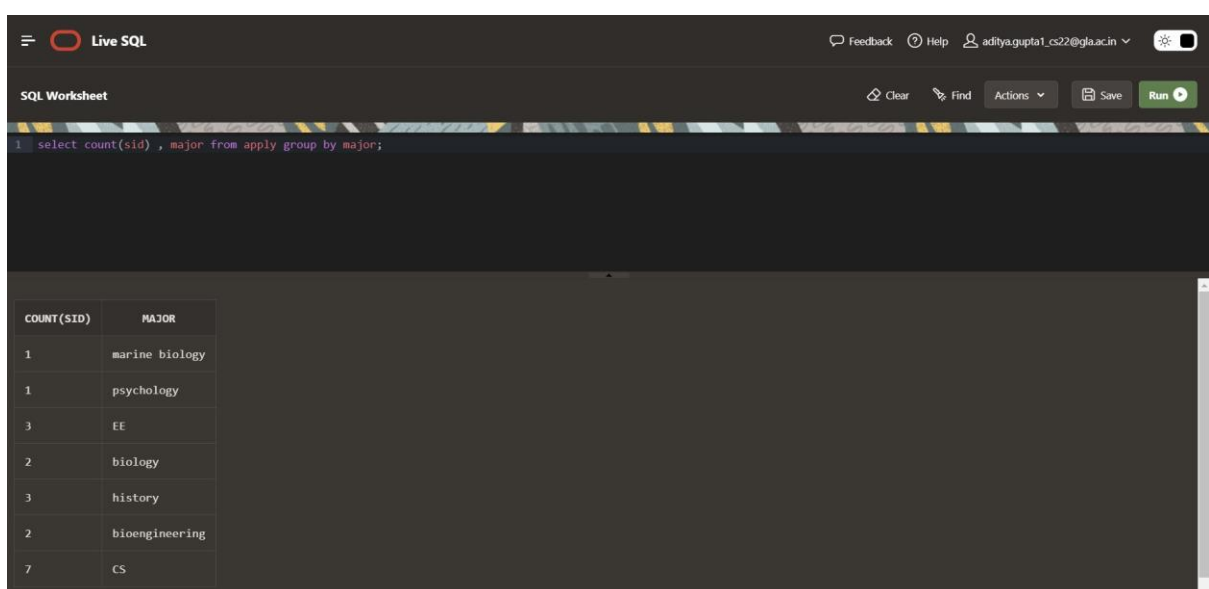


The screenshot shows the Live SQL interface with a query executed. The query is: `select count(sid) , decision from apply group by decision;`. The results are displayed in a table with two columns: COUNT(SID) and DECISION. There are 2 rows selected.

COUNT(SID)	DECISION
11	Y
8	N

Download CSV  
2 rows selected.

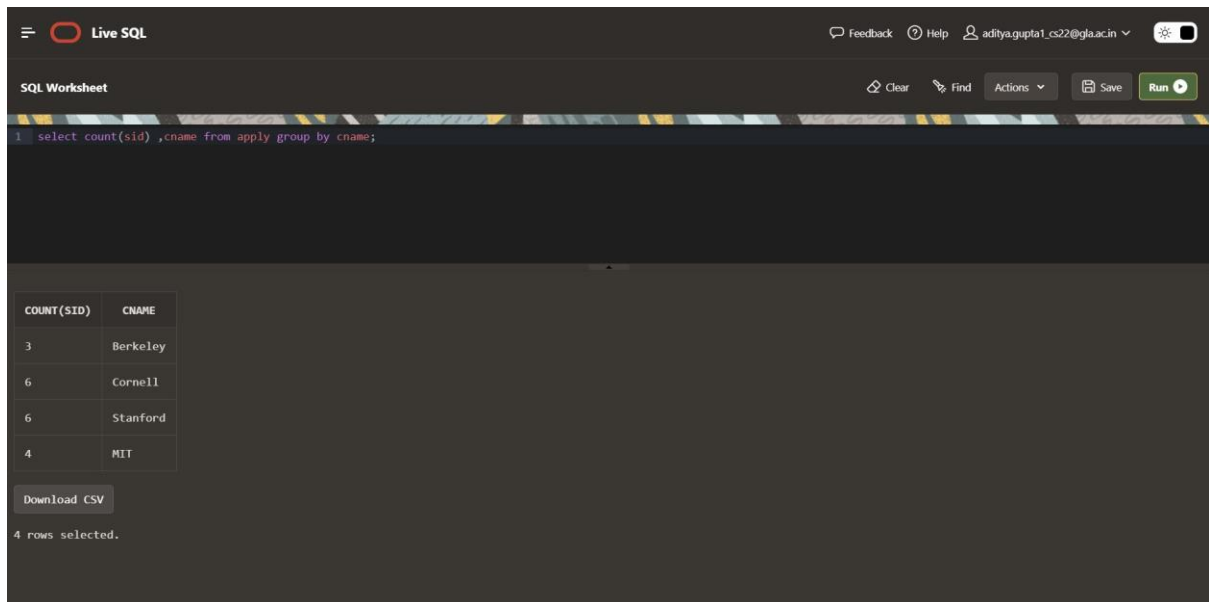
Q14. Find how many students applied to a particular major. (show count(sid) as No\_of\_applications).



The screenshot shows the Live SQL interface with a query executed. The query is: `select count(sid) , major from apply group by major;`. The results are displayed in a table with two columns: COUNT(SID) and MAJOR. There are 7 rows.

COUNT(SID)	MAJOR
1	marine biology
1	psychology
3	EE
2	biology
3	history
2	bioengineering
7	CS

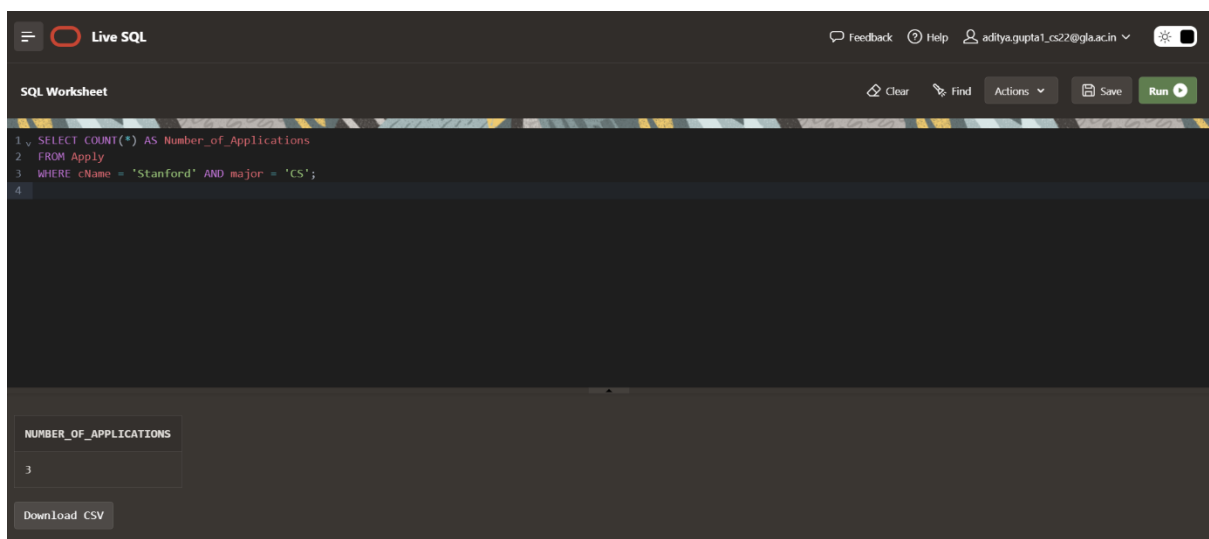
Q15. Find number of applications received by particular college.



The screenshot shows the Live SQL interface. The SQL query entered is: `1 select count(sid) ,cname from apply group by cname;`. The results are displayed in a table with two columns: **COUNT(SID)** and **CNAME**. The data rows are: (3, Berkeley), (6, Cornell), (6, Stanford), and (4, MIT). Below the table is a 'Download CSV' button and the text '4 rows selected.'.

COUNT(SID)	CNAME
3	Berkeley
6	Cornell
6	Stanford
4	MIT

Q16. Find number of applications received in a particular major at a particular college.



The screenshot shows the Live SQL interface. The SQL query entered is: `1 SELECT COUNT(*) AS Number_of_Applications  
2 FROM Apply  
3 WHERE cName = 'Stanford' AND major = 'CS';  
4`. The results are displayed in a table with one column: **NUMBER\_OF\_APPLICATIONS**. The data row is: (3). Below the table is a 'Download CSV' button.

NUMBER_OF_APPLICATIONS
3



Q17. Give the college name and major, where number of applications received are greater than or equal to 2.

The screenshot shows the Live SQL interface with a SQL query entered in the worksheet. The query is:

```
1 SELECT cName, major
2 FROM Apply
3 GROUP BY cName, major
4 HAVING COUNT(*) >= 2;
```

The results are displayed in a table below the query:

CNAME	MAJOR
Berkeley	CS
Stanford	history
Stanford	CS
Cornell	EE

Q18. Give the name and no of applications of all those colleges which receives applications from 3 or more students.

The screenshot shows the Live SQL interface with a SQL query entered in the worksheet. The query is:

```
1 SELECT c.cName AS College_Name, COUNT(a.sID) AS Number_of_Applications
2 FROM College c
3 JOIN Apply a ON c.cName = a.cName
4 GROUP BY c.cName
5 HAVING COUNT(DISTINCT a.sID) >= 3;
```

The results are displayed in a table below the query:

COLLEGE_NAME	NUMBER_OF_APPLICATIONS
Berkeley	3
Cornell	6
Stanford	6
MIT	4

Q19. Give state and number of colleges of a state that has more than 1 college.

The screenshot shows the Live SQL interface with the following SQL query:

```
1 SELECT state, COUNT(cName) AS Number_of_Colleges
2 FROM College
3 GROUP BY state
4 HAVING COUNT(cName) > 1;
```

The results table displays the following data:

STATE	NUMBER_OF_COLLEGES
MA	2
CA	2

Below the table, there is a "Download CSV" button and a status message: "2 rows selected."

Q20. Find the name of students that are duplicate. Q20. Find the name of students that are duplicate.

The screenshot shows the Live SQL interface with the following SQL query:

```
1 SELECT sName, COUNT(*) AS Duplicate_Count
2 FROM Student
3 GROUP BY sName
4 HAVING COUNT(*) > 1;
```

The results table displays the following data:

SNAME	DUPLICATE_COUNT
Amy	2
Craig	2

Below the table, there is a "Download CSV" button and a status message: "2 rows selected."

NAME → ADITYA GUPTA

SECTION → B

UNIV ROLLNO. → 2215000094

CLASS ROLLNO. → 47