Query 1   ASSIGNMENT 1   **ASSIGNMENT 2** ×

SCHEMAS

Filter objects

- org
- org1
- **studentmanagement**
  - Tables
    - enrolments
    - courses
    - students
  - Views
  - Stored Procedures
  - Functions
- temp
- temp2
- temp3

Limit to 1000 rows

```sql
1    USE STUDENTMANAGEMENT;
2
3    -- CREATE COURSES TBALE
4    CREATE TABLE COURSES (
5        COURSE_ID INT AUTO_INCREMENT PRIMARY KEY,
6        COURSE_NAME VARCHAR(50),
7        COURSE_DESCRIPTION TEXT
8    );
9
10   -- CREATE ENROLMETS TABLE
11   CREATE TABLE ENROLMENTS (
12       ENROLMENT_ID INT AUTO_INCREMENT PRIMARY KEY,
13       STUDENT_ID INT,
14       COURSE_ID INT,
15       ENROLMENT_DATE DATE,
16       FOREIGN KEY (STUDENT_ID) REFERENCES STUDENTS(STUDENT_ID),
17       FOREIGN KEY (COURSE_ID) REFERENCES COURSES(COURSE_ID)
18   );
19
20
21
```

Administration   Schemas

Information

No object selected

Output

Action Output

#	Time	Action	Message
✓ 1	20:58:06	USE STUDENTMANAGEMENT	0 row(s) affected
✓ 2	20:58:20	CREATE TABLE COURSES (COURSE_ID INT AUTO_INCREMENT PRIMARY KEY, COURSE_NAME VA...	0 row(s) affected
✓ 3	20:58:45	CREATE TABLE ENROLMENTS (ENROLMENT_ID INT AUTO_INCREMENT PRIMARY KEY, STUDENT...	0 row(s) affected

---

Navigator

Query 1   ASSIGNMENT 1   **ASSIGNMENT 2*** ×

SCHEMAS

Filter objects

- org
- org1
- **studentmanagement**
  - Tables
    - enrolments
    - courses
    - students
  - Views
  - Stored Procedures
  - Functions
- temp
- temp2
- temp3

Limit to 1000 rows

```sql
17       FOREIGN KEY (COURSE_ID) REFERENCES COURSES(COURSE_ID)
18   );
19
20   -- INSERT DATA INTO COURSES TABLE
21   INSERT INTO COURSES (COURSE_NAME, COURSE_DESCRIPTION) VALUES
22   ('Mathematics', 'Advanced Algebra and Calculus'),
23   ('Physics', 'Mechanics and Thermodynamics'),
24   ('English Literature', 'Study of Classic and Modern Literature'),
25   ('Biology', 'Cellular Biology and Genetics'),
26   ('Computer Science', 'Introduction to Programming');
27
28   SELECT * FROM COURSES;
29
```

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

COURSE_ID	COURSE_NAME	COURSE_DESCRIPTION
1	Mathematics	Advanced Algebra and Calculus
2	Physics	Mechanics and Thermodynamics
3	English Literature	Study of Classic and Modern Literature
4	Biology	Cellular Biology and Genetics
5	Computer Science	Introduction to Programming
*	NULL	NULL	NULL

COURSES 1 ×

Administration   Schemas

Information

No object selected

Output

Action Output

#	Time	Action
✓ 1	20:58:06	USE STUDENTMANAGEMENT
✓ 2	20:58:20	CREATE TABLE COURSES (COURSE_ID INT AUTO_INCREMENT PRIMARY KEY, COURSE_NAME V...
✓ 3	20:58:45	CREATE TABLE ENROLMENTS (ENROLMENT_ID INT AUTO_INCREMENT PRIMARY KEY, STUDEN...
✓ 4	21:01:16	INSERT INTO COURSES (COURSE_NAME, COURSE_DESCRIPTION) VALUES ('Mathematics', 'Advanced Al...
✓ 5	21:01:24	SELECT * FROM COURSES LIMIT 0, 1000

Object Info   Session

Query 1   ASSIGNMENT 1   ASSIGNMENT 2* ✕

Limit to 1000 rows

```
35        (2, 2, '2025-01-11'),
36        (3, 1, '2025-01-10'),
37        (3, 3, '2025-01-12'),
38        (4, 2, '2025-01-11'),
39        (5, 4, '2025-01-13'),
40        (6, 3, '2025-01-12'),
41        (6, 1, '2025-01-10'),
42        (7, 2, '2025-01-11'),
43        (8, 4, '2025-01-13'),
44        (9, 1, '2025-01-10'),
45        (10, 3, '2025-01-12');
46
47  ●  SELECT * FROM ENROLMENTS;
```

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

ENROLMENT_ID	STUDENT_ID	COURSE_ID	ENROLMENT_DATE
5	3	1	2025-01-10
6	3	3	2025-01-12
7	4	2	2025-01-11
8	5	4	2025-01-13
9	6	3	2025-01-12
10	6	1	2025-01-10
11	7	2	2025-01-11
12	8	4	2025-01-13
13	9	1	2025-01-10
14	10	3	2025-01-12
NULL	NULL	NULL	NULL

ENROLMENTS 2 ✕

Output

Action Output

#	Time	Action
✅	6 21:04:40	INSERT INTO ENROLMENTS (STUDENT_ID, COURSE_ID, ENROLMENT_DATE) VALUES (1, 1, '2025-01-...
✅	7 21:04:48	SELECT * FROM ENROLMENTS LIMIT 0, 1000

Query 1   ASSIGNMENT 1   ASSIGNMENT 2* ✕

Limit to 1000 rows

```
48
49    -- QUERY 1 (LIST ALL STUDENTS AND THE COURSES THEY ARE ENROLLED IN)
50    -- USES INNER JOIN TO INCLUDE ONLY STUDENTS WITHT ENROLMENTS
51
52  ●  SELECT s.NAME, c.COURSE_NAME
53    FROM STUDENTS s
54    INNER JOIN ENROLMENTS e ON s.STUDENT_ID = e.STUDENT_ID
55    INNER JOIN COURSES c ON e.COURSE_ID = c.COURSE_ID
56    ORDER BY s.NAME, c.COURSE_NAME;
57
```

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

NAME	COURSE_NAME
ALICE	English Literature
ALICE	Mathematics
BOB	Mathematics
BOB	Physics
CHARLIE	English Literature
CHARLIE	Mathematics
DIANA	Physics
Ethan	Biology
Fiona	English Literature
Fiona	Mathematics
George	Physics
Hannah	Biology
Ian	Mathematics
Jane	English Literature

Result 3 ✕

Output

Action Output

#	Time	Action
✅	7 21:04:48	SELECT * FROM ENROLMENTS LIMIT 0, 1000
✅	8 21:06:52	SELECT s.NAME, c.COURSE_NAME FROM STUDENTS s INNER JOIN ENROLMENTS e ON s.STUDENT...

Query 1   ASSIGNMENT 1   ASSIGNMENT 2* ×

Limit to 1000 rows

```sql
57
58    -- QUERY 2 (FIND THE NUMBER OF STUDENTS ENROLLED IN EACH  COURSE)
59    -- USES LEFT JOIN TO INCLUDE ALL COURSES, EVEN THOSE WITHOUT ENROLMENTS
60
61 ●  SELECT c.COURSE_ID, c.COURSE_NAME, COUNT(e.STUDENT_ID) as STUDENT_COUNT
62    FROM COURSES c
63    LEFT JOIN ENROLMENTS e ON c.COURSE_ID = e.COURSE_ID
64    GROUP BY c.COURSE_ID, c.COURSE_NAME
65    ORDER BY c.COURSE_ID;
66
```

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

COURSE_ID	COURSE_NAME	STUDENT_COUNT
1	Mathematics	5
2	Physics	3
3	English Literature	4
4	Biology	2
5	Computer Science	0

Query 1   ASSIGNMENT 1   ASSIGNMENT 2* ×

Limit to 1000 rows

```sql
67    -- QUERY 3 (LIST STUDENTS WHO HAVE ENROLLED IN MORE THAN ONE COURSE)
68    -- USES GROUP BY AND HAVING TO FILTER STUDENTS WITH MORE THAN ONE COURSE ENROLMENT
69
70 ●  SELECT s.STUDENT_ID, s.NAME, COUNT(e.COURSE_ID) as COURSE_COUNT
71    FROM STUDENTS s
72    INNER JOIN ENROLMENTS e ON s.STUDENT_ID = e.STUDENT_ID
73    GROUP BY s.STUDENT_ID, s.NAME
74    HAVING COUNT(e.COURSE_ID) > 1
75    ORDER BY s.STUDENT_ID;
76
```

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

STUDENT_ID	NAME	COURSE_COUNT
1	ALICE	2
2	BOB	2
3	CHARLIE	2
6	Fiona	2

SCHEMAS

Filter objects

- ▶ 🗄 org
- ▶ 🗄 org1
- ▼ 🗄 **studentmanagement**
  - ▼ 📑 Tables
    - ▶ 🗐 enrolments
    - ▶ 🗐 courses
    - ▶ 🗐 students
  - 📑 Views
  - 📑 Stored Procedures
  - 📑 Functions
- ▶ 🗄 temp
- ▶ 🗄 temp2
- ▶ 🗄 temp3

Query 1   ASSIGNMENT 1   ASSIGNMENT 2 ✕

Search table data for text in objects selected in the sidebar schema tree

```sql
77    -- QUERY 4 (FIND COURSES WITH NO ENROLLRD STUDENTS)
78    -- USES LEFT JOIN AND WHERE CLAUSE TO FIND COURSES WITH NO MATCHING ENROLMENTS
79
80 ●  SELECT c.COURSE_ID, c.COURSE_NAME
81    FROM COURSES c
82    LEFT JOIN ENROLMENTS e ON c.COURSE_ID = e.COURSE_ID
83    WHERE e.ENROLMENT_ID IS NULL
84    ORDER BY c.COURSE_ID;
85
```

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

COURSE_ID	COURSE_NAME
▶ 5	Computer Science