```
48 •   select * from employee;
```

**Result Grid** | Filter Rows: [          ] | E

| Emp_ID | Emp_Name | Emp_No |
|--------|----------|--------|
| ▶ 101  | Ashish Kaktan | 9450425345 |
| 102    | Raj Choudhary | 8462309621 |
| 103    | Vivek Oberoi | 7512309034 |
| 104    | Shantanu Khandelwal | 9020330023 |
| 105    | Khanak Desai | 8451004522 |
| * NULL | NULL | NULL |

```
66 •       select * from Employment;
67
```

Result Grid | Filter Rows: [   ] | Edit: 🖉 📑 📑 |

Emp_ID	Emp_Profile	Emp_Country	Emp_Join_Date
▶ 101	Content Writer	Germany	2021-04-20
104	Data Analyst	India	2022-12-11
105	Software Engineer	India	2022-01-03
108	Development Executive	Europe	2023-02-15
109	Marketing Manager	Mexico	2020-05-23
* NULL	NULL	NULL	NULL

Join these two tables together using a SELECT statement,

```
68 •   SELECT Employee.Emp_ID, Emp_Name, Emp_No, Emp_Profile, Emp_Country
69     FROM Employee, Employment
70     WHERE Employee.Emp_ID = Employment.Emp_ID;
71
72
```

Result Grid | Filter Rows: [   ] | Export: 📇 | Wrap Cell Content: 🗛

Emp_ID	Emp_Name	Emp_No	Emp_Profile	Emp_Country
▶ 101	Ashish Kaktan	9450425345	Content Writer	Germany
104	Shantanu Khandelwal	9020330023	Data Analyst	India
105	Khanak Desai	8451004522	Software Engineer	India

INNER JOIN on these two tables using a SELECT statement

```sql
73 ● SELECT Emp_Name, Emp_No, Emp_Profile, Emp_Country, Emp_Join_Date
74   FROM Employee INNER JOIN Employment
75   ON Employee.Emp_ID = Employment.Emp_ID;
76
```

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

Emp_Name	Emp_No	Emp_Profile	Emp_Country	Emp_Join_Date
▶ Ashish Kaktan	9450425345	Content Writer	Germany	2021-04-20
Shantanu Khandelwal	9020330023	Data Analyst	India	2022-12-11
Khanak Desai	8451004522	Software Engineer	India	2022-01-03

LEFT JOIN on these two tables using a SELECT statement

```sql
76
77 ● SELECT Employee.Emp_Name, Employee.Emp_No, Employment.Emp_Profile, Employment.Emp_Country
78   FROM Employee LEFT JOIN Employment
79   ON Employee.Emp_ID=Employment.Emp_ID;
80
```

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

Emp_Name	Emp_No	Emp_Profile	Emp_Country
▶ Ashish Kaktan	9450425345	Content Writer	Germany
Raj Choudhary	8462309621	NULL	NULL
Vivek Oberoi	7512309034	NULL	NULL
Shantanu Khandelwal	9020330023	Data Analyst	India
Khanak Desai	8451004522	Software Engineer	India

LEFT JOIN on these two tables using a SELECT statement

```sql
80
81 ● SELECT Employee.Emp_Name, Employee.Emp_No, Employment.Emp_Profile, Employment.Emp_Join_Date
82   FROM Employee LEFT JOIN Employment
83   ON Employee.Emp_ID=Employment.Emp_ID;
84
```

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

Emp_Name	Emp_No	Emp_Profile	Emp_Join_Date
▶ Ashish Kaktan	9450425345	Content Writer	2021-04-20
Raj Choudhary	8462309621	NULL	NULL
Vivek Oberoi	7512309034	NULL	NULL
Shantanu Khandelwal	9020330023	Data Analyst	2022-12-11
Khanak Desai	8451004522	Software Engineer	2022-01-03

RIGHT JOIN on these two tables using a SELECT statement

```
04
85 •    SELECT Employee.Emp_Name, Employee.Emp_No, Employment.Emp_Profile, Employment.Emp_Join_Date
86      FROM Employee RIGHT JOIN Employment
87      ON Employee.Emp_ID=Employment.Emp_ID;
88
```

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

Emp_Name	Emp_No	Emp_Profile	Emp_Join_Date
Ashish Kaktan	9450425345	Content Writer	2021-04-20
Shantanu Khandelwal	9020330023	Data Analyst	2022-12-11
Khanak Desai	8451004522	Software Engineer	2022-01-03
NULL	NULL	Development Executive	2023-02-15
NULL	NULL	Marketing Manager	2020-05-23

3rd Table

```
104
105 •    select * from EmpDetail;
106
```

Result Grid | Filter Rows: | Edit: | Export/Ir

Emp_ID	Emp_Country	Emp_Email	Emp_JoinDate
101	Germany	ashish@scaler.com	2021-04-20
104	India	shantanu@scaler.com	2022-12-11
105	India	khanak@scaler.com	2022-01-03
108	Mexico	nikita@scaler.com	2020-05-23
109	Europe	akshay@scaler.com	2023-02-15
NULL	NULL	NULL	NULL

SQL query to join three tables

```
107 •    SELECT e.Emp_Name, e.Emp_No, m.Emp_Profile, m.Emp_Country, d.Emp_JoinDate
108      FROM Employee e
109      INNER JOIN Employment m ON e.Emp_ID = m.Emp_ID
110      INNER JOIN EmpDetail d ON d.Emp_ID = e.Emp_ID;
111
```

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

Emp_Name	Emp_No	Emp_Profile	Emp_Country	Emp_JoinDate
Ashish Kaktan	9450425345	Content Writer	Germany	2021-04-20
Shantanu Khandelwal	9020330023	Data Analyst	India	2022-12-11
Khanak Desai	8451004522	Software Engineer	India	2022-01-03

How should data be structured to support Join Operations in a many-to-many relationship?

natural join

```
111
112 •    SELECT * FROM Employee NATURAL JOIN Employment;
```

Emp_ID	Emp_Name	Emp_No	Emp_Profile	Emp_Country	Emp_Join_Date
101	Ashish Kaktan	9450425345	Content Writer	Germany	2021-04-20
104	Shantanu Khandelwal	9020330023	Data Analyst	India	2022-12-11
105	Khanak Desai	8451004522	Software Engineer	India	2022-01-03

Equi Join

```
114 •    SELECT Emp_Name, Emp_Country FROM Employee, Employment WHERE Employee.Emp_ID = Employment.Emp_ID;
```

Emp_Name	Emp_Country
Ashish Kaktan	Germany
Shantanu Khandelwal	India
Khanak Desai	India

Non-Equi Join

```
116 •   SELECT Emp_Name, Emp_Country FROM Employee, Employment WHERE Employee.Emp_ID != Employment.Emp_ID;
```

Emp_Name	Emp_Country
Khanak Desai	Germany
Shantanu Khandelwal	Germany
Vivek Oberoi	Germany
Raj Choudhary	Germany
Khanak Desai	India
Vivek Oberoi	India
Raj Choudhary	India
Ashish Kaktan	India
Shantanu Khandelwal	India
Vivek Oberoi	India
Raj Choudhary	India
Ashish Kaktan	India
Khanak Desai	Europe
Shantanu Khandelwal	Europe
Vivek Oberoi	Europe
Raj Choudhary	Europe
Ashish Kaktan	Europe
Khanak Desai	Mexico
Shantanu Khandelwal	Mexico
Vivek Oberoi	Mexico
Raj Choudhary	Mexico
Ashish Kaktan	Mexico

Result 24 ✕

Cross Join

```
117
118 •  SELECT * FROM Employee CROSS JOIN Employment;
119
```

Emp_ID	Emp_Name	Emp_No	Emp_ID	Emp_Profile	Emp_Country	Emp_Join_Date
104	Shantanu Khandelwal	9020330023	101	Content Writer	Germany	2021-04-20
103	Vivek Oberoi	7512309034	101	Content Writer	Germany	2021-04-20
102	Raj Choudhary	8462309621	101	Content Writer	Germany	2021-04-20
101	Ashish Kaktan	9450425345	101	Content Writer	Germany	2021-04-20
105	Khanak Desai	8451004522	104	Data Analyst	India	2022-12-11
104	Shantanu Khandelwal	9020330023	104	Data Analyst	India	2022-12-11
103	Vivek Oberoi	7512309034	104	Data Analyst	India	2022-12-11
102	Raj Choudhary	8462309621	104	Data Analyst	India	2022-12-11
101	Ashish Kaktan	9450425345	104	Data Analyst	India	2022-12-11
105	Khanak Desai	8451004522	105	Software Engi...	India	2022-01-03
104	Shantanu Khandelwal	9020330023	105	Software Engi...	India	2022-01-03
103	Vivek Oberoi	7512309034	105	Software Engi...	India	2022-01-03
102	Raj Choudhary	8462309621	105	Software Engi...	India	2022-01-03
101	Ashish Kaktan	9450425345	105	Software Engi...	India	2022-01-03
105	Khanak Desai	8451004522	108	Development ...	Europe	2023-02-15
104	Shantanu Khandelwal	9020330023	108	Development ...	Europe	2023-02-15
103	Vivek Oberoi	7512309034	108	Development ...	Europe	2023-02-15
102	Raj Choudhary	8462309621	108	Development ...	Europe	2023-02-15
101	Ashish Kaktan	9450425345	108	Development ...	Europe	2023-02-15
105	Khanak Desai	8451004522	109	Marketing Ma...	Mexico	2020-05-23
104	Shantanu Khandelwal	9020330023	109	Marketing Ma...	Mexico	2020-05-23
103	Vivek Oberoi	7512309034	109	Marketing Ma...	Mexico	2020-05-23
102	Raj Choudhary	8462309621	109	Marketing Ma...	Mexico	2020-05-23
101	Ashish Kaktan	9450425345	109	Marketing Ma...	Mexico	2020-05-23

Result 25 ✕

Q2) Create Student table with the fields (rollno,sname,saddress,gender) apply all possible constraints on Student table

```
121
122 •  ⊖  CREATE TABLE Student (
123            rollno INT PRIMARY KEY,
124            sname VARCHAR(50) NOT NULL,
125            saddress VARCHAR(255),
126            gender ENUM('Male', 'Female', 'Other') NOT NULL,
127            UNIQUE KEY (sname), -- Ensures unique student names
128            CHECK (gender IN ('Male', 'Female', 'Other')), -- Ensures gender is one of the specified values
129            CONSTRAINT chk_rollno_positive CHECK (rollno > 0) -- Ensures rollno is a positive number
130        );
131 •    select * from Student;
```

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

rollno	sname	saddress	gender
NULL	NULL	NULL	NULL

Student 26 ×

Output

Action Output ▾

#	Time	Action	Message
✅ 57	18:23:24	SELECT * FROM Employee CROSS JOIN Employment LIMIT 0, 1000	25 row(s) returned
✅ 58	18:29:48	CREATE TABLE Student (rollno INT PRIMARY KEY, sname VARCHAR(50) NOT NULL, saddress VA...	0 row(s) affected
✅ 59	18:30:54	select * from Student LIMIT 0, 1000	0 row(s) returned