



SQL - Project

VIKAS KUMAR
MAURYA

skill
academy
by testbook

1. WRITE A QUERY TO CREATE THE TRIANGLES TABLE.

CREATE DATABASE HACKTHON;

USE HACKTHON;

CREATE TABLE TRIANGLES

(Triangle INT primary KEY,
Side_A int ,
Side_B int,
Side_C int);

Action Output			
#	Time	Action	Message
1	14:40:32	CREATE DATABASE HACKTHON	1 row(s) affected

Action Output			
#	Time	Action	Message
2	14:43:08	USE HACKTHON	0 row(s) affected
3	14:47:27	CREATE TABLE TRIANGLES (Triangle INT primary KEY, Side_A int , Side_B int, Side_C int)	0 row(s) affected

INSERT INTO TRIANGLES (Triangle,Side_A,Side_B,Side_C)
values (1,20,20,23),
(2,20,20,20),
(3,20,21,22),
(4,13,14,30);

```
INSERT INTO TRIANGLES (Triangle,Side_A,Side_B,Side_C)
values (1,20,20,23),
(2,20,20,20),
(3,20,21,22),
(4,13,14,30);
```

SELECT * FROM TRIANGLES;

The screenshot shows the MySQL Workbench interface with a result grid. The grid has four columns: Triangle, Side_A, Side_B, and Side_C. The data is as follows:

	Triangle	Side_A	Side_B	Side_C
▶	1	20	20	23
	2	20	20	20
	3	20	21	22
	4	13	14	30
•	NULL	NULL	NULL	NULL

2. Write a query to create the Employees table

CREATE TABLE
EMPLOYEES (employee_id
 int primary key,
 name varchar(50),
 month int ,
 salary int);

The screenshot shows the MySQL Workbench interface with a code editor containing a CREATE TABLE statement:

```
CREATE TABLE EMPLOYEES (employee_id int primary key,
name varchar(50),
month int ,
salary int
);
```

```
INSERT INTO EMPLOYEES (
employee_id, name, month , salary)
values (12228, 'Rahul', 15, 10000),
(33645, 'Amit', 1, 15000),
(45692, 'Aditi', 17, 18000),
(56188, 'Pavan', 11, 21000 );
```

```
INSERT INTO EMPLOYEES (employee_id, name, month , salary)
values (12228, 'Rahul', 15, 10000),
(33645, 'Amit', 1, 15000),
(45692, 'Aditi', 17, 18000),
(56188, 'Pavan', 11, 21000 );
```

```
SELECT * FROM EMPLOYEES;
```

	employee_id	name	month	salary
▶	12228	Rahul	15	10000
	33645	Amit	1	15000
	45692	Aditi	17	18000
	56188	Pawan	11	21000
•	NULL	NULL	NULL	NULL

a. Write a query to obtain the sum of side_A of all triangles.

SELECT sum(Side_A) FROM TRIANGLES;

Result Grid		Filter Rows:	<input type="text"/>	Export:
▶	sum(Side_A)			
	73			

b. Write a query to obtain an equilateral triangle from the table.

SELECT Triangle FROM TRIANGLES where (Side_A = Side_B and Side_B = Side_C and Side_C= Side_A);

The screenshot shows a software interface for viewing database results. At the top, there are tabs for "Result Grid" (which is selected), "Edit", and other options. Below the tabs is a toolbar with icons for refresh, filter, and edit. A search bar labeled "Filter Rows:" is present. The main area displays a grid with one row. The first column is empty, and the second column is labeled "Triangle". The data row contains three cells: a right-pointing arrow icon, the number "2", and the word "NULL".

	Triangle
▶	2
✖	NULL

- c. Write a query to obtain an isosceles triangle from the table

SELECT Triangle FROM TRIANGLES where (Side_A = Side_B or Side_B = Side_C or Side_C= Side_A);

The screenshot shows a software interface for viewing database results. At the top, there are tabs for "Result Grid" (selected), "Edit", and other options. Below the tabs is a toolbar with icons for refresh, filter, and edit. A search bar labeled "Filter Rows:" is present. The main area displays a grid with two rows. The first column is empty, and the second column is labeled "Triangle". The first data row contains three cells: a right-pointing arrow icon, the number "1", and the word "NULL". The second data row contains three cells: a right-pointing arrow icon, the number "2", and the word "NULL".

	Triangle
▶	1
▶	2
✖	NULL

d. Find the no. of triangles in the table.

SELECT count(triangle) FROM TRIANGLES;

Result Grid		Filter Rows:	Export:
count(triangle)			
▶	4		

e. Find the length of side_B of Triangle 3.

**SELECT Side_B FROM TRIANGLES where
Triangle=3;**

Result Grid		Filter Rows:	Export:
Side_B			
▶	21		

2.a Count the total no. of employees.

```
SELECT count(employee_id) FROM  
EMPLOYEES;
```

Result Grid		Filter Rows:	Export
count(employee_id)			
▶	4		

2. b Find the salary of Rahul.

```
SELECT salary FROM EMPLOYEES  
where name= 'Rahul';
```

Result Grid		Filter Rows:	Export
salary			
▶	10000		

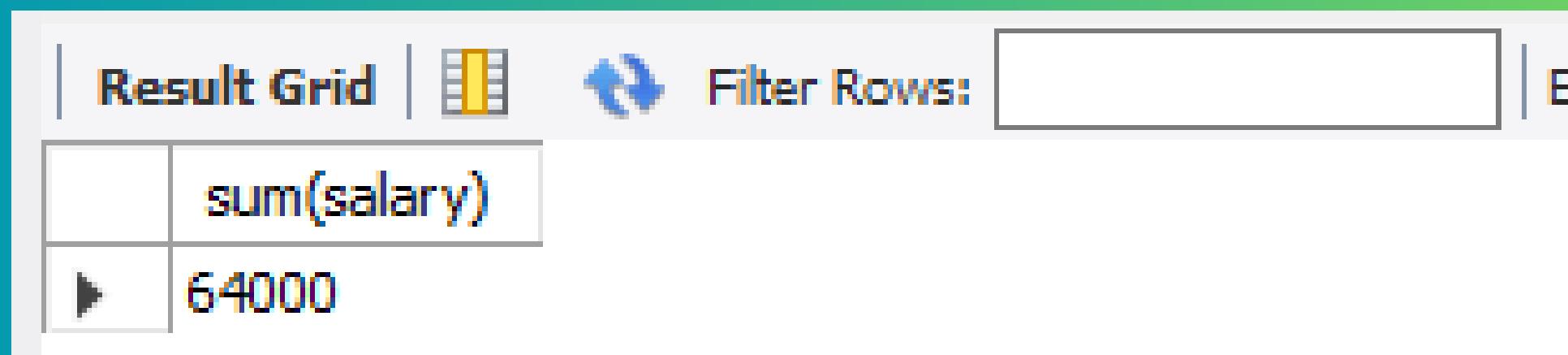
2.c Set Amit's months to 12.

```
update EMPLOYEES set month =12  
where name='Amit';
```

	employee_id	name	month	salary
▶	12228	Rahul	15	10000
	33645	Amit	12	15000
	45692	Aditi	17	18000
	56188	Pawan	11	21000
✳	HULL	HULL	HULL	HULL

2.d Find the sum of salaries of all employees.

```
SELECT sum(salary) FROM EMPLOYEES;
```



The screenshot shows a MySQL Workbench interface with a result grid. The grid has one row and two columns. The first column is empty, and the second column is labeled 'sum(salary)'. The value '64000' is displayed in the cell under the 'sum(salary)' header.

	sum(salary)
▶	64000

2.e Find no. of employees whose name starts with ‘A’.

```
SELECT count(employee_id) as Number FROM  
EMPLOYEES where name like 'A%';
```

	Number
▶	2

**Prepared by -
Vikas Kumar Maurya**

Thank You