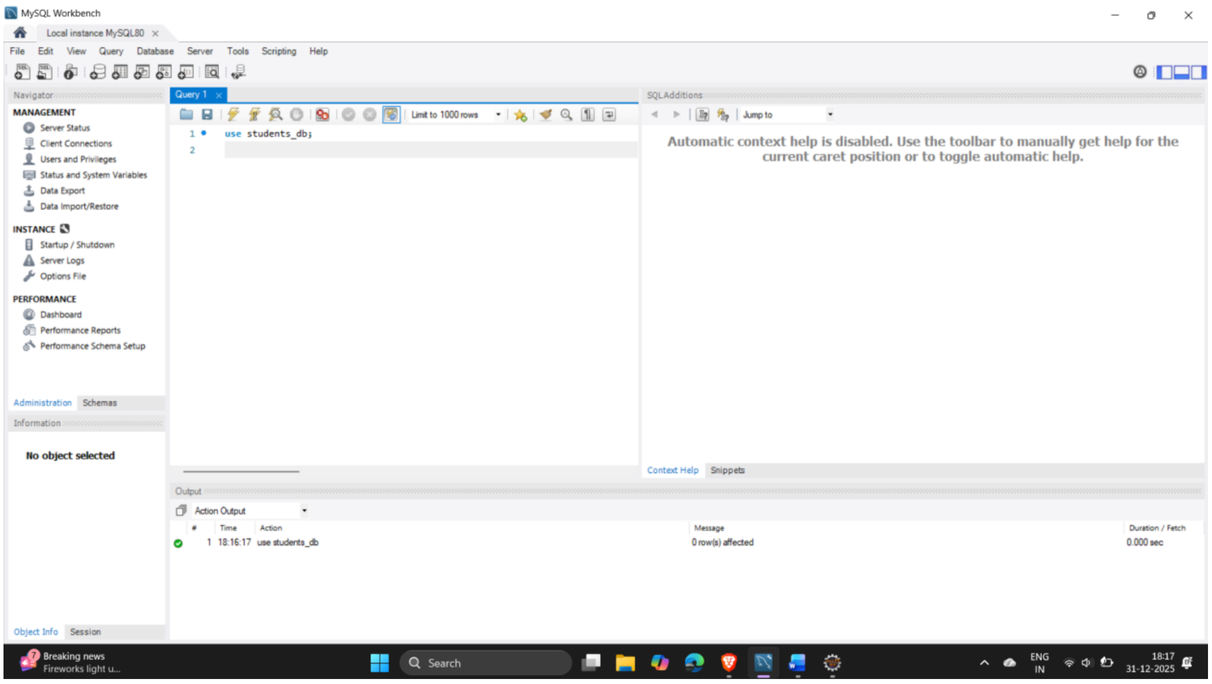
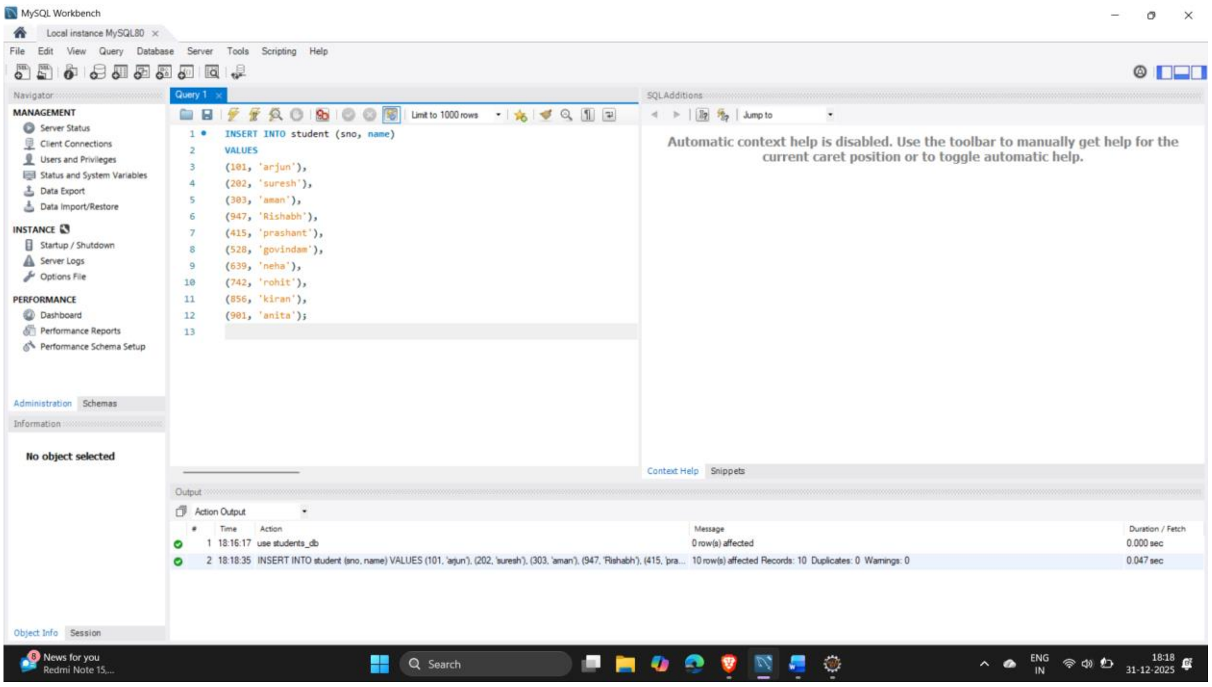


```
use students_db;
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action           Message            Duration / Fetch
1   18:16:17  use students_db  0 row(s) affected  0.000 sec
```

Inserting Values in table student

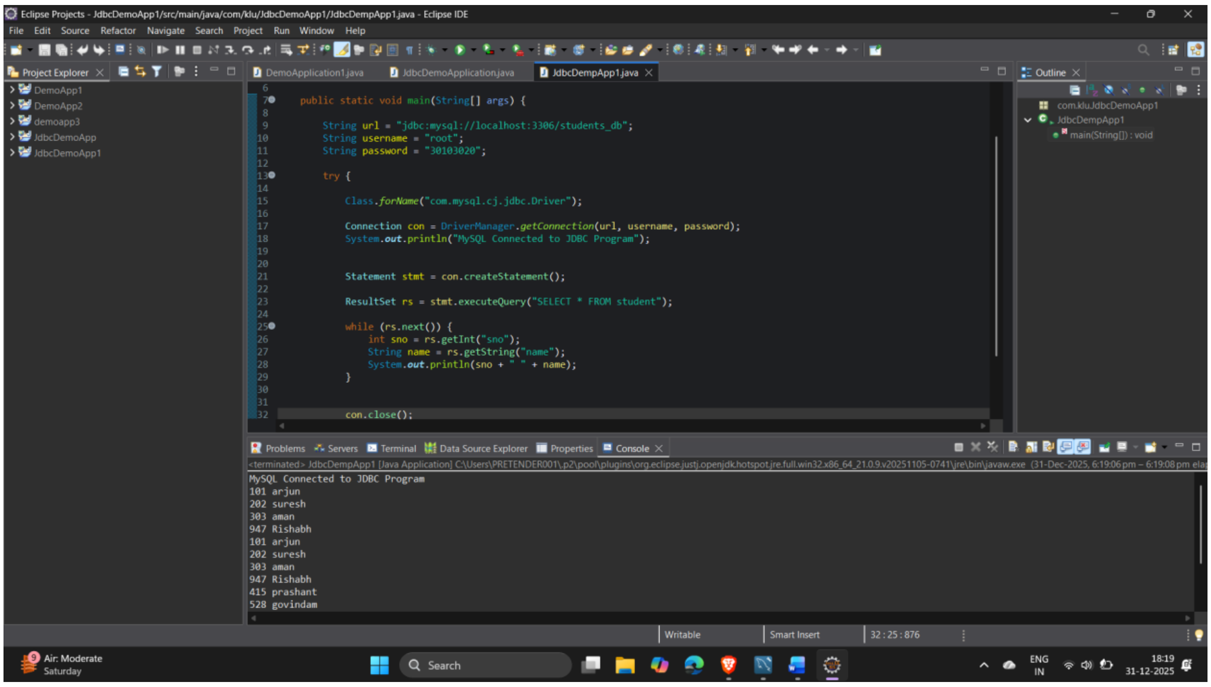


```
INSERT INTO student (sno, name)
VALUES
(101, 'arjun'),
(202, 'suresh'),
(303, 'aman'),
(947, 'Rishabh'),
(415, 'prashant'),
(528, 'govindam'),
(639, 'neha'),
(742, 'rohit'),
(856, 'kiran'),
(901, 'anita');
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                                                Message                                                        Duration / Fetch
1   18:16:17  use students_db                                                                                       0 row(s) affected                                              0.000 sec
2   18:18:35  INSERT INTO student (sno, name) VALUES (101, 'arjun'), (202, 'suresh'), (303, 'aman'), (947, 'Rishabh'), (415, 'pra...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0  0.047 sec
```

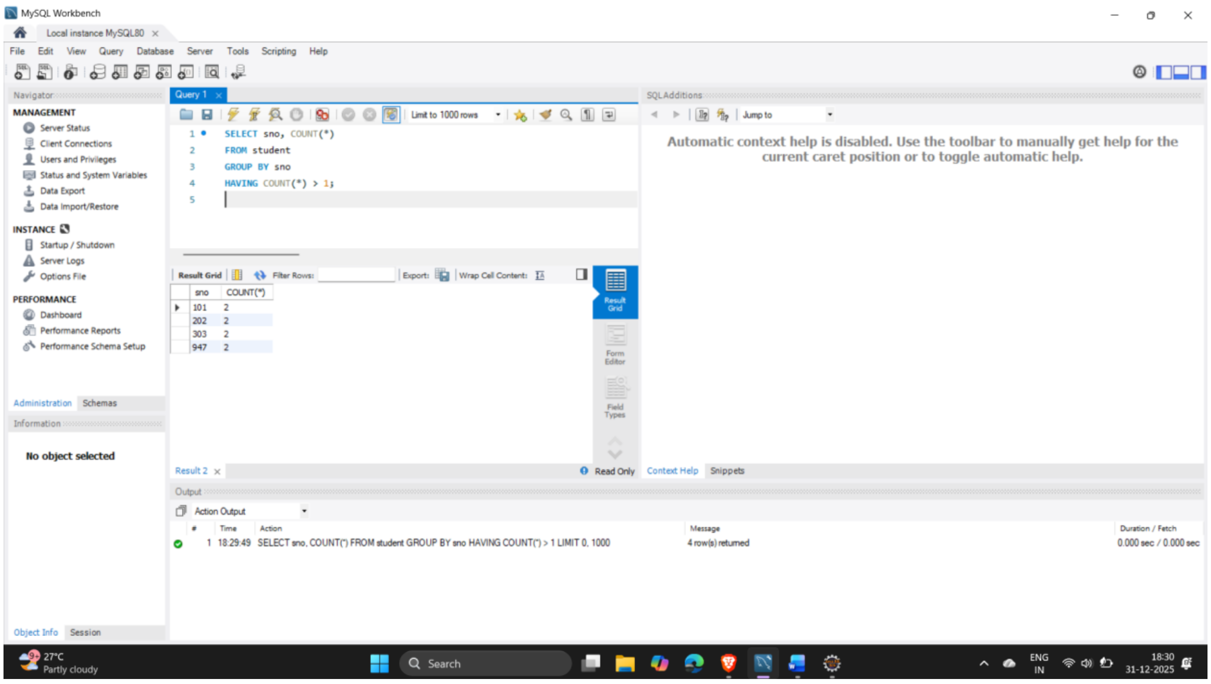
Now run as - java application



```
    public static void main(String[] args) {

        String url = "jdbc:mysql://localhost:3306/students_db";
        String username = "root";
        String password = "30103020";

        try {

            Class.forName("com.mysql.cj.jdbc.Driver");

            Connection con = DriverManager.getConnection(url, username, password);
            System.out.println("MySQL Connected to JDBC Program");


            Statement stmt = con.createStatement();

            ResultSet rs = stmt.executeQuery("SELECT * FROM student");

            while (rs.next()) {
                int sno = rs.getInt("sno");
                String name = rs.getString("name");
                System.out.println(sno + " " + name);
            }



            con.close();
```

```
<terminated> JdbcDempApp1 [Java Application] C:\Users\PRETENDER001\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-0741\jre\bin\javaw.exe (31-Dec-2025, 6:19:06 pm – 6:19:08 pm ela
MySQL Connected to JDBC Program
101 arjun
202 suresh
303 aman
947 Rishabh
101 arjun
202 suresh
303 aman
947 Rishabh
415 prashant
528 govindam
```

Query for finding duplicates

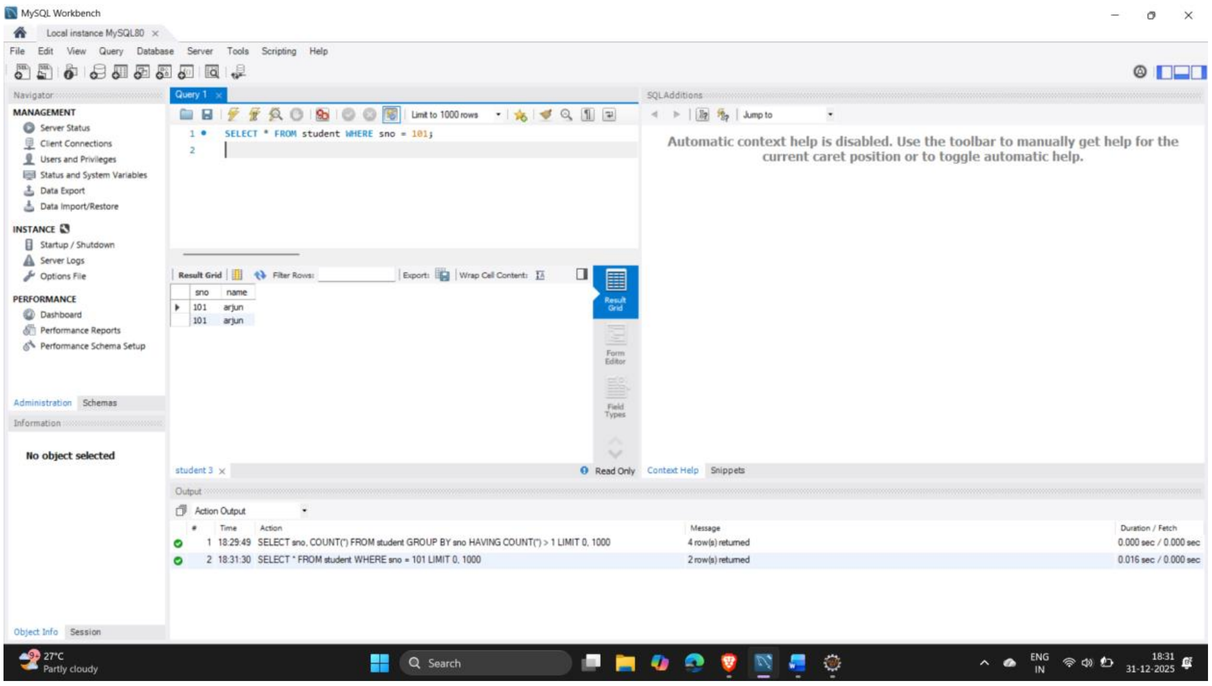


```
SELECT sno, COUNT(*)
FROM student
GROUP BY sno
HAVING COUNT(*) > 1;
```

sno	COUNT(*)
101	2
202	2
303	2
947	2

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                  Message            Duration / Fetch
1   18:29:49  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000   4 row(s) returned   0.000 sec / 0.000 sec
```

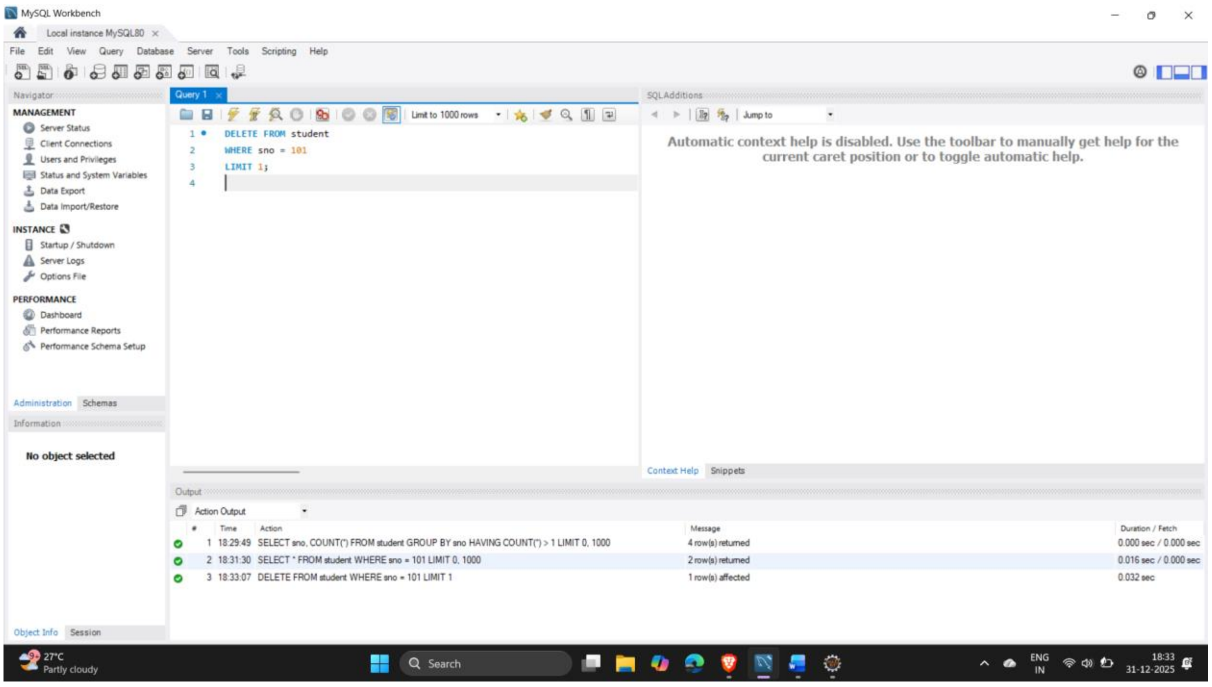


```
SELECT * FROM student WHERE sno = 101;
```

sno	name
101	arjun
101	arjun

```
#  Time      Action                                                                      Message          Duration / Fetch
1  18:29:49  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  4 row(s) returned  0.000 sec / 0.000 sec
2  18:31:30  SELECT * FROM student WHERE sno = 101 LIMIT 0, 1000                              2 row(s) returned  0.016 sec / 0.000 sec
```

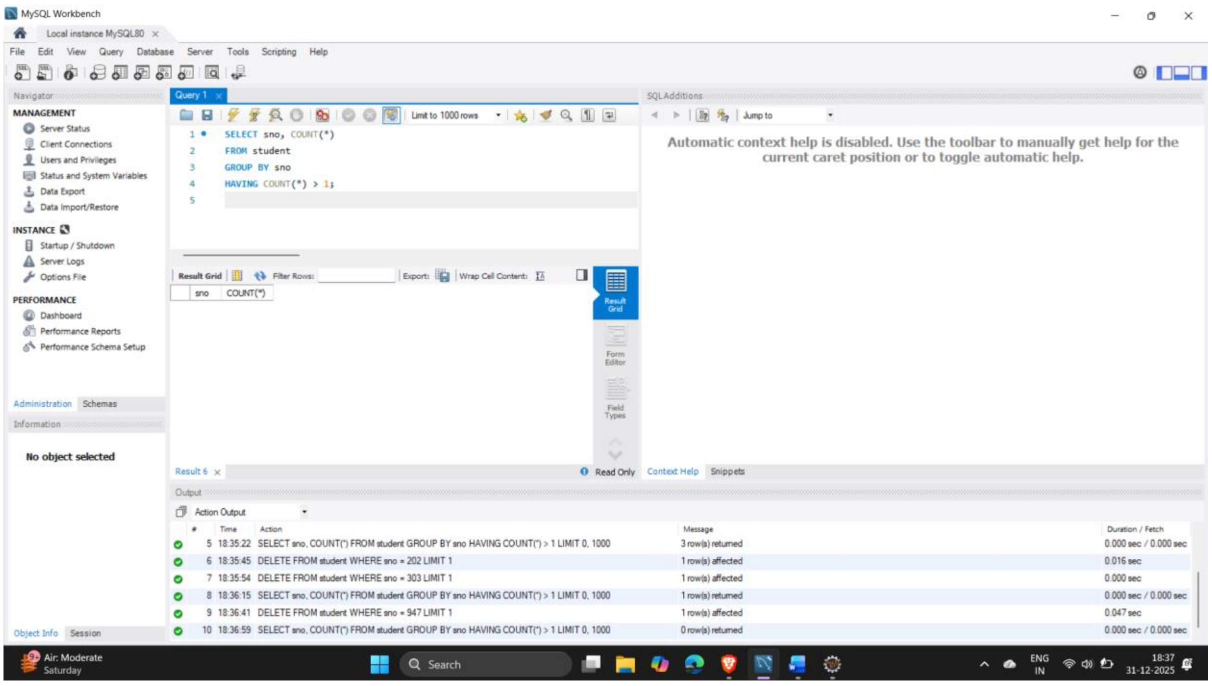
Fixing duplicates



```
DELETE FROM student
WHERE sno = 101
LIMIT 1;
```

```
#  Time      Action                                                                      Message          Duration / Fetch
1  18:29:49  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  4 row(s) returned  0.000 sec / 0.000 sec
2  18:31:30  SELECT * FROM student WHERE sno = 101 LIMIT 0, 1000                              2 row(s) returned  0.016 sec / 0.000 sec
3  18:33:07  DELETE FROM student WHERE sno = 101 LIMIT 1                                      1 row(s) affected  0.032 sec
```

Delete all duplicates one by one like above until you get something like this



```sql
SELECT sno, COUNT(*)
FROM student
GROUP BY sno
HAVING COUNT(*) > 1;
```

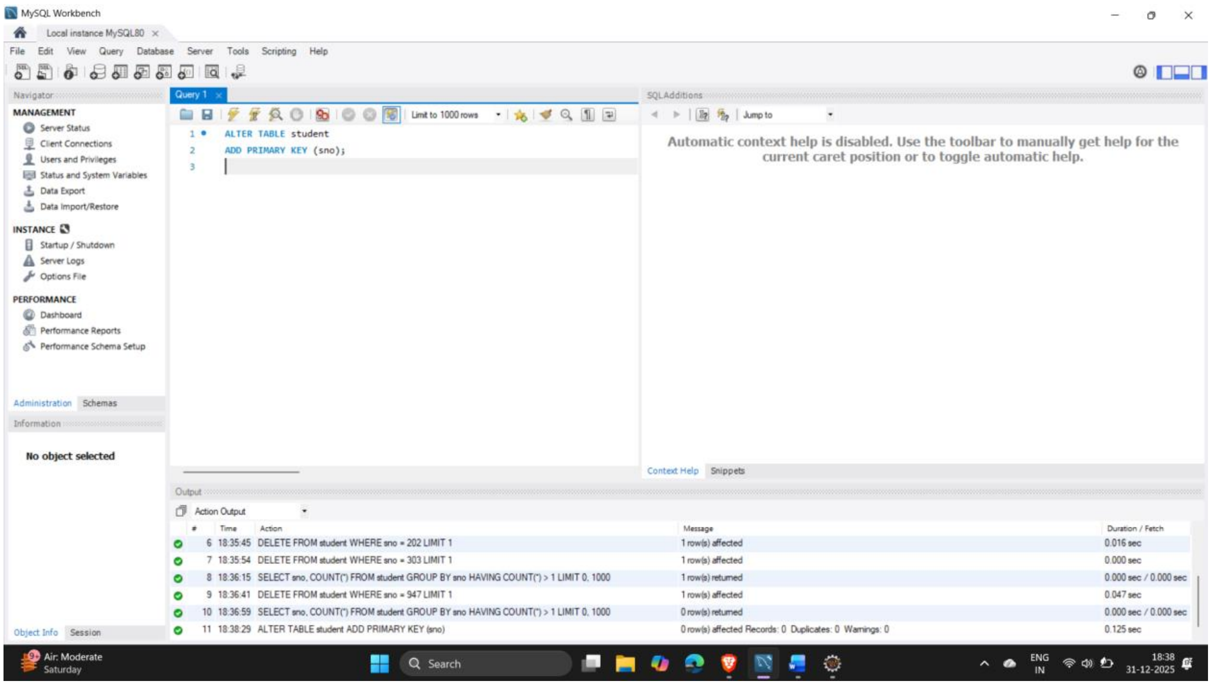
Result Grid

sno	COUNT(*)

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                          Message           Duration / Fetch
5   18:35:22  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  3 row(s) returned  0.000 sec / 0.000 sec
6   18:35:45  DELETE FROM student WHERE sno = 202 LIMIT 1                                     1 row(s) affected  0.016 sec
7   18:35:54  DELETE FROM student WHERE sno = 303 LIMIT 1                                     1 row(s) affected  0.000 sec
8   18:36:15  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  1 row(s) returned  0.000 sec / 0.000 sec
9   18:36:41  DELETE FROM student WHERE sno = 947 LIMIT 1                                     1 row(s) affected  0.047 sec
10  18:36:59  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  0 row(s) returned  0.000 sec / 0.000 sec
```

Now add Primary Key(sno)

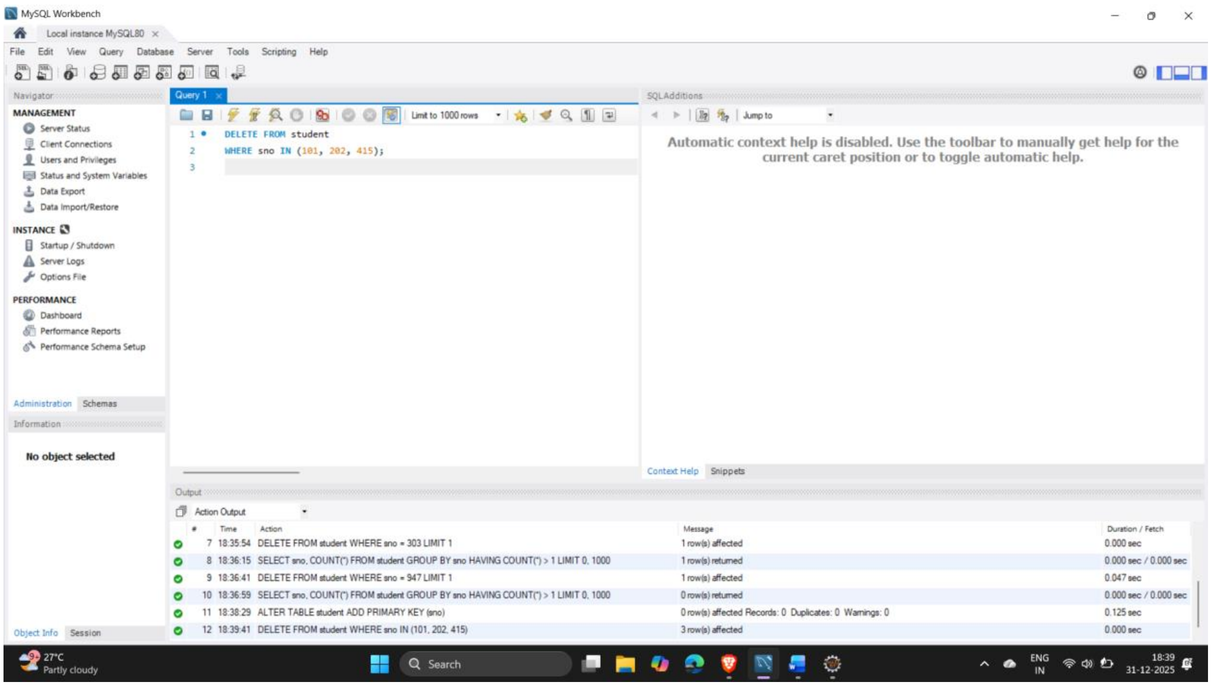


```sql
ALTER TABLE student
ADD PRIMARY KEY (sno);
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                          Message                                                 Duration / Fetch
6   18:35:45  DELETE FROM student WHERE sno = 202 LIMIT 1                                     1 row(s) affected                                       0.016 sec
7   18:35:54  DELETE FROM student WHERE sno = 303 LIMIT 1                                     1 row(s) affected                                       0.000 sec
8   18:36:15  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  1 row(s) returned                                       0.000 sec / 0.000 sec
9   18:36:41  DELETE FROM student WHERE sno = 947 LIMIT 1                                     1 row(s) affected                                       0.047 sec
10  18:36:59  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000  0 row(s) returned                                       0.000 sec / 0.000 sec
11  18:38:29  ALTER TABLE student ADD PRIMARY KEY (sno)                                       0 row(s) affected Records: 0  Duplicates: 0  Warnings: 0  0.125 sec
```

Query for deleting records

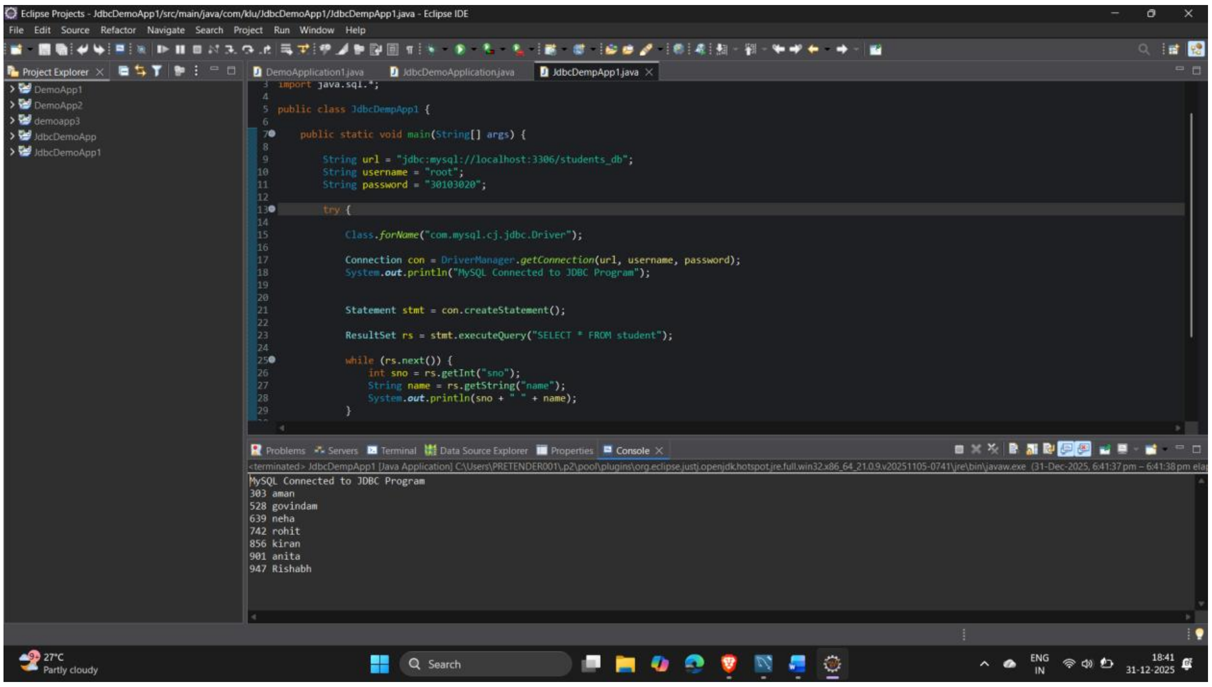


```sql
DELETE FROM student
WHERE sno IN (101, 202, 415);
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

#	Time	Action	Message	Duration / Fetch
7	18:35:54	DELETE FROM student WHERE sno = 303 LIMIT 1	1 row(s) affected	0.000 sec
8	18:36:15	SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
9	18:36:41	DELETE FROM student WHERE sno = 947 LIMIT 1	1 row(s) affected	0.047 sec
10	18:36:59	SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
11	18:38:29	ALTER TABLE student ADD PRIMARY KEY (sno)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.125 sec
12	18:39:41	DELETE FROM student WHERE sno IN (101, 202, 415)	3 row(s) affected	0.000 sec

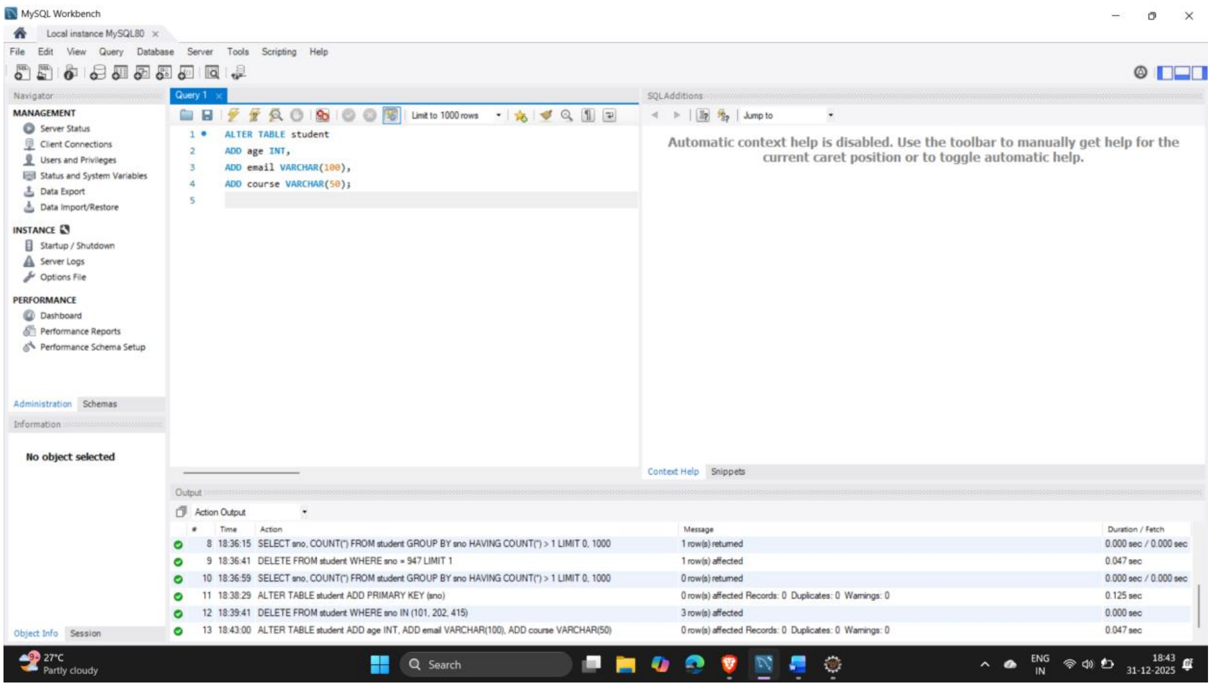
Verify in Eclipse Ide



```java
import java.sql.*;

public class JdbcDempApp1 {

    public static void main(String[] args) {

        String url = "jdbc:mysql://localhost:3306/students_db";
        String username = "root";
        String password = "30103020";

        try {

            Class.forName("com.mysql.cj.jdbc.Driver");

            Connection con = DriverManager.getConnection(url, username, password);
            System.out.println("MySQL Connected to JDBC Program");


            Statement stmt = con.createStatement();

            ResultSet rs = stmt.executeQuery("SELECT * FROM student");

            while (rs.next()) {
                int sno = rs.getInt("sno");
                String name = rs.getString("name");
                System.out.println(sno + " " + name);
            }
```

```
MySQL Connected to JDBC Program
303 aman
528 govindam
639 neha
742 rohit
856 kiran
901 anita
947 Rishabh
```

Adding New columns

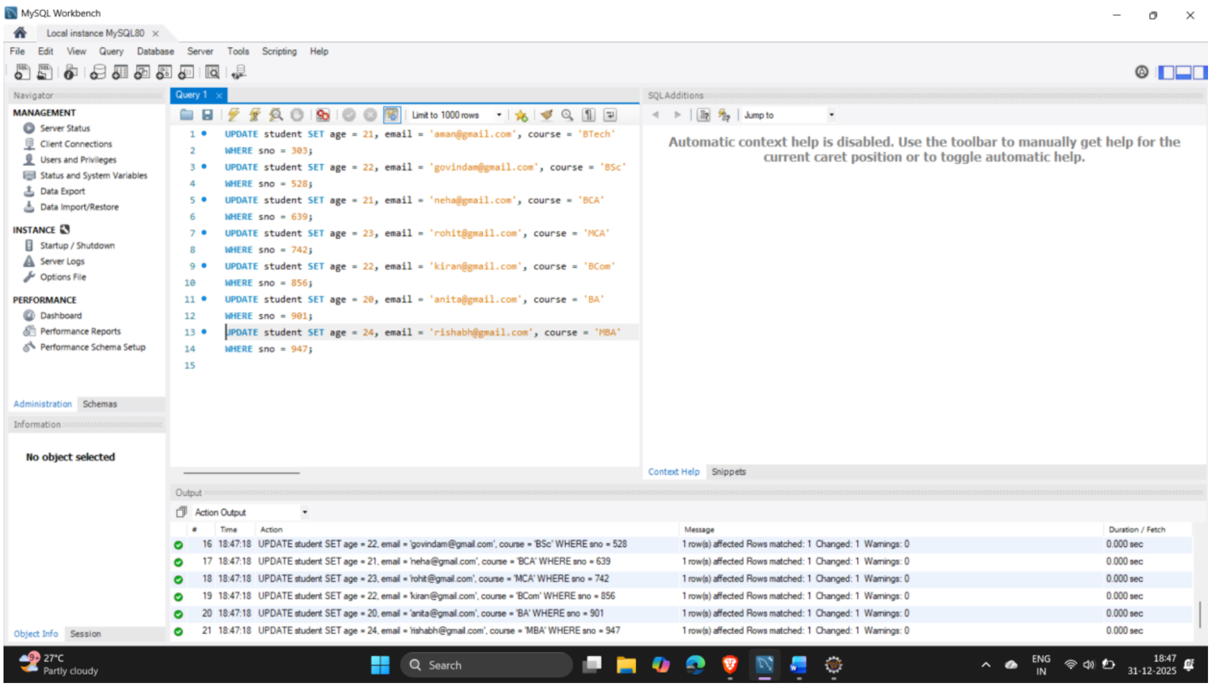


```
ALTER TABLE student
ADD age INT,
ADD email VARCHAR(100),
ADD course VARCHAR(50);
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                  Message                                                   Duration / Fetch
8   18:36:15  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000   1 row(s) returned                                0.000 sec / 0.000 sec
9   18:36:41  DELETE FROM student WHERE sno = 947 LIMIT 1                             1 row(s) affected                                         0.047 sec
10  18:36:59  SELECT sno, COUNT(*) FROM student GROUP BY sno HAVING COUNT(*) > 1 LIMIT 0, 1000   0 row(s) returned                                0.000 sec / 0.000 sec
11  18:38:29  ALTER TABLE student ADD PRIMARY KEY (sno)                               0 row(s) affected Records: 0  Duplicates: 0  Warnings: 0  0.125 sec
12  18:39:41  DELETE FROM student WHERE sno IN (101, 202, 415)                        3 row(s) affected                                         0.000 sec
13  18:43:00  ALTER TABLE student ADD age INT, ADD email VARCHAR(100), ADD course VARCHAR(50)   0 row(s) affected Records: 0  Duplicates: 0  Warnings: 0  0.047 sec
```

Update the records

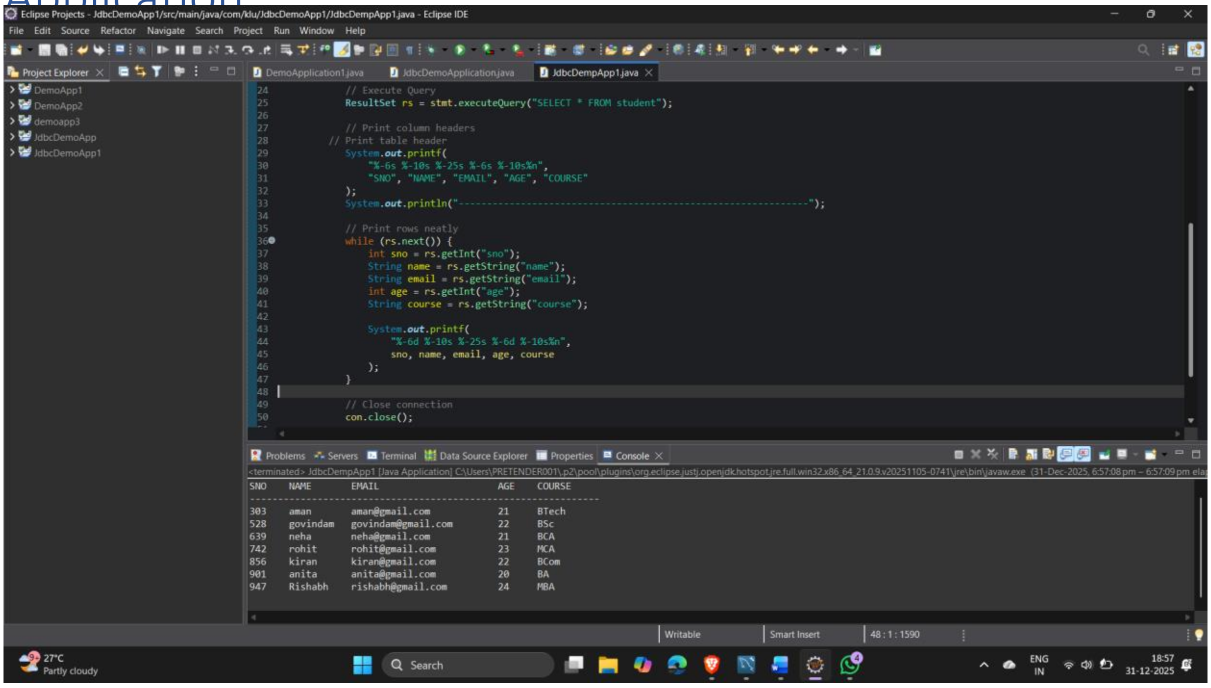


```
UPDATE student SET age = 21, email = 'aman@gmail.com', course = 'BTech'
WHERE sno = 303;
UPDATE student SET age = 22, email = 'govindam@gmail.com', course = 'BSc'
WHERE sno = 528;
UPDATE student SET age = 21, email = 'neha@gmail.com', course = 'BCA'
WHERE sno = 639;
UPDATE student SET age = 23, email = 'rohit@gmail.com', course = 'MCA'
WHERE sno = 742;
UPDATE student SET age = 22, email = 'kiran@gmail.com', course = 'BCom'
WHERE sno = 856;
UPDATE student SET age = 20, email = 'anita@gmail.com', course = 'BA'
WHERE sno = 901;
UPDATE student SET age = 24, email = 'rishabh@gmail.com', course = 'MBA'
WHERE sno = 947;
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                             Message                                                    Duration / Fetch
16  18:47:18  UPDATE student SET age = 22, email = 'govindam@gmail.com', course = 'BSc' WHERE sno = 528   1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.000 sec
17  18:47:18  UPDATE student SET age = 21, email = 'neha@gmail.com', course = 'BCA' WHERE sno = 639      1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.000 sec
18  18:47:18  UPDATE student SET age = 23, email = 'rohit@gmail.com', course = 'MCA' WHERE sno = 742     1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.000 sec
19  18:47:18  UPDATE student SET age = 22, email = 'kiran@gmail.com', course = 'BCom' WHERE sno = 856    1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.000 sec
20  18:47:18  UPDATE student SET age = 20, email = 'anita@gmail.com', course = 'BA' WHERE sno = 901      1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.000 sec
21  18:47:18  UPDATE student SET age = 24, email = 'rishabh@gmail.com', course = 'MBA' WHERE sno = 947   1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.000 sec
```

Update the code in Eclipse Ide – Run as – Java Application



```java
        // Execute Query
        ResultSet rs = stmt.executeQuery("SELECT * FROM student");

        // Print column headers
     // Print table header
        System.out.printf(
            "%-6s %-10s %-25s %-6s %-10s%n",
            "SNO", "NAME", "EMAIL", "AGE", "COURSE"
        );
        System.out.println("------------------------------------------------------------");

        // Print rows neatly
        while (rs.next()) {
            int sno = rs.getInt("sno");
            String name = rs.getString("name");
            String email = rs.getString("email");
            int age = rs.getInt("age");
            String course = rs.getString("course");

            System.out.printf(
                "%-6d %-10s %-25s %-6d %-10s%n",
                sno, name, email, age, course
            );
        }

        // Close connection
        con.close();
```

```
<terminated> JdbcDempApp1 [Java Application] C:\Users\PRETENDER001\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-0741\jre\bin\javaw.exe (31-Dec-2025, 6:57:08 pm – 6:57:09 pm ela
SNO    NAME       EMAIL                     AGE    COURSE
------------------------------------------------------------
303    aman       aman@gmail.com            21     BTech
528    govindam   govindam@gmail.com        22     BSc
639    neha       neha@gmail.com            21     BCA
742    rohit      rohit@gmail.com           23     MCA
856    kiran      kiran@gmail.com           22     BCom
901    anita      anita@gmail.com           20     BA
947    Rishabh    rishabh@gmail.com         24     MBA
```