

NAME: **Sristi Sengupta**

DEPARTMENT: **B.Tech(ECE)**

COLLEGE: **Netaji**

**Subhash**

**Engineering College**

STUDENT ID: **AF0453862**

```
CREATE DATABASE ScoreDB;
```

```
USE ScoreDB;
```

```
CREATE TABLE Student (
```

```
    STUDENT_ID INT PRIMARY KEY,
```

```
    FIRST_NAME VARCHAR(50),
```

```
    LAST_NAME VARCHAR(50),
```

```
    PHONE_NUMBER VARCHAR(15),
```

```
    MARKS INT,
```

```
    COURSE_ID INT
```

```
);
```

```
CREATE TABLE Course (
```

```
    COURSE_ID INT PRIMARY KEY,
```

```
    COURSE_NAME VARCHAR(100)
```

```
);
```

```
INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES
```

```
(101, 'Mathematics'),
```

```
(102, 'Physics'),
```

```
(103, 'Chemistry'),
```

```
(104, 'Biology'),
```

```
(105, 'Computer Science');
```

```
INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS,  
COURSE_ID) VALUES
```

```
(1, 'John', 'Doe', '1234567890', 85, 101),
```

```
(2, 'Jane', 'Smith', '9876543210', 90, 102),
```

```
(3, 'Alice', 'Brown', '1122334455', 88, 103),
```

```
(4, 'Bob', 'Johnson', '9988776655', 76, 101),
```

```
(5, 'Charlie', 'Davis', '5566778899', 92, 104),
```

```
(6, 'Emily', 'Clark', '6677889900', 89, 105),
```

```
(7, 'David', 'Martinez', '2233445566', 79, 102),
```

```
(8, 'Sophia', 'Garcia', '3344556677', 95, 105);
```

```
SELECT COURSE_ID, COUNT(*) AS Student_Count
FROM Student
GROUP BY COURSE_ID;

SELECT
    S.FIRST_NAME || ' ' || S.LAST_NAME AS Student_Name,
    C.COURSE_NAME,
    S.MARKS
FROM
    Student S
JOIN
    Course C
ON
    S.COURSE_ID = C.COURSE_ID;

SELECT
    C.COURSE_NAME,
    AVG(S.MARKS) AS Average_Marks
FROM
    Student S
JOIN
    Course C
ON
    S.COURSE_ID = C.COURSE_ID
GROUP BY
    C.COURSE_NAME;

SELECT * FROM Student;

SELECT * FROM Course;
```

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

sample

Tables

Views

Stored Procs

Functions

studentdb

studentdbcond

sys

testdb

world

Administrative

Information

No object selected

Object Info

employees

Limit to 1000 rows

SQL Editor

```
45 USE ScoreDB;
46 CREATE TABLE Student (
47     STUDENT_ID INT PRIMARY KEY,
48     FIRST_NAME VARCHAR(50),
49     LAST_NAME VARCHAR(50),
50     PHONE_NUMBER VARCHAR(15),
51     MARKS INT,
52     COURSE_ID INT
53 );
54 CREATE TABLE Course (
55     COURSE_ID INT PRIMARY KEY,
56     COURSE_NAME VARCHAR(100)
57 );
```

Result Grid

COURSE_NAME	Average_Marks
Mathematics	80.5000
Physics	84.5000
Chemistry	88.0000
Biology	92.0000
Computer Science	92.0000

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:21:36	CREATE DATABASE ScoreDB	0 rows(s) affected	0.180 sec
2	17:21:36	USE ScoreDB	0 rows(s) affected	0.000 sec
3	17:21:36	CREATE TABLE Student ( STUDENT_ID INT PRIMARY KEY, FIRST_NAME VARCHAR(50), LAST_NAME VARCHAR(50), PHONE_NUMBER VARCHAR(15), MARKS INT, COURSE_ID INT)	0 rows(s) affected	0.188 sec
4	17:21:36	CREATE TABLE Course ( COURSE_ID INT PRIMARY KEY, COURSE_NAME VARCHAR(100))	0 rows(s) affected	0.313 sec
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry'), (104, 'Biology'), (105, 'Computer Science');	5 rows(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VALUES (1, 'John', 'Doe', '1234567890', 85, 101), (2, 'Jane', 'Smith', '9876543210', 90, 102), (3, 'Alice', 'Brown', '1122334455', 88, 103), (4, 'Bob', 'Johnson', '9988776655', 76, 104), (5, 'Charlie', 'Davis', '5566778899', 92, 105);	5 rows(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.031 sec

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

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

sample

Tables

Views

Stored Procs

Functions

studentdb

studentdbcond

sys

testdb

world

Administrative

Information

No object selected

Object Info

employees

Limit to 1000 rows

SQL Editor

```
57 );
58 INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES
59 (101, 'Mathematics'),
60 (102, 'Physics'),
61 (103, 'Chemistry'),
62 (104, 'Biology'),
63 (105, 'Computer Science');
64 INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VALUES
65 (1, 'John', 'Doe', '1234567890', 85, 101),
66 (2, 'Jane', 'Smith', '9876543210', 90, 102),
67 (3, 'Alice', 'Brown', '1122334455', 88, 103),
68 (4, 'Bob', 'Johnson', '9988776655', 76, 104),
69 (5, 'Charlie', 'Davis', '5566778899', 92, 105);
```

Result Grid

COURSE_NAME	Average_Marks
Mathematics	80.5000
Physics	84.5000
Chemistry	88.0000
Biology	92.0000
Computer Science	92.0000

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	17:21:36	CREATE TABLE Course ( COURSE_ID INT PRIMARY KEY, COURSE_NAME VARCHAR(100))	0 rows(s) affected	0.313 sec
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry'), (104, 'Biology'), (105, 'Computer Science');	5 rows(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VALUES (1, 'John', 'Doe', '1234567890', 85, 101), (2, 'Jane', 'Smith', '9876543210', 90, 102), (3, 'Alice', 'Brown', '1122334455', 88, 103), (4, 'Bob', 'Johnson', '9988776655', 76, 104), (5, 'Charlie', 'Davis', '5566778899', 92, 105);	5 rows(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 rows(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S, C	5 rows(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON S.COURSE_ID = C.COURSE_ID	5 rows(s) returned	0.015 sec / 0.000 sec

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

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

sample

Tables

Views

Stored Procs

Functions

studentdb

studentdbcond

sys

testdb

world

Administrative

Information

No object selected

Object Info

employees

Limit to 1000 rows

SQL Editor

```
65 (1, 'John', 'Doe', '1234567890', 85, 101),
66 (2, 'Jane', 'Smith', '9876543210', 90, 102),
67 (3, 'Alice', 'Brown', '1122334455', 88, 103),
68 (4, 'Bob', 'Johnson', '9988776655', 76, 104),
69 (5, 'Charlie', 'Davis', '5566778899', 92, 105),
70 (6, 'Emily', 'Clark', '6677889900', 89, 105),
71 (7, 'David', 'Martinez', '2233445566', 79, 102),
72 (8, 'Sophia', 'Garcia', '3344556677', 95, 105);
73
74 SELECT COURSE_ID, COUNT(*) AS Student_Count
75 FROM Student
76 GROUP BY COURSE_ID;
77 SELECT
```

Result Grid

COURSE_NAME	Average_Marks
Mathematics	80.5000
Physics	84.5000
Chemistry	88.0000
Biology	92.0000
Computer Science	92.0000

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	17:21:36	CREATE TABLE Course ( COURSE_ID INT PRIMARY KEY, COURSE_NAME VARCHAR(100))	0 rows(s) affected	0.313 sec
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry'), (104, 'Biology'), (105, 'Computer Science');	5 rows(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VALUES (1, 'John', 'Doe', '1234567890', 85, 101), (2, 'Jane', 'Smith', '9876543210', 90, 102), (3, 'Alice', 'Brown', '1122334455', 88, 103), (4, 'Bob', 'Johnson', '9988776655', 76, 104), (5, 'Charlie', 'Davis', '5566778899', 92, 105), (6, 'Emily', 'Clark', '6677889900', 89, 105), (7, 'David', 'Martinez', '2233445566', 79, 102), (8, 'Sophia', 'Garcia', '3344556677', 95, 105);	8 rows(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 rows(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S, C	8 rows(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON S.COURSE_ID = C.COURSE_ID	5 rows(s) returned	0.015 sec / 0.000 sec

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

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

sakila  
sample  
Tables  
Views  
Stored Pro  
Functions  
studentsdb  
studentscond  
sys  
testdb  
world

Information

No object selected

employees

Limit to 1000 rows

72 (s, 'sophia', 'Garcia', '73544556677', 95, 105);

73

74 SELECT COURSE\_ID, COUNT(\*) AS Student\_Count

75 FROM Student

76 GROUP BY COURSE\_ID;

77 SELECT

78 S.FIRST\_NAME || ' ' || S.LAST\_NAME AS Student\_Name,

79 C.COURSE\_NAME,

80 S.MARKS

81 FROM

82 Student S

83 JOIN

84 Course C

Result Grid

Filter Rows

Export

Wrap Cell Contents

COURSE_NAME	Average_Marks
Mathematics	80.5000
Physics	84.5000
Chemistry	88.0000
Biology	92.0000
Computer Science	92.0000

Result 1 Result 2 Result 3 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	17:21:36	CREATE TABLE Course ( COURSE_ID INT PRIMARY KEY, COURSE_NAME VARCHAR(100))	0 row(s) affected	0.313 sec
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry')...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VAL...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S...	8 row(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON ...	5 row(s) returned	0.015 sec / 0.000 sec

Object Info

SQL Additions

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

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

sakila  
sample  
Tables  
Views  
Stored Pro  
Functions  
studentsdb  
studentscond  
sys  
testdb  
world

Information

No object selected

employees

Limit to 1000 rows

82 Student S

83 JOIN

84 Course C

85 ON

86 S.COURSE\_ID = C.COURSE\_ID;

87 SELECT

88 C.COURSE\_NAME,

89 AVG(S.MARKS) AS Average\_Marks

90 FROM

91 Student S

92 JOIN

93 Course C

94 ON

Result Grid

Filter Rows

Export

Wrap Cell Contents

COURSE_NAME	Average_Marks
Mathematics	80.5000
Physics	84.5000
Chemistry	88.0000
Biology	92.0000
Computer Science	92.0000

Result 1 Result 2 Result 3 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	17:21:36	CREATE TABLE Course ( COURSE_ID INT PRIMARY KEY, COURSE_NAME VARCHAR(100))	0 row(s) affected	0.313 sec
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry')...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VAL...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S...	8 row(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON ...	5 row(s) returned	0.015 sec / 0.000 sec

Object Info

SQL Additions

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

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

sakila  
sample  
Tables  
Views  
Stored Pro  
Functions  
studentsdb  
studentscond  
sys  
testdb  
world

Information

No object selected

employees

Limit to 1000 rows

90 FROM

91 Student S

92 JOIN

93 Course C

94 ON

95 S.COURSE\_ID = C.COURSE\_ID

96 GROUP BY

97 C.COURSE\_NAME,

98 SELECT \* FROM Student;

99 SELECT \* FROM Course;

100

101

102

Result Grid

Filter Rows

Export

Wrap Cell Contents

COURSE_NAME	Average_Marks
Mathematics	80.5000
Physics	84.5000
Chemistry	88.0000
Biology	92.0000
Computer Science	92.0000

Result 1 Result 2 Result 3 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	17:21:36	CREATE TABLE Course ( COURSE_ID INT PRIMARY KEY, COURSE_NAME VARCHAR(100))	0 row(s) affected	0.313 sec
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry')...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VAL...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S...	8 row(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON ...	5 row(s) returned	0.015 sec / 0.000 sec

Object Info

SQL Additions

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

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

- Filter objects
- sakila
- sample
- Tables
- Views
- Stored Procs
- Functions
- studentdb
- studentdbcore
- sys
- testdb
- world

Administrative

Information

No object selected

Object Info

employees

Limit to 1000 rows

```
90 FROM
91 Student S
92 JOIN
93 Course C
94 ON
95 S.COURSE_ID = C.COURSE_ID
96 GROUP BY
97 C.COURSE_NAME;
98 SELECT * FROM Student;
99 SELECT * FROM Course;
```

Result Grid

STUDENT_ID	FIRST_NAME	LAST_NAME	PHONE_NUMBER	MARKS	COURSE_ID
1	John	Doe	1234567890	85	101
2	Jane	Smith	9876543210	90	102
3	Alice	Brown	1122334455	88	103
4	Bob	Johnson	9988776655	76	101
5	Charlie	Davis	5566778899	92	104

Student 4 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
5	17:21:36	INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (101, 'Mathematics'), (102, 'Physics'), (103, 'Chemistry')...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VAL...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S...	5 row(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON	5 row(s) returned	0.015 sec / 0.000 sec
10	17:24:11	SELECT * FROM Student LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

- Filter objects
- sakila
- sample
- Tables
- Views
- Stored Procs
- Functions
- studentdb
- studentdbcore
- sys
- testdb
- world

Administrative

Information

No object selected

Object Info

employees

Limit to 1000 rows

```
90 FROM
91 Student S
92 JOIN
93 Course C
94 ON
95 S.COURSE_ID = C.COURSE_ID
96 GROUP BY
97 C.COURSE_NAME;
98 SELECT * FROM Student;
99 SELECT * FROM Course;
```

Result Grid

COURSE_ID	COURSE_NAME
101	Mathematics
102	Physics
103	Chemistry
104	Biology
105	Computer Science

Course 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
6	17:21:36	INSERT INTO Student (STUDENT_ID, FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID) VAL...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.031 sec
7	17:21:36	SELECT COURSE_ID, COUNT(*) AS Student_Count FROM Student GROUP BY COURSE_ID LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
8	17:21:36	SELECT S.FIRST_NAME    ' '    S.LAST_NAME AS Student_Name, C.COURSE_NAME, S.MARKS FROM S...	5 row(s) returned	0.000 sec / 0.000 sec
9	17:21:36	SELECT C.COURSE_NAME, AVG(S.MARKS) AS Average_Marks FROM Student S JOIN Course C ON	5 row(s) returned	0.015 sec / 0.000 sec
10	17:24:11	SELECT * FROM Student LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec
11	17:24:27	SELECT * FROM Course LIMIT 0, 1000	5 row(s) returned	0.016 sec / 0.000 sec