```sql
-- 1. CRUD ON Departments
-- CREATE
INSERT INTO Departments VALUES (6,'ARTS');

-- READ
SELECT * FROM Departments;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

DepartmentID	DepartmentName
1	Data Science
2	Statistics
3	PBL
4	Mechanical
5	Civil Engineering
6	ARTS
NULL	NULL

```sql
-- UPDATE
UPDATE Departments
SET DepartmentName='science'
WHERE DepartmentID=6;

-- DELETE
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

DepartmentID	DepartmentName
1	Data Science
2	Statistics
3	PBL
4	Mechanical
5	Civil Engineering
6	science
NULL	NULL

```sql
-- DELETE
DELETE FROM Departments
WHERE DepartmentID=6;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

DepartmentID	DepartmentName
1	Data Science
2	Statistics
3	PBL
4	Mechanical
5	Civil Engineering
NULL	NULL

```
136     (11,'PRINCE','PRAJAPATI','PRINCE@gmail.com','2001-05-15','2024-02-10');
137
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
1	Prince	Prajapati	prince1@gmail.com	2001-01-12	2022-08-01
2	Aarav	Shah	aarav@gmail.com	2000-02-15	2021-08-01
3	Riya	Patel	riya@gmail.com	2001-03-18	2023-01-10
4	Neel	Mehta	neel@gmail.com	1999-04-20	2020-08-01
5	Khushi	Joshi	khushi@gmail.com	2002-05-25	2023-08-01
6	Yash	Desai	yash@gmail.com	2001-06-10	2022-08-01
7	Nidhi	Sharma	nidhi@gmail.com	2000-07-05	2021-08-01
8	Harsh	Verma	harsh@gmail.com	2001-08-09	2023-08-01
9	Pooja	Rathod	pooja@gmail.com	1999-09-22	2020-08-01
10	Meet	Gandhi	meet@gmail.com	2002-10-30	2022-08-01
11	PRINCE	PRAJAPATI	PRINCE@gmail.com	2001-05-15	2024-02-10
NULL	NULL	NULL	NULL	NULL	NULL

```
141     -- UPDATE
142     UPDATE Students
143     SET FirstName='RAJ'
144     WHERE StudentID=11;
145
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
1	Prince	Prajapati	prince1@gmail.com	2001-01-12	2022-08-01
2	Aarav	Shah	aarav@gmail.com	2000-02-15	2021-08-01
3	Riya	Patel	riya@gmail.com	2001-03-18	2023-01-10
4	Neel	Mehta	neel@gmail.com	1999-04-20	2020-08-01
5	Khushi	Joshi	khushi@gmail.com	2002-05-25	2023-08-01
6	Yash	Desai	yash@gmail.com	2001-06-10	2022-08-01
7	Nidhi	Sharma	nidhi@gmail.com	2000-07-05	2021-08-01
8	Harsh	Verma	harsh@gmail.com	2001-08-09	2023-08-01
9	Pooja	Rathod	pooja@gmail.com	1999-09-22	2020-08-01
10	Meet	Gandhi	meet@gmail.com	2002-10-30	2022-08-01
11	RAJ	PRAJAPATI	PRINCE@gmail.com	2001-05-15	2024-02-10
NULL	NULL	NULL	NULL	NULL	NULL

```sql
-- DELETE
DELETE FROM Students
WHERE StudentID=11;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
1	Prince	Prajapati	prince1@gmail.com	2001-01-12	2022-08-01
2	Aarav	Shah	aarav@gmail.com	2000-02-15	2021-08-01
3	Riya	Patel	riya@gmail.com	2001-03-18	2023-01-10
4	Neel	Mehta	neel@gmail.com	1999-04-20	2020-08-01
5	Khushi	Joshi	khushi@gmail.com	2002-05-25	2023-08-01
6	Yash	Desai	yash@gmail.com	2001-06-10	2022-08-01
7	Nidhi	Sharma	nidhi@gmail.com	2000-07-05	2021-08-01
8	Harsh	Verma	harsh@gmail.com	2001-08-09	2023-08-01
9	Pooja	Rathod	pooja@gmail.com	1999-09-22	2020-08-01
10	Meet	Gandhi	meet@gmail.com	2002-10-30	2022-08-01
NULL	NULL	NULL	NULL	NULL	NULL

```sql
-- 3. CRUD ON Courses
-- CREATE
INSERT INTO Courses VALUES (111,'Deep Learning',1,4);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CourseID	CourseName	DepartmentID	Credits
101	Introduction to SQL	1	3
102	Python Programming	2	4
103	Data Structures	1	3
104	Linear Algebra	5	4
105	Discrete Mathematics	3	3
106	AI/ML	4	3
107	DATA Analyst	1	4
108	Data science	5	3
109	Cyber Security	2	3
110	LLM	4	4
111	Deep Learning	1	4
NULL	NULL	NULL	NULL

```sql
-- UPDATE
UPDATE Courses
SET Credits=5
WHERE CourseID=111;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CourseID	CourseName	DepartmentID	Credits
101	Introduction to SQL	1	3
102	Python Programming	2	4
103	Data Structures	1	3
104	Linear Algebra	5	4
105	Discrete Mathematics	3	3
106	AI/ML	4	3
107	DATA Analyst	1	4
108	Data science	5	3
109	Cyber Security	2	3
110	LLM	4	4
111	Deep Learning	1	5
NULL	NULL	NULL	NULL

```sql
-- DELETE
DELETE FROM Courses
WHERE CourseID=111;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CourseID	CourseName	DepartmentID	Credits
101	Introduction to SQL	1	3
102	Python Programming	2	4
103	Data Structures	1	3
104	Linear Algebra	5	4
105	Discrete Mathematics	3	3
106	AI/ML	4	3
107	DATA Analyst	1	4
108	Data science	5	3
109	Cyber Security	2	3
110	LLM	4	4
NULL	NULL	NULL	NULL

```sql
-- 4. CRUD ON Instructors
-- CREATE
INSERT INTO Instructors VALUES
(11,'GOPAL','MAKWANA','GOPAL@univ.com',1,70000);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

InstructorID	FirstName	LastName	Email	DepartmentID	Salary
1	Ramesh	Iyer	ramesh@univ.com	1	75000
2	Anita	Sharma	anita@univ.com	2	72000
3	Suresh	Patel	suresh@univ.com	3	70000
4	Vikas	Mehta	vikas@univ.com	4	68000
5	Neha	Joshi	neha@univ.com	5	65000
6	kalpesh	goswami	kalpeshh@univ.com	1	75000
7	raju	Sharma	raju@univ.com	2	72000
8	Ranjeet	Patel	Ranjeet@univ.com	3	70000
9	shiv	Rathod	shiv@univ.com	4	68000
10	Priyanshu	Lakhani	Priyanshu@univ.com	5	65000
11	GOPAL	MAKWANA	GOPAL@univ.com	1	70000
NULL	NULL	NULL	NULL	NULL	NULL

```sql
-- UPDATE
UPDATE Instructors
SET Salary=95000
WHERE InstructorID=11;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

InstructorID	FirstName	LastName	Email	DepartmentID	Salary
1	Ramesh	Iyer	ramesh@univ.com	1	75000
2	Anita	Sharma	anita@univ.com	2	72000
3	Suresh	Patel	suresh@univ.com	3	70000
4	Vikas	Mehta	vikas@univ.com	4	68000
5	Neha	Joshi	neha@univ.com	5	65000
6	kalpesh	goswami	kalpeshh@univ.com	1	75000
7	raju	Sharma	raju@univ.com	2	72000
8	Ranjeet	Patel	Ranjeet@univ.com	3	70000
9	shiv	Rathod	shiv@univ.com	4	68000
10	Priyanshu	Lakhani	Priyanshu@univ.com	5	65000
11	GOPAL	MAKWANA	GOPAL@univ.com	1	95000
NULL	NULL	NULL	NULL	NULL	NULL

```sql
-- DELETE
DELETE FROM Instructors
WHERE InstructorID=11;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

InstructorID	FirstName	LastName	Email	DepartmentID	Salary
1	Ramesh	Iyer	ramesh@univ.com	1	75000
2	Anita	Sharma	anita@univ.com	2	72000
3	Suresh	Patel	suresh@univ.com	3	70000
4	Vikas	Mehta	vikas@univ.com	4	68000
5	Neha	Joshi	neha@univ.com	5	65000
6	kalpesh	goswami	kalpeshh@univ.com	1	75000
7	raju	Sharma	raju@univ.com	2	72000
8	Ranjeet	Patel	Ranjeet@univ.com	3	70000
9	shiv	Rathod	shiv@univ.com	4	68000
10	Priyanshu	Lakhani	Priyanshu@univ.com	5	65000
NULL	NULL	NULL	NULL	NULL	NULL

```sql
#  2. Retrieve students enrolled after 2022

SELECT * FROM Students
WHERE EnrollmentDate > '2022-12-31';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
3	Riya	Patel	riya@gmail.com	2001-03-18	2023-01-10
5	Khushi	Joshi	khushi@gmail.com	2002-05-25	2023-08-01
8	Harsh	Verma	harsh@gmail.com	2001-08-09	2023-08-01
NULL	NULL	NULL	NULL	NULL	NULL

```sql
# 3. Retrieve courses offered by the Mathematics department with a limit of 5 courses.

SELECT * FROM Courses
WHERE DepartmentID=2
LIMIT 5;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CourseID	CourseName	DepartmentID	Credits
102	Python Programming	2	4
109	Cyber Security	2	3
NULL	NULL	NULL	NULL

```sql
# 4. Get the number of students enrolled in each course, filtering for courses with more than 5 students

SELECT c.CourseId,c.CourseName, COUNT(e.StudentId) AS Total_Student
FROM Courses AS c
INNER JOIN Enrollments AS e
ON c.CourseId = e.CourseId
GROUP BY c.CourseId, c.CourseName
HAVING COUNT(e.StudentId) > 5;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CourseId	CourseName	Total_Student
101	Introduction to SQL	7

```sql
#  5. Find students who are enrolled in both Introduction to SQL and Data Structures.

SELECT StudentId, FirstName, LastName
FROM Students
WHERE StudentId IN
(SELECT StudentId
    FROM Enrollments
    WHERE CourseId IN (101,102)
    GROUP BY StudentId
    HAVING COUNT(DISTINCT CourseId) = 2);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	FirstName	LastName
2	Aarav	Shah

```sql
# 6. Find students who are either enrolled in Introduction to SQL or Data Structures.

SELECT DISTINCT s.StudentId, s.FirstName, s.LastName
FROM Students AS s
JOIN Enrollments AS e
ON s.StudentId = e.StudentId
WHERE e.CourseId IN (101,102);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	FirstName	LastName
1	Prince	Prajapati
2	Aarav	Shah
3	Riya	Patel
4	Neel	Mehta
5	Khushi	Joshi
6	Yash	Desai
7	Nidhi	Sharma

```
238     #  7. Calculate the average number of credits for all courses.
239
240 •   SELECT AVG(Credits) AS AverageCredits FROM Courses;
241
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

AverageCredits
3.4000

```
242     #  8. Find the maximum salary of instructors in the Computer Science department.
243
244 •   SELECT MAX(i.Salary) AS MAX_SALARY
245     FROM Instructors i
246     JOIN Departments d
247     ON i.DepartmentID = d.DepartmentID
248     WHERE d.DepartmentName = 'Data Science';
249
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

MAX_SALARY
75000

```sql
# 9. Count the number of students enrolled in each department.

SELECT d.DepartmentName, COUNT(DISTINCT e.StudentID) AS TotalStudents
FROM Departments d
JOIN Courses c ON d.DepartmentID = c.DepartmentID
JOIN Enrollments e ON c.CourseID = e.CourseID
GROUP BY d.DepartmentName;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

DepartmentName	TotalStudents
Civil Engineering	2
Data Science	7
Mechanical	2
PBL	1
Statistics	2

```sql
# 10. INNER JOIN: Retrieve students and their corresponding courses.

SELECT s.StudentId,s.FirstName,s.LastName,c.CourseName
FROM Students s
INNER JOIN Enrollments e
ON s.StudentId = e.StudentId
INNER JOIN Courses c
ON e.CourseId = c.CourseId;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	FirstName	LastName	CourseName
1	Prince	Prajapati	Introduction to SQL
2	Aarav	Shah	Python Programming
2	Aarav	Shah	Introduction to SQL
3	Riya	Patel	Data Structures
3	Riya	Patel	Introduction to SQL
4	Neel	Mehta	Linear Algebra
4	Neel	Mehta	Introduction to SQL
5	Khushi	Joshi	Discrete Mathematics
5	Khushi	Joshi	Introduction to SQL
6	Yash	Desai	AI/ML
6	Yash	Desai	Introduction to SQL
7	Nidhi	Sharma	DATA Analyst
7	Nidhi	Sharma	Introduction to SQL
8	Harsh	Verma	Data science
9	Pooja	Rathod	Cyber Security
10	Meet	Gandhi	LLM

```
268     # 11. LEFT JOIN: Retrieve all students and their corresponding courses, if any.
269
270 •   SELECT s.StudentId,s.FirstName,s.LastName,c.CourseName
271     FROM Students s
272     LEFT JOIN Enrollments e
273     ON s.StudentId = e.StudentId
274     LEFT JOIN Courses c
275     ON e.CourseId = c.CourseId;
276
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	FirstName	LastName	CourseName
1	Prince	Prajapati	Introduction to SQL
2	Aarav	Shah	Python Programming
2	Aarav	Shah	Introduction to SQL
3	Riya	Patel	Data Structures
3	Riya	Patel	Introduction to SQL
4	Neel	Mehta	Linear Algebra
4	Neel	Mehta	Introduction to SQL
5	Khushi	Joshi	Discrete Mathematics
5	Khushi	Joshi	Introduction to SQL
6	Yash	Desai	AI/ML
6	Yash	Desai	Introduction to SQL
7	Nidhi	Sharma	DATA Analyst
7	Nidhi	Sharma	Introduction to SQL
8	Harsh	Verma	Data science
9	Pooja	Rathod	Cyber Security
10	Meet	Gandhi	LLM

```sql
# 11. LEFT JOIN: Retrieve all students and their corresponding courses, if any.

SELECT s.StudentId,s.FirstName,s.LastName,c.CourseName
FROM Students s
LEFT JOIN Enrollments e
ON s.StudentId = e.StudentId
LEFT JOIN Courses c
ON e.CourseId = c.CourseId;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	FirstName	LastName	CourseName
1	Prince	Prajapati	Introduction to SQL
2	Aarav	Shah	Python Programming
2	Aarav	Shah	Introduction to SQL
3	Riya	Patel	Data Structures
3	Riya	Patel	Introduction to SQL
4	Neel	Mehta	Linear Algebra
4	Neel	Mehta	Introduction to SQL
5	Khushi	Joshi	Discrete Mathematics
5	Khushi	Joshi	Introduction to SQL
6	Yash	Desai	AI/ML
6	Yash	Desai	Introduction to SQL
7	Nidhi	Sharma	DATA Analyst
7	Nidhi	Sharma	Introduction to SQL
8	Harsh	Verma	Data science
9	Pooja	Rathod	Cyber Security
10	Meet	Gandhi	LLM

```sql
# 12. Subquery: Find students enrolled in courses that have more than 5 students

SELECT StudentID, FirstName, LastName
FROM Students
WHERE StudentID IN (
    SELECT e.StudentID
    FROM Enrollments e
    WHERE e.CourseID IN (
        SELECT CourseID
        FROM Enrollments
        GROUP BY CourseID
        HAVING COUNT(StudentID) > 5
        )
);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

StudentID	FirstName	LastName
1	Prince	Prajapati
2	Aarav	Shah
3	Riya	Patel
4	Neel	Mehta
5	Khushi	Joshi
6	Yash	Desai
7	Nidhi	Sharma
NULL	NULL	NULL

```
292     # 13. Extract the year from the EnrollmentDate of students.
293
294 •   SELECT StudentId,FirstName,LastName,
295     YEAR(EnrollmentDate) AS Enroll_year
296     FROM Students;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentId	FirstName	LastName	Enroll_year
1	Prince	Prajapati	2022
2	Aarav	Shah	2021
3	Riya	Patel	2023
4	Neel	Mehta	2020
5	Khushi	Joshi	2023
6	Yash	Desai	2022
7	Nidhi	Sharma	2021
8	Harsh	Verma	2023
9	Pooja	Rathod	2020
10	Meet	Gandhi	2022

```
298     # 14. Concatenate the instructor's first and last name.
299
300 •   SELECT InstructorId, CONCAT(FirstName,' ',LastName) AS Fullname
301     FROM Instructors;
302
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

InstructorId	Fullname
1	Ramesh Iyer
2	Anita Sharma
3	Suresh Patel
4	Vikas Mehta
5	Neha Joshi
6	kalpesh goswami
7	raju Sharma
8	Ranjeet Patel
9	shiv Rathod
10	Priyanshu Lakhani

```sql
303     # 15. Calculate the running total of students enrolled in courses.
304
305     SELECT  EnrollmentID, CourseID, COUNT(StudentID)
306     OVER (ORDER BY EnrollmentID) AS RunningTotalStudents
307     FROM Enrollments;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

EnrollmentID	CourseID	RunningTotalStudents
1	101	1
2	102	2
3	103	3
4	104	4
5	105	5
6	106	6
7	107	7
8	108	8
9	109	9
10	110	10
11	101	11
12	101	12
13	101	13
14	101	14
15	101	15
16	101	16

```sql
309     # 16. Label students as 'Senior' or 'Junior' based on their year of enrollment.
310
311     SELECT StudentID, FirstName, LastName, EnrollmentDate,
312     CASE
313     WHEN YEAR(EnrollmentDate) <= 2022 THEN 'Senior'
314     ELSE 'Junior'
315     END AS StudentLevel
316     FROM Students;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentID	FirstName	LastName	EnrollmentDate	StudentLevel
1	Prince	Prajapati	2022-08-01	Senior
2	Aarav	Shah	2021-08-01	Senior
3	Riya	Patel	2023-01-10	Junior
4	Neel	Mehta	2020-08-01	Senior
5	Khushi	Joshi	2023-08-01	Junior
6	Yash	Desai	2022-08-01	Senior
7	Nidhi	Sharma	2021-08-01	Senior
8	Harsh	Verma	2023-08-01	Junior
9	Pooja	Rathod	2020-08-01	Senior
10	Meet	Gandhi	2022-08-01	Senior