

Q1. Create a table named students with fields:

- stdid INT PRIMARY KEY
- stdname VARCHAR(50)
- age INT
- city VARCHAR(50)

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database schema with several other databases listed.
- SQL Editor:** Contains the following SQL code:
 

```

1 • create database ITPR;
2 • use ITPR;
3 • create table students(
4     stdid INT PRIMARY KEY,
5     stdname VARCHAR(50),
6     age INT,
7     city VARCHAR(50)
8 );
9 • desc students;
10 • select * from students;
      
```
- Result Grid:** Displays the structure of the 'students' table with columns: stdid, stdname, age, and city. A single row is present with all values set to NULL.
- Output Window:** Shows the execution history with the following log entries:
 

#	Time	Action	Message	Duration / Fetch
1	15:32:47	create database ITPR	1 row(s) affected	0.063 sec
2	15:32:58	use ITPR	0 row(s) affected	0.000 sec
3	15:34:12	create table students/stdid INT PRIMARY KEY stdname VARCHAR(50) age INT city VA	Error Code: 1064. You have an error in your SQL syntax; check the manual that came with MySQL for the right syntax to use near 'city VA' at line 1	0.015 sec

Q2. Insert the following records into the students table: stdid stdname age city

1	Rohan	20	Pune
2	Meera	22	Mumbai
3	Arjun	21	Delhi
4	Kavya	23	Pune
5	Neha	22	Kolkata

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database schema with several other databases listed.
- SQL Editor:** Contains the following SQL code:
 

```

8 );
9 • desc students;
10 • select * from students;
11 • insert into students values
12     (1 , 'Rohan',20,'Pune'),
13     (2 , 'Meera',22,'Mumbai'),
14     (3 , 'Arjun',21,'Delhi'),
15     (4 , 'Kavya',23,'Pune'),
16     (5 , 'Neha',22,'Kolkata');
17 • select * from students;
      
```
- Result Grid:** Displays the contents of the 'students' table with the inserted records.
- Output Window:** Shows the execution history with the following log entries:
 

#	Time	Action	Message	Duration / Fetch
7	15:42:24	insert into students values (1 , 'Rohan',20,'Pune'), (2 , 'Meera',22,'Mumbai'), (3 , 'Arjun',21,'Delhi'), (4 , 'Kavya',23,'Pune'), (5 , 'Neha',22,'Kolkata')	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.031 sec
8	15:42:42	select * from students LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

### Q3. Display all student records.

The screenshot shows the MySQL Workbench interface with the following details:

- SQL Editor:** Contains the following SQL code:

```
9 • desc students;
10 • select * from students;
11 • insert into students values
12 (1 , 'Rohan',20,'Pune'),
13 (2,'Meera',22,'Mumbai'),
14 (3,'Arjun',21,'Delhi'),
15 (4,'Kavya',23,'Pune'),
16 (5,'Neha',22,'Kolkata');
17 • select * from students;
18 • select stdname , age from students ;
```
- Result Grid:** Displays the results of the last query:

stdname	age
Rohan	20
Meera	22
Arjun	21
Kavya	23
Neha	22
- Output Panel:** Shows the execution log:

#	Time	Action	Message	Duration / Fetch
8	15:42:42	select *from students LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
9	15:45:22	select stdname , age from students LIMIT 0, 1000	5 row(s) returned	0.015 sec / 0.000 sec

### Q4. Display only the name and age of all students.

The screenshot shows the MySQL Workbench interface with the following details:

- SQL Editor:** Contains the following SQL code:

```
18 (5,'Neha',22,'Kolkata');
19 -- Q3. Display all student records.
20 • select * from students;
21 -- Q4. Display only the name and age of all students.
22 • select stdname , age from students ;
23 -- Q5. Display students who are from Pune.
24 • select * from students where city="pune";
25 -- Q6. Display students whose age is greater than 21.
26 • select * from students where age>21;
27 -- Q7. Display students in descending order of age.
```
- Result Grid:** Displays the results of the last query:

stdname	age
Rohan	20
Meera	22
Arjun	21
Kavya	23
- Output Panel:** Shows the execution log:

#	Time	Action	Message	Duration / Fetch
27	16:37:38	select students stdname,marks.marks from students inner join marks on students stdid=marks...	2 row(s) returned	0.000 sec / 0.000 sec
28	16:44:32	select stdname , age from students LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

### Q5. Display students who are from Pune.

The screenshot shows the MySQL Workbench interface. In the SQL Editor tab, the following SQL code is written:

```
10 • select * from students;
11 • insert into students values
12   (1 , 'Rohan',20,'Pune'),
13   (2,'Meera',22,'Mumbai'),
14   (3,'Arjun',21,'Delhi'),
15   (4,'Kavya',23,'Pune'),
16   (5,'Neha',22,'Kolkata');
17 • select * from students;
18 • select stdname , age  from students ;
19 • select * from students where city="pune";
```

The Result Grid shows the following data:

stdid	stdname	age	city
1	Rohan	20	Pune
4	Kavya	23	Pune
*			

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
9	15:45:22	select stdname , age  from students LIMIT 0, 1000	5 row(s) returned	0.015 sec / 0.000 sec
10	15:46:56	select * from students where city="pune" LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

### Q6. Display students whose age is greater than 21

The screenshot shows the MySQL Workbench interface. In the SQL Editor tab, the following SQL code is written:

```
11 • insert into students values
12   (1 , 'Rohan',20,'Pune'),
13   (2,'Meera',22,'Mumbai'),
14   (3,'Arjun',21,'Delhi'),
15   (4,'Kavya',23,'Pune'),
16   (5,'Neha',22,'Kolkata');
17 • select * from students;
18 • select stdname , age  from students ;
19 • select * from students where city="pune";
20 • select * from students where age>21;
```

The Result Grid shows the following data:

stdid	stdname	age	city
2	Meera	22	Mumbai
4	Kavya	23	Pune
5	Neha	22	Kolkata
*			

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
10	15:46:56	select * from students where city="pune" LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
11	15:48:14	select * from students where age>21 LIMIT 0, 1000	3 row(s) returned	0.031 sec / 0.000 sec

### Q7. Display students in descending order of age.

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Schemas: college\_db, employeemana, hospital, student, student\_db, studentmanage, sys

SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* Limit to 1000 rows

```

12   (1 , 'Rohan',20,'Pune'),
13   (2,'Meera',22,'Mumbai'),
14   (3,'Arjun',21,'Delhi'),
15   (4,'Kavya',23,'Pune'),
16   (5,'Neha',22,'Kolkata');
17 • select * from students;
18 • select stdname , age from students ;
19 • select * from students where city="pune";
20 • select * from students where age>21;
21 • select * from students order by age desc;

```

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

No object selected

stdid	stdname	age	city
4	Kavya	23	Pune
2	Meera	22	Mumbai
5	Neha	22	Kolkata
3	Arjun	21	Delhi
1	Rohan	20	Pune

students 7 x

Output

Action	Time	Action	Message	Duration / Fetch
11	15:48:14	select *from students where age>21 LIMIT 0, 1000	3 row(s) returned	0.031 sec / 0.000 sec
12	15:49:15	select *from students order by age desc LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info

Q8. Count how many students belong to each city. (Use GROUP BY)

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Schemas: college\_db, employeemana, hospital, student, student\_db, studentmanage, sys

SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* Limit to 1000 rows

```

24 • select * from students where city="pune";
25   -- Q6. Display students whose age is greater than 21.
26 • select * from students where age>21;
27   -- Q7. Display students in descending order of age.
28 • select * from students order by age desc;
29   -- Q8. Count how many students belong to each city. (Use GROUP BY)
30 • SELECT city, COUNT(*) AS total_students
31   FROM students
32   GROUP BY city;
33   -- Q9. Display students whose name starts with 'K'. (Use LIKE)

```

Administration | Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid Form Editor

No object selected

city	total_students
Pune	2
Mumbai	1
Delhi	1

Result 19 x

Output

Action	Time	Action	Message	Duration / Fetch
28	16:44:32	select stdname , age from students LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
29	16:50:08	SELECT city,COUNT(*) AS total_students FROM students GROUP BY city LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Object Info

### Q9. Display students whose name starts with 'K'. (Use LIKE)

The screenshot shows the MySQL Workbench interface with the following details:

- SQL Editor:** Contains the following SQL code:
 

```

26 •    select * from students where age>21;
27 -- Q7. Display students in descending order of age.
28 •    select * from students order by age desc;
29 -- Q8. Count how many students belong to each city. (Use GROUP BY)
30 •    SELECT city, COUNT(*) AS total_students
31     FROM students
32     GROUP BY city;
33 -- Q9. Display students whose name starts with 'K'. (Use LIKE)
34 •    select * from students where stdname like 'K%';
35
      
```
- Result Grid:** Displays the output of the last query (Q9) with the following data:
 

stdid	stdname	age	city
4	Kavya	23	Pune
*	NULL	NULL	NULL
- Action Output:** Shows the execution log with the following entries:
 

#	Time	Action	Message	Duration / Fetch
13	15:52:10	SELECT city, COUNT(*) AS total_students FROM students GROUP BY city LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
14	15:59:08	select * from students where stdname like 'K%' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

### Q10. Delete student whose stdid = 5.

The screenshot shows the MySQL Workbench interface with the following details:

- SQL Editor:** Contains the following SQL code:
 

```

30 •    SELECT city, COUNT(*) AS total_students
31     FROM students
32     GROUP BY city;
33 -- Q9. Display students whose name starts with 'K'. (Use LIKE)
34 •    select * from students where stdname like 'K%';
35 -- Q10. Delete student whose stdid = 5.
36 •    delete from students where stdid=5;
37
      
```
- Action Output:** Shows the execution log with the following entries:
 

#	Time	Action	Message	Duration / Fetch
17	16:06:02	create table marks(stdid int primary key, subject varchar(300) , marks int )	0 row(s) affected	0.078 sec
18	16:09:27	insert into marks values (1,'Maths',88), (2,'Maths',76), (3,'Maths',92), (5,'Maths',57)	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.032 sec
19	16:09:41	select * from marks LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
20	16:14:13	SELECT students.student_name, marks.marks FROM students INNER JOIN marks ON stud... Error Code: 1054 Unknown column 'students.student_name' in field list'		0.000 sec
21	16:14:41	SELECT students.student_name, marks.marks FROM students INNER JOIN marks ON students... 3 row(s) returned		0.000 sec / 0.000 sec
22	16:15:52	SELECT students.stdid, students.adname, marks.marks FROM students LEFT JOIN marks ... 4 row(s) returned		0.000 sec / 0.000 sec
23	16:16:12	SELECT students.stdname, marks.marks FROM students INNER JOIN marks ON students.a... 3 row(s) returned		0.000 sec / 0.000 sec
24	16:20:01	SELECT students.stdid, students.adname, marks.marks FROM students LEFT JOIN marks ... 4 row(s) returned		0.000 sec / 0.000 sec
25	16:31:16	SELECT marks.subject , students.adname, marks.stdid , marks.marks FROM students Right... 4 row(s) returned		0.000 sec / 0.000 sec
26	16:34:05	select students.stdname,marks.subject , from students cross join marks LIMIT 0, 1000 16 row(s) returned		0.000 sec / 0.000 sec
27	16:37:38	select students.stdname,marks.marks from students inner join marks on students.stdid=marks.stdid 2 row(s) returned		0.000 sec / 0.000 sec
28	16:44:32	select stdname , age from students LIMIT 0, 1000 4 row(s) returned		0.000 sec / 0.000 sec
29	16:50:08	SELECT city, COUNT(*) AS total_students FROM students GROUP BY city LIMIT 0, 1000 3 row(s) returned		0.000 sec / 0.000 sec
30	16:51:14	delete from students where stdid=5 0 row(s) affected		0.000 sec

### INNER JOIN Q11. Display student name and marks of only those students who have matching IDs in both tables.

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* Limit to 1000 rows

```

45     marks in );
46 •   insert into marks values
47     (1,'Maths',88),
48     (2,'Maths',76),
49     (3,'Maths',92),
50     (5,'Maths',67);
51 •   select * from marks;
52
53 •   SELECT students.stdname, marks.marks
      FROM students
      JOIN marks
      ON students.stdid = marks.stdid;
  
```

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

stdname	marks
Rohan	88
Meera	76
Arjun	92

Result 20 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
27	16:37:38	select students.stdname,marks.marks from students inner join marks on students.stdid=marks.stdid;	2 row(s) returned	0.000 sec / 0.000 sec
28	16:44:32	select stdname , age from students LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
29	16:50:08	SELECT city, COUNT(*) AS total_students FROM students GROUP BY city LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
30	16:51:14	delete from students where stdid=5	0 row(s) affected	0.000 sec
31	16:52:36	SELECT students.stdname, marks.marks FROM students INNER JOIN marks ON students.s...	3 row(s) returned	0.000 sec / 0.000 sec

No object selected

Object Info

## Q12. Display all students and their marks

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* Limit to 1000 rows

```

57
58 -- Q12. Display all students and their marks.
59 -- (If marks not available, show NULL.)
60 •   SELECT students.stdid, students.stdname, marks.marks
61   FROM students
62   LEFT JOIN marks
63   ON students.stdid = marks.stdid;
64
65 -- RIGHT JOIN
66   -- Q13. Display all marks records along with student names.
  
```

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

stdid	stdname	marks
1	Rohan	88
2	Meera	76
3	Arjun	92
4	Kavya	NULL

Result 21 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
28	16:44:32	select stdname , age from students LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
29	16:50:08	SELECT city, COUNT(*) AS total_students FROM students GROUP BY city LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
30	16:51:14	delete from students where stdid=5	0 row(s) affected	0.000 sec
31	16:52:36	SELECT students.stdname, marks.marks FROM students INNER JOIN marks ON students.s...	3 row(s) returned	0.000 sec / 0.000 sec
32	16:53:45	SELECT students.stdid, students.stdname, marks.marks FROM students LEFT JOIN marks ...	4 row(s) returned	0.015 sec / 0.000 sec

No object selected

Object Info

## Q13. Display all marks records along with student names.

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Schemas: college\_db, employemana, hospital, student, student\_db, studentmanage, sys

SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* x

Limit to 1000 rows

```

65 -- RIGHT JOIN
66 -- Q13. Display all marks records along with student names.
67 -- (If student doesn't exist in students table, show NULL.)
68
69 • SELECT marks.subject , students.stdname , marks.stdid , marks.marks
70 FROM students
71 Right JOIN marks
72 ON students.stdid = marks.stdid;
73
74 CLOSE CURSOR;
    
```

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

subject	stdname	stdid	marks
Maths	Rohan	1	88
Maths	Meera	2	76
Maths	Arjun	3	92
Maths	Kavya	5	67

Result 22 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
29	16:50:08	SELECT city, COUNT(*) AS total_students FROM students GROUP BY city LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
30	16:51:14	delete from students where stdid=5	0 row(s) affected	0.000 sec
31	16:52:36	SELECT students.stdname, marks.marks FROM students INNER JOIN marks ON students.s...	3 row(s) returned	0.000 sec / 0.000 sec
32	16:53:45	SELECT students.stdid, students.stdname, marks.marks FROM students LEFT JOIN marks ...	4 row(s) returned	0.015 sec / 0.000 sec
33	16:54:59	SELECT marks.subject , students.stdname , marks.stdid , marks.marks FROM students Right...	4 row(s) returned	0.000 sec / 0.000 sec

Object Info

Q14. Display all possible combinations of students and subjects. (Use CROSS JOIN between students and marks table to show every pair.)

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Schemas: college\_db, employemana, hospital, student, student\_db, studentmanage, sys

SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* x

Limit to 1000 rows

```

72 ON students.stdid = marks.stdid;
73
74 -- CROSS JOIN
75 -- Q14. Display all possible combinations of students and subjects.
76 -- (Use CROSS JOIN between students and marks table to show every pair.)
77 • select students.stdname,marks.subject
78 from students
79 cross join marks;
80
81
    
```

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

stdname	subject
Kavya	Maths
Arjun	Maths
Meera	Maths
Rohan	Maths
Kavya	Maths

Result 23 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
30	16:51:14	delete from students where stdid=5	0 row(s) affected	0.000 sec
31	16:52:36	SELECT students.stdname, marks.marks FROM students INNER JOIN marks ON students.s...	3 row(s) returned	0.000 sec / 0.000 sec
32	16:53:45	SELECT students.stdid, students.stdname, marks.marks FROM students LEFT JOIN marks ...	4 row(s) returned	0.015 sec / 0.000 sec
33	16:54:59	SELECT marks.subject , students.stdname , marks.stdid , marks.marks FROM students Right...	4 row(s) returned	0.000 sec / 0.000 sec
34	16:55:53	select students.stdname,marks.subject from students cross join marks LIMIT 0, 1000	16 row(s) returned	0.000 sec / 0.000 sec

Object Info

Q15. Using INNER JOIN, display students who scored more than 80.

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- college\_db
- employemana
- hospital
- student
- student\_db
- studentmanage
- sys

SQL File 1\* SQL File 4\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\*

Limit to 1000 rows

```
75 -- Q14. Display all possible combinations of students and subjects.
76 -- (Use CROSS JOIN between students and marks table to show every pair.)
77 • select students.stdname,marks.subject
78   from students
79   cross join marks;
80
81 -- JOIN with Filtering
82 -- Q15. Using INNER JOIN, display students who scored more than 80.
83 • select students.stdname,marks.marks from students
84
85   where marks >= 80;
```

SQLAdditions

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

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

stdname	marks
Rohan	88
Arjun	92

Result Grid

Result 24 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
31	16:52:36	SELECT students.stdname, marks.marks FROM students INNER JOIN marks ON students.s...	3 row(s) returned	0.000 sec / 0.000 sec
32	16:53:45	SELECT students.stdid, students.stdname, marks.marks FROM students LEFT JOIN marks ...	4 row(s) returned	0.015 sec / 0.000 sec
33	16:54:59	SELECT marks.subject , students.stdname, marks.stdid , marks.marks FROM students Right... <td>4 row(s) returned</td> <td>0.000 sec / 0.000 sec</td>	4 row(s) returned	0.000 sec / 0.000 sec
34	16:55:53	select students.stdname,marks.subject from students cross join marks LIMIT 0, 1000	16 row(s) returned	0.000 sec / 0.000 sec
35	16:56:38	select students.stdname,marks.marks from students inner join marks on students.stdid=marks...	2 row(s) returned	0.015 sec / 0.000 sec

No object selected

Object Info