

Data Base Management Systems Record

BY 20VV1A1263
2-1 B.Tech IT (R20)

EXPERIMENT NO: 1

AIM: Queries to facilitate acquaintance of Built-In Functions, String Functions, Numeric Functions, Date Functions and Conversion Functions.

DESCRIPTION:

String functions are used to perform an operation on input string and return an output string.

Ex:concat(),lower(),upper(),length(),position(),replace(),reverse(),substr()

Numeric functions are used to perform an operation on Numbers and return Numbers.

Ex : abs(), ceil(), div(), floor(), mod(), round(), sqrt(),truncate()

Date functions are used to display date formats

Ex : now(), curdate(), curtime(), date(), date_add(), date_sub(),format().

Conversation functions are used to convert data from one type to another type.

Ex : varchar2 or char TO number, varchar2 or char TO Number ,Date to varchar2, number to varchar2

QUERY:

SQL STRING FUNCTIONS:

1. ascii()

```
select ascii('yaswanth');
```

2. length()

```
select length('yaswanth');
```

3. upper()

```
select upper('Yaswanth');
```

4. substring()

```
select substring('learning sql',2,50);
```

5. strcmp()

```
select strcmp('yaswanth','battu');
```

6. rtrim()

```
select rtrim('venkat ');
```

7. right()

```
select right('learning sql is cool',4);
```

8. rpad()

```
select rpad('yaswanth',20,'battu');
```

9. reverse()

```
select reverse('string');
```

10. replace()

```
select replace('venkata yaswanth','venkata','battu');
```

11. repeat()

```
select repeat('amazing!',3);
```

12. position()

```
select position('n' in 'aeroplane');
```

13. mid()

```
select mid('harry potter',3,6);
```

14. ltrim()

```
select ltrim(' harekrishna!');
```

15. lpad()

```
select lpad('yaswanth',15,'battu');
```

16. lower()

```
select lower('RAMESH');
```

17. left()

```
select left('harry',3);
```

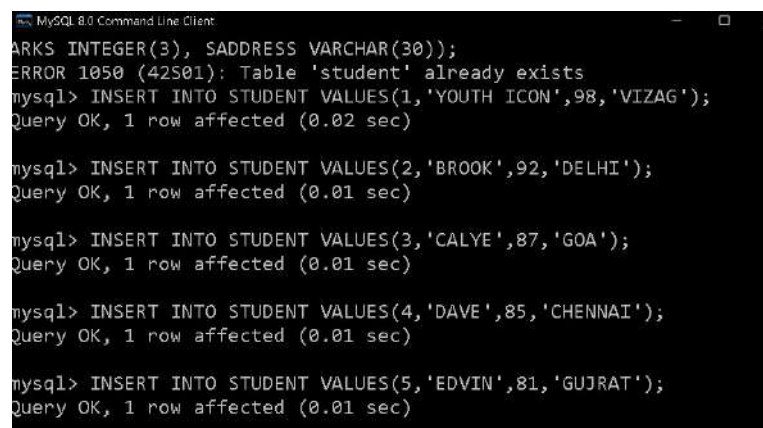
18. concat()

```
select concat('harry','potter');
```

19. insert()

```
select insert('prince of persia',1,6,'king');
```

OUTPUT:



```
MySQL 8.0 Command Line Client
ERROR 1050 (42S01): Table 'student' already exists
mysql> INSERT INTO STUDENT VALUES(1,'YOUTH ICON',98,'VIZAG');
Query OK, 1 row affected (0.02 sec)

mysql> INSERT INTO STUDENT VALUES(2,'BROOK',92,'DELHI');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO STUDENT VALUES(3,'CALYE',87,'GOA');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO STUDENT VALUES(4,'DAVE',85,'CHENNAI');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO STUDENT VALUES(5,'EDVIN',81,'GUJRAT');
```

```

MySQL 8.0 Command Line Client
mysql> /* SQL STRING FUNCTIONS */
mysql> SELECT ASCII('YASWANTH');
+-----+
| ASCII('YASWANTH') |
+-----+
| 89 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LENGTH('YASWANTH');
+-----+
| LENGTH('YASWANTH') |
+-----+
| 8 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT UPPER('Yaswanth');
+-----+
| UPPER('Yaswanth') |
+-----+
| YASWANTH |
+-----+
1 row in set (0.01 sec)

```

```

mysql> SELECT SUBSTRING('LEARNING SQL',2,50);
+-----+
| SUBSTRING('LEARNING SQL',2,50) |
+-----+
| EARNING SQL |
+-----+
1 row in set (0.00 sec)

mysql> SELECT STRCMP('YASWANTH','BATTU');
+-----+
| STRCMP('YASWANTH','BATTU') |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT RTIM('VENKAT ');
ERROR 1046 (3D000): No database selected
mysql> SELECT RTRIM('VENKAT ');
+-----+
| RTRIM('VENKAT ') |
+-----+
| VENKAT |
+-----+
1 row in set (0.01 sec)

```

```

mysql> SELECT RIGHT('LEARNING SQL IS COOL',4);
+-----+
| RIGHT('LEARNING SQL IS COOL',4) |
+-----+
| COOL |
+-----+
1 row in set (0.00 sec)

mysql> SELECT RPAD('YASWANTH',20,'BATTU');
+-----+
| RPAD('YASWANTH',20,'BATTU') |
+-----+
| YASWANTHBATTUBATTUBA |
+-----+
1 row in set (0.01 sec)

mysql> SELECT REVERSE('STRING');
+-----+
| REVERSE('STRING') |
+-----+
| GNIRTS |
+-----+
1 row in set (0.01 sec)

```

```

mysql> SELECT REPLACE("VENKATA YASWANTH","VENKATA","BATTU");
+-----+
| REPLACE("VENKATA YASWANTH","VENKATA","BATTU") |
+-----+
| BATTU YASWANTH |
+-----+
1 row in set (0.00 sec)

mysql> SELECT REPEAT('AMAZING!',3);
+-----+
| REPEAT('AMAZING!',3) |
+-----+
| AMAZING!AMAZING!AMAZING! |
+-----+
1 row in set (0.00 sec)

mysql> SELECT POSITION('N' IN 'AEROPLANE');
+-----+
| POSITION('N' IN 'AEROPLANE') |
+-----+
| 8 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT MID('HARRY POTTER',3,6);
+-----+
| MID('HARRY POTTER',3,6) |
+-----+
| RRY PO |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LTRIM(' HARE KRISHNA!');
+-----+
| LTRIM(' HARE KRISHNA!') |
+-----+
| HARE KRISHNA! |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LPAD('YASWANTH',15,'BATTU');
+-----+
| LPAD('YASWANTH',15,'BATTU') |
+-----+
| BATTUBAYASWANTH |
+-----+
1 row in set (0.00 sec)

```

```

mysql> SELECT LOWER('RAMESH');
+-----+
| LOWER('RAMESH') |
+-----+
| ramesh |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LEFT('HARRY',3);
+-----+
| LEFT('HARRY',3) |
+-----+
| HAR |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CONCAT('HARRY ','POTTER');
+-----+
| CONCAT('HARRY ','POTTER') |
+-----+
| HARRY POTTER |
+-----+
1 row in set (0.00 sec)

mysql> SELECT INSERT('PRINCE OF PERSIA',1,6,'KING');
+-----+
| INSERT('PRINCE OF PERSIA',1,6,'KING') |
+-----+
| KING OF PERSIA |
+-----+
1 row in set (0.00 sec)

```

SQL NUMERIC FUNCTIONS:

1. abs()

```
select abs(-1908);
```

2. acos()

```
select acos(90);
```

3. asin()

```
select asin(90);
```

4. sin()

```
select sin(90);
```

5. sum()

```
select sum(smarks) from student;
```

6. sqrt()

```
select sqrt(36);
```

7. sign()

```
select sign(-9098);
```

8. round()

```
select round(89.90);
```

9. rand()

```
select rand();
```

10. radians()

```
select radian(90);
```

11. power()

```
select power(2,4);
```

12. mod()

```
select mod(10,5);
```

13. log()

```
select log(2);
```

14. log10()

```
select log10(100);
```

15. least()

```
select least(1,0,6,-100,98,7);
```

16. floor()
select floor(2.6);

17. ceil()
select ceil(2.6);

18. degrees()
select degrees(1.57);

OUTPUT:

```
MySQL 8.0 Command Line Client
mysql> /* SQL NUMERIC FUNCTIONS */
mysql> SELECT ABS(-1908);
+-----+
| ABS(-1908) |
+-----+
|         1908 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT ACOS(90);
+-----+
| ACOS(90) |
+-----+
|      NULL |
+-----+
1 row in set (0.00 sec)

mysql> SELECT ASIN(90);
+-----+
| ASIN(90) |
+-----+
|      NULL |
+-----+
1 row in set (0.00 sec)

mysql> SELECT SIN(90);
+-----+
| SIN(90) |
+-----+
| 0.8939966636005579 |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> SELECT SUM(SMARKS) FROM STUDENT
+-----+
| SUM(SMARKS) |
+-----+
|          443 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT SQRT(36);
+-----+
| SQRT(36) |
+-----+
|         6 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT SIGN(-9098);
+-----+
| SIGN(-9098) |
+-----+
|          -1 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT ROUND(89.90);
+-----+
| ROUND(89.90) |
+-----+
|           90 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT RAND();
+-----+
| RAND() |
+-----+
| 0.4367624404251226 |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> SELECT RADIANS(90);
+-----+
| RADIANS(90) |
+-----+
| 1.5707963267948966 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT POWER(2,4);
+-----+
| POWER(2,4) |
+-----+
|         16 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT MOD(10,5);
+-----+
| MOD(10,5) |
+-----+
|         0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LOG(2);
+-----+
| LOG(2) |
+-----+
| 0.6931471805599453 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LOG10(100);
+-----+
| LOG10(100) |
+-----+
|         2 |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
ERROR 1582 (42000): Incorrect param
mysql> SELECT LEAST(1,0,6,-100,98,7);
+-----+
| LEAST(1,0,6,-100,98,7) |
+-----+
|          -100 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT FLOOR(2.6);
+-----+
| FLOOR(2.6) |
+-----+
|         2 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CEIL(2.6);
+-----+
| CEIL(2.6) |
+-----+
|         3 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT DEGREES(1.57);
+-----+
| DEGREES(1.57) |
+-----+
| 89.95437383553924 |
+-----+
1 row in set (0.00 sec)
```

SQL DATE FUNCTIONS:

1. adddate()

```
select adddate('2017-06-15',10);
```

2. addtime()

```
select addtime('2018-06-15 09:34:21','2');
```

3. curdate()

```
select curdate();
```

4. curtime()

```
select curtime();
```

5. date()

```
select date('2018-06-15 09:34:21');
```

6. datediff()

```
select datediff('2018-06-12','2018-08-12');
```

7. dformat()

```
select dformat'2017-06-15','%y');
```

8. dayname()

```
select dayname('2017-06-15');
```

9. dayofmonth()

```
select dayofmonth('2017-06-15');
```

10. dayofweek();

```
select dayofweek('2017-06-15');
```

11. dayofyear()

```
select dayofyear('2-17-06-15');
```

12. extract()

```
select extract(month from '2017-06-16');
```

13. from_days()

```
select fromdays(685467);
```

14. hour()

```
select hour('2017-06-20 09:34:00');
```

15. dayofmonth()

```
select dayofmonth('2017-06-15');
```

16. localtime()
select localtime();

17. localtimestamp()
select localtimestamp();

18. maketime()
select maketime(11,35,4);

19. timestamp()
select timestamp('2017-07-23','13:10:11');

20. time_format()
select time-format('19:30:10','%h %i %s');
OUTPUT:

```
MySQL 8.0 Command Line Client
mysql> /* SQL DATE FUNCTIONS */
mysql> SELECT ADDDATE('2017-06-15',10);
+-----+
| ADDDATE('2017-06-15',10) |
+-----+
| 2017-06-25               |
+-----+
1 row in set (0.00 sec)

mysql> SELECT ADDTIME('2018-06-15 09:34:21','2');
+-----+
| ADDTIME('2018-06-15 09:34:21','2') |
+-----+
| 2018-06-15 09:34:23               |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CURDATE();
+-----+
| CURDATE() |
+-----+
| 2022-01-13 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CURTIME();
+-----+
| CURTIME() |
+-----+
| 16:12:35  |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> SELECT DATE('2018-06-15 09:34:21');
+-----+
| DATE('2018-06-15 09:34:21') |
+-----+
| 2018-06-15                   |
+-----+
1 row in set (0.01 sec)

mysql> SELECT DATDIFF('2018-06-12','2018-08-12');
ERROR 1305 (42000): FUNCTION dbms_record_work.DATD
mysql> SELECT DATEDIFF('2018-06-12','2018-08-12');
+-----+
| DATEDIFF('2018-06-12','2018-08-12') |
+-----+
| -61 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT DATE_FORMAT('2017-06-15','%y');
+-----+
| DATE_FORMAT('2017-06-15','%y') |
+-----+
| 17 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT DAYNAME('2017-06-15','%y');
ERROR 1582 (42000): Incorrect parameter count in t
DAYNAME'
mysql> SELECT DAYNAME('2017-06-15');
+-----+
| DAYNAME('2017-06-15') |
+-----+
| Thursday               |
+-----+
1 row in set (0.01 sec)
```



```
mysql> SELECT DAYMONTH("2017-06-15");
ERROR 1305 (42000): FUNCTION dbms_record_work.DAY
mysql> SELECT DAYOFMONTH("2017-06-15");
+-----+
| DAYOFMONTH("2017-06-15") |
+-----+
| 15 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT DAYOFWEEK("2017-06-15");
+-----+
| DAYOFWEEK("2017-06-15") |
+-----+
| 5 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT DAYOFYEAR("2017-06-15");
+-----+
| DAYOFYEAR("2017-06-15") |
+-----+
| 166 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT EXTRACT(MONTH FROM '2017-06-16');
+-----+
| EXTRACT(MONTH FROM '2017-06-16') |
+-----+
| 6 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT MAKETIME(11,35,4);
+-----+
| MAKETIME(11,35,4) |
+-----+
| 11:35:04 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT TIMESTAMP("2017-07-23","13:10:11");
+-----+
| TIMESTAMP("2017-07-23","13:10:11") |
+-----+
| 2017-07-23 13:10:11 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT TIME_FORMAT("19:30:10","%H %I %S");
+-----+
| TIME_FORMAT("19:30:10","%H %I %S") |
+-----+
| 19 07 10 |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> SELECT FROM_DAYS(685467);
+-----+
| FROM_DAYS(685467) |
+-----+
| 1876-09-29 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT HOUR('2017-06-20 09:34:00');
+-----+
| HOUR('2017-06-20 09:34:00') |
+-----+
| 9 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT DAYOFMONTH('2017-06-15');
+-----+
| DAYOFMONTH('2017-06-15') |
+-----+
| 15 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LOCALTIME();
+-----+
| LOCALTIME() |
+-----+
| 2022-01-13 17:59:50 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT LOCALTIMESTAMP();
+-----+
| LOCALTIMESTAMP() |
+-----+
| 2022-01-13 18:00:03 |
+-----+
1 row in set (0.00 sec)
```

SQL CONVERSATION FUNCTIONS:

1. CONVERT()

select covert(14.85,binary);

select convert(18.85,signed);

2. cast()

select cast('2017-08-25' as datetime);

```
mysql> SELECT CONVERT(14.85,BINARY);
+-----+
| CONVERT(14.85,BINARY) |
+-----+
| 0x31342E3835          |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CONVERT(14.85,CHAR);
+-----+
| CONVERT(14.85,CHAR) |
+-----+
| 14.85               |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CONVERT(14.85,SIGNED);
+-----+
| CONVERT(14.85,SIGNED) |
+-----+
| 15                    |
+-----+
1 row in set (0.00 sec)

mysql> SELECT CONVERT(14.85,UNSIGNED);
+-----+
| CONVERT(14.85,UNSIGNED) |
+-----+
| 15                      |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT CAST('2017-08-25' AS datetime);
+-----+
| CAST('2017-08-25' AS datetime) |
+-----+
| 2017-08-25 00:00:00            |
+-----+
1 row in set (0.00 sec)
```

EXPERIMENT NO: 2

AIM: Queries using operators in SQL

DESCRIPTION:

An Operator perform an operation on given operands. And in SQL different types of operators are there namely -

1. Arithmetic
2. Comparison
3. Logical,
4. Bitwise,
5. Compound

Arithmetic

these are basic operators (+, -, *, /, %)

Comparison

these are used to compare the values and return values. (=, >, <, >=, <=, <>)

Logical

these are used to perform logical calculation and return values (ALL, AND, ANY, BETWEEN, EXISTS, IN, LIKE, NOT, OR, SOME)

Compound

+=, -=, *=, /=, &=, ^=, *= these are compound operators

QUERY:

SQL ARITHMETIC OPERATORS:

```
select 40+20;  
select 40-20;  
select 40*20;  
select 40/20;  
select 40%20;
```

SQL BITWISE OPERATORS:

```
select 40&20;  
select 40 | 20;  
select 40 ^ 20;
```

SQL COMPARISON OPERATORS:

```
select 40 = 20;
select 40 > 20;
select 40 < 20;
select 40 >= 20;
select 40 <= 20;
select 40 <> 20;
```

SQL LOGICAL OPERATORS:

```
select 40 and 0;
select 40 or 0;
select not 0;
select * from student where smaksr between 80 and 90;
select * from student where sname like '%M__';
```

OUTPUT:

```
MySQL 8.0 Command Line Client
mysql> /* SQL ARITHMETIC OPERATORS */
mysql> SELECT 40+20;
+-----+
| 40+20 |
+-----+
|    60 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40-20;
+-----+
| 40-20 |
+-----+
|    20 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40*20;
+-----+
| 40*20 |
+-----+
|   800 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40/20;
+-----+
| 40/20 |
+-----+
| 2.0000 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40%20;
+-----+
| 40%20 |
+-----+
|     0 |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /* SQL BITWISE OPERATORS */
mysql> SELECT 40&20;
+-----+
| 40&20 |
+-----+
|     0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 | 20;
+-----+
| 40 | 20 |
+-----+
|   60 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40^20;
+-----+
| 40^20 |
+-----+
|     60 |
+-----+
1 row in set (0.00 sec)
```

```

MySQL 8.0 Command Line Client
mysql> /* SQL COMPARISION OPERATORS */
mysql> SELECT 40 = 20;
+-----+
| 40 = 20 |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 > 20;
+-----+
| 40 > 20 |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 < 20;
+-----+
| 40 < 20 |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 <= 20;
+-----+
| 40 <= 20 |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 >= 20;
+-----+
| 40 >= 20 |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 <> 20;
+-----+
| 40 <> 20 |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)

```

```

MySQL 8.0 Command Line Client
mysql> SELECT 40 AND 0;
+-----+
| 40 AND 0 |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT 40 OR 0;
+-----+
| 40 OR 0 |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT NOT 0;
+-----+
| NOT 0 |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)

mysql> USE DBMS_RECORD_WORK;
Database changed
mysql>
mysql> SELECT * FROM STUDENT WHERE SMARKS BETWEEN 80 AND 90;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | SADDRESS |
+-----+-----+-----+-----+
| 3   | CALYE | 87    | GOA      |
| 4   | DAVE  | 85    | CHENNAI  |
| 5   | EDVIN | 81    | GUJRAT   |
| 10  | LAKSHMAN | 90   | KAKINADA |
| 9   | LAKSHMAN | 89   | GODAVARI |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> SELECT * FROM STUDENT WHERE SNAME LIKE '%M__';
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | SADDRESS |
+-----+-----+-----+-----+
| 10  | LAKSHMAN | 90   | KAKINADA |
| 9   | LAKSHMAN | 89   | GODAVARI |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

```

EXPERIMENT NO: 3

AIM: Queries to Retrieve and Change Data: Select, Insert, Delete, and Update.

DESCRIPTION:

SELECT :

This select key word used to retrieve selected data from table.

- All selection from table.
- Column selection from table.
- Row selection from table.

INSERT :

This insert key used to insert data into table.

UPDATE :

This Update key used to update data in the table.

DELETE :

this delete key is used to delete data from table.

- Delete a row
- Delete all columns

QUERY:

SQL SELECT COMMAND:

```
select sname from student;  
select * from student;
```

SQL INSERT COMMAND:

```
insert into student values(14,'arjun',100,'gudivada');  
insert into student values(15,'siva',100,'rajamundry');
```

SQL DELETE COMMAND:

```
delete from student where sname='lakshman';
```

OUTPUT:

```
MySQL 8.0 Command Line Client
mysql> /* SQL SELECT COMMAND */
mysql> SELECT SNAME FROM STUDENTS;
ERROR 1146 (42S02): Table 'dbms_recond_work.student'
mysql> SELECT SNAME FROM STUDENT;
+-----+
| SNAME |
+-----+
| YOUTH ICON |
| BROOK |
| CALYE |
| DAVE |
| EDVIN |
| LAKSHMAN |
| SUBASH |
| LAKSHMAN |
| SUBASH |
+-----+
0 rows in set (0.00 sec)

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 85 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 10 | LAKSHMAN | 90 | KAKINADA |
| 12 | SUBASH | 99 | ELURU |
| 9 | LAKSHMAN | 89 | GODAVARI |
| 12 | SUBASH | 99 | RAJIL |
+----+-----+-----+-----+
0 rows in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /* SQL INSERT COMMAND */
mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 85 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 10 | LAKSHMAN | 90 | KAKINADA |
| 12 | SUBASH | 99 | ELURU |
| 9 | LAKSHMAN | 89 | GODAVARI |
| 12 | SUBASH | 99 | RAJIL |
+----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> INSERT INTO STUDENT VALUES(14,'ARJUN',100,'GUDIVADA');
ERROR 1064 (42000): You have an error in your SQL syntax; check
the manual that corresponds to your MySQL server version for the
right syntax to use near '100,'GUDIVADA'
mysql> INSERT INTO STUDENT VALUES(14,'ARJUN',100,'GUDIVADA');
ERROR 1064 (42000): You have an error in your SQL syntax; check
the manual that corresponds to your MySQL server version for the
right syntax to use near '100,'GUDIVADA'
mysql> INSERT INTO STUDENT VALUES(14,'ARJUN',100,'GUDIVADA');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO STUDENT VALUES(15,'SIVA',100,'RAJAMUNDY');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 85 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 10 | LAKSHMAN | 90 | KAKINADA |
| 12 | SUBASH | 99 | ELURU |
| 9 | LAKSHMAN | 89 | GODAVARI |
| 12 | SUBASH | 99 | RAJIL |
| 14 | ARJUN | 100 | GUDIVADA |
| 15 | SIVA | 100 | RAJAMUNDY |
+----+-----+-----+-----+
11 rows in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /*SQL DELETE COMMAND */
mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 85 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 10 | LAKSHMAN | 90 | KAKINADA |
| 12 | SUBASH | 99 | ELURU |
| 9 | LAKSHMAN | 89 | GODAVARI |
| 12 | SUBASH | 99 | RAJIL |
| 14 | ARJUN | 100 | GUDIVADA |
| 15 | SIVA | 100 | RAJAMUNDY |
+----+-----+-----+-----+
11 rows in set (0.00 sec)

mysql> DELETE FROM STUDENT WHERE SNAME='LAKSHMAN';
Query OK, 2 rows affected (0.01 sec)

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 85 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 12 | SUBASH | 99 | ELURU |
| 12 | SUBASH | 99 | RAJIL |
| 14 | ARJUN | 100 | GUDIVADA |
| 15 | SIVA | 100 | RAJAMUNDY |
+----+-----+-----+-----+
9 rows in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /*SQL DELETE COMMAND */
mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 85 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 12 | SUBASH | 99 | ELURU |
| 12 | SUBASH | 99 | RAJIL |
| 14 | ARJUN | 100 | GUDIVADA |
| 15 | SIVA | 100 | RAJAMUNDY |
+----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> UPDATE STUDENT SET SMARKS=90 WHERE SNAME = 'DAVE';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 90 | CHENNAI |
| 5 | EDVIN | 81 | GUJRAT |
| 12 | SUBASH | 99 | ELURU |
| 12 | SUBASH | 99 | RAJIL |
| 14 | ARJUN | 100 | GUDIVADA |
| 15 | SIVA | 100 | RAJAMUNDY |
+----+-----+-----+-----+
9 rows in set (0.00 sec)
```

EXPERIMENT NO: 4

AIM: Queries using Group By, Order By, and Having Clauses

DESCRIPTION:

ORDER BY

- The ORDER BY keyword is used to sort the result-set in ascending or descending order.
- The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

GROUP BY

- GROUP BY clause is used with the SELECT statement.
- In the query, GROUP BY clause is placed after the WHERE clause.
- In the query, GROUP BY clause is placed before ORDER BY clause if used any.

HAVING

- The HAVING clause was added to SQL because the WHERE keyword cannot be used with aggregate functions.

QUERY:

SQL GROUP BY CLAUSE:

```
select * from student group by smakrs;  
select * from students;
```

SQL GROUP BY CLAUSE:

```
select * from student order by smarks asc;  
select * from student order by sid desc;
```

SQL HAVING CLAUSE:

```
select * from student having smarks >= 90;  
select * from student having smarks < 50;  
select * from student having smarks < 80;  
select * from student having smarks < 90;
```

SQL GROUP BY AND HAVING CLAUSE:

```
select * from student GROUP by smarks having smarks > 90;
```


OUTPUT:

```
MySQL 8.0 Command Line Client
mysql> /* SQL GROUP BY */
mysql> SELECT * FROM STUDENT GROUP BY SMARKS;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 1   | YOUTH ICON | 98 | VIZAG |
| 2   | BROOK | 92 | DELHI |
| 3   | CALYE | 87 | GOA |
| 4   | DAVE | 90 | CHENNAI |
| 5   | EDVIN | 81 | GUJRAT |
| 12  | SUBASH | 99 | ELURU |
| 14  | ARJUN | 100 | GUDIVADA |
+-----+-----+-----+-----+
7 rows in set (0.00 sec)

mysql> SELECT * FROM STUDENT;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 1   | YOUTH ICON | 98 | VIZAG |
| 2   | BROOK | 92 | DELHI |
| 3   | CALYE | 87 | GOA |
| 4   | DAVE | 90 | CHENNAI |
| 5   | EDVIN | 81 | GUJRAT |
| 12  | SUBASH | 99 | ELURU |
| 12  | SUBASH | 99 | RAJIL |
| 14  | ARJUN | 100 | GUDIVADA |
| 15  | SIVA | 100 | RAJAMUNDRY |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /* SQL HAVING CLAUSE */
mysql> SELECT 8 FROM STUDENT HAVING SMARKS >= 90;
ERROR 1054 (42S22): Unknown column 'SMARKS' in 'having clause'
mysql> SELECT * FROM STUDENT HAVING SMARKS >= 90;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 1   | YOUTH ICON | 98 | VIZAG |
| 2   | BROOK | 92 | DELHI |
| 4   | DAVE | 90 | CHENNAI |
| 12  | SUBASH | 99 | ELURU |
| 12  | SUBASH | 99 | RAJIL |
| 14  | ARJUN | 100 | GUDIVADA |
| 15  | SIVA | 100 | RAJAMUNDRY |
+-----+-----+-----+-----+
7 rows in set (0.00 sec)

mysql> SELECT * FROM STUDENT HAVING SMARKS < 50;
Empty set (0.00 sec)

mysql> SELECT * FROM STUDENT HAVING SMARKS < 80;
Empty set (0.00 sec)

mysql> SELECT * FROM STUDENT HAVING SMARKS < 90;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 3   | CALYE | 87 | GOA |
| 5   | EDVIN | 81 | GUJRAT |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /* SQL ORDER BY */
mysql> SELECT * FROM STUDENT ORDER BY SMARKS ASC;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 5   | EDVIN | 81 | GUJRAT |
| 3   | CALYE | 87 | GOA |
| 4   | DAVE | 90 | CHENNAI |
| 2   | BROOK | 92 | DELHI |
| 1   | YOUTH ICON | 98 | VIZAG |
| 12  | SUBASH | 99 | ELURU |
| 12  | SUBASH | 99 | RAJIL |
| 14  | ARJUN | 100 | GUDIVADA |
| 15  | SIVA | 100 | RAJAMUNDRY |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> SELECT * FROM STUDENT ORDER BY SID DSC;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
comes with this MySQL server version for the right syntax to use near '*'
mysql> SELECT * FROM STUDENT ORDER BY SID DESC;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 5   | EDVIN | 81 | GUJRAT |
| 4   | DAVE | 90 | CHENNAI |
| 3   | CALYE | 87 | GOA |
| 2   | BROOK | 92 | DELHI |
| 15  | SIVA | 100 | RAJAMUNDRY |
| 14  | ARJUN | 100 | GUDIVADA |
| 12  | SUBASH | 99 | ELURU |
| 12  | SUBASH | 99 | RAJIL |
| 1   | YOUTH ICON | 98 | VIZAG |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

```
mysql> /* SQL GROUP BY AND HAVING CLAUSE */
mysql> SELECT * FROM STUDENT GROUP BY SMARKS HAVING SMARKS > 90;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 1   | YOUTH ICON | 98 | VIZAG |
| 2   | BROOK | 92 | DELHI |
| 12  | SUBASH | 99 | ELURU |
| 14  | ARJUN | 100 | GUDIVADA |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> /* SQL GROUP BY AND HAVING, ORDER BY CLAUSE */
mysql> SELECT * FROM STUDENT GROUP BY SMARKS HAVING SMARKS > 90 ORDER BY SID DESC;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 2   | BROOK | 92 | DELHI |
| 14  | ARJUN | 100 | GUDIVADA |
| 12  | SUBASH | 99 | ELURU |
| 1   | YOUTH ICON | 98 | VIZAG |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

EXPERIMENT NO: 5

AIM: Queries on Controlling Data: Commit, Rollback, and Save point

DESCRIPTION:

COMMIT:

All clauses after the COMMIT keyword are optional. If you specify only COMMIT, then the default is COMMIT WORK WRITE IMMEDIATE WAIT.

ROLLBACK:

ROLLBACK in SQL is a transactional control language which is used to undo the transactions that have not been saved in database. The command is only be used to undo changes since the last COMMIT.

SAVEPOINT:

A SAVEPOINT is a point in a transaction in which you can roll the transaction back to a certain point without rolling back the entire transaction.

QUERY:

```
start transaction;
select * from student;
savepoint a;
insert into student value(12,'namo',88,'vzm');
insert into student values(13,'raju',78,'vzm');
savepoint b;
select * from students;
rollback to savepoint a;
rollback to savepoint a;
select * from student;
commit;
```

OUTPUT:

```
mysql> ROLLBACK TO SAVEPOINT A;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM STUDENT;
+-----+-----+-----+-----+
| SID | SNAME   | SMARKS | SADDRESS |
+-----+-----+-----+-----+
| 1   | YOUTH ICON | 98    | VIZAG   |
| 2   | BROOK     | 92    | DELHI   |
| 3   | CALYE     | 87    | GOA     |
| 4   | DAVE      | 90    | CHENNAI |
| 10  | HARISH    | 80    | VZM     |
| 11  | HARI      | 89    | RJY     |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> COMMIT;
Query OK, 0 rows affected (0.01 sec)
```

```
Select MySQL 8.0 Command Line Client
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | SADDRESS |
+----+-----+-----+-----+
| 1   | YOUTH ICON | 98 | VIZAG |
| 2   | BROOK | 92 | DELHI |
| 3   | CALYE | 87 | GOA |
| 4   | DAVE | 90 | CHENNAI |
| 10  | HARISH | 80 | VZM |
| 11  | HARI | 89 | RJY |
+----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> SAVEPOINT A;
Query OK, 0 rows affected (0.00 sec)

mysql> INSERT INTO STUDENT VALUES(12,'NAMO',88,'VZM');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO STUDENT VALUES(13,'RAJU',78,'VZM');
Query OK, 1 row affected (0.00 sec)

mysql> SAVEPOINT B;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+
| SID | SNAME | SMARKS | SADDRESS |
+----+-----+-----+-----+
| 1   | YOUTH ICON | 98 | VIZAG |
| 2   | BROOK | 92 | DELHI |
| 3   | CALYE | 87 | GOA |
| 4   | DAVE | 90 | CHENNAI |
| 10  | HARISH | 80 | VZM |
| 11  | HARI | 89 | RJY |
| 12  | NAMO | 88 | VZM |
| 13  | RAJU | 78 | VZM |
+----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> ROLLBACK TO SAVEPOINT A;
Query OK, 0 rows affected (0.00 sec)
```

EXPERIMENT NO: 7

AIM: Queries for Creating, Dropping, and Altering Tables, Views, and Constraints

DESCRIPTION:

CREATING:

The CREATE TABLE command creates a new table in the database.

DROPPING:

The DROP TABLE statement is used to drop an existing table in a database.

ALTERING:

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table. The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

VIEW:

In SQL, a view is a virtual table based on the result-set of an SQL statement. A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database. You can add SQL statements and functions to a view and present the data as if the data were coming from one single table. A view is created with the CREATE VIEW statement.

CONSTRAINTS:

Constraints can be specified when the table is created with the CREATE TABLE statement, or after the table is created with the ALTER TABLE statement.

QUERY:

SQL CREATING TABLES:

```
create table employee(eid integer, ename varchar(15), ecity varchar(10));  
desc employee;
```

SQL DROP TABLES:

```
drop employee;  
desc employee;  
select * from employee;
```

SQL ALTERING TABLES:

```
select * from student;
```

```
alter table student drop sid;
```

```
select * from student;
```

```
alter table student add sid integer(5) null;
```

```
select * from student;
```

```
select * from student;
```

```
alter table student rename id to sid;
```

```
alter table student rename column id to sid;
```

```
select * from student;
```

```
desc student;
```

```
alter table student modify column sid varchar(5);
```

```
desc student;
```

SQL VIEWS:

```
create view first as select snmae,smarks from student;
```

```
select * from student;
```

```
create view second as select sname,saddress from student;
```

```
select * from student;
```

SQL CONSTRAINTS:

```
alter table student modify column sid integer(5);
```

```
select * from student;
```

```
alter table student add column sid integer(5) not null default 1;
```

```
select * from student;
```

```
alter table student modify column sname varchar(20) primary key;
```

```
alter table student modify column smarks integer(5) not null;
```

```
desc student;
```

```
insert into student values('dave',90,'eng',2);
```

```
insert into student values('ram',null,'eng',3);
```

OUTPUT:

```
MySQL 8.0 Command Line Client
mysql> /* SQL CREATING TABLES */
mysql> CREATE TABLE EMPLOYEE ( EID INTEGER,ENAME VARCHAR(15),ECITY VARCHAR(10),SALARY INT);
Query OK, 0 rows affected, 1 warning (0.08 sec)

mysql> SHOW TABLE EMPLOYEE;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
  corresponds to your MySQL server version for the right syntax to use near 'EMPLOYEE' at line 1
mysql> DESC EMPLOYEE;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| EID    | int  | YES  |     | NULL    |       |
| ENAME  | varchar(15) | YES  |     | NULL    |       |
| ECITY  | varchar(10) | YES  |     | NULL    |       |
| SALARY | int  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)

mysql> /* DROP TABLE */
mysql> DROP EMPLOYEE;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
  corresponds to your MySQL server version for the right syntax to use near 'EMPLOYEE' at line 1
mysql> DROP TABLE EMPLOYEE;
Query OK, 0 rows affected (0.03 sec)

mysql> DESC EMPLOYEE;
ERROR 1146 (42S02): Table 'dbms_record_work.employee' doesn't exist
mysql> SELECT * FROM EMPLOYEE;
ERROR 1146 (42S02): Table 'dbms_record_work.employee' doesn't exist
```

```
MySQL 8.0 Command Line Client
mysql> SELECT * FROM STUDENT;
+-----+-----+-----+-----+
| SNAME | SMARKS | ADDRESS | ID |
+-----+-----+-----+-----+
| YOUTH ICON | 98 | VIZAG | NULL |
| BROOK | 92 | DELHI | NULL |
| CALYE | 87 | GOA | NULL |
| DAVE | 90 | CHENNAI | NULL |
| HARISH | 80 | VZM | NULL |
| HARI | 89 | RJY | NULL |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> ALTER TABLE STUDENT RENAME ID TO SID;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
  corresponds to your MySQL server version for the right syntax to use near 'ID' at line 1
mysql> ALTER TABLE STUDENT RENAME COLUMN ID TO SID;
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> SELECT * FROM STUDENT;
+-----+-----+-----+-----+
| SNAME | SMARKS | ADDRESS | SID |
+-----+-----+-----+-----+
| YOUTH ICON | 98 | VIZAG | NULL |
| BROOK | 92 | DELHI | NULL |
| CALYE | 87 | GOA | NULL |
| DAVE | 90 | CHENNAI | NULL |
| HARISH | 80 | VZM | NULL |
| HARI | 89 | RJY | NULL |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> /* ALTERING TABLES */
mysql> SELECT * FROM STUDENT;
+-----+-----+-----+-----+
| SID | SNAME | SMARKS | ADDRESS |
+-----+-----+-----+-----+
| 1 | YOUTH ICON | 98 | VIZAG |
| 2 | BROOK | 92 | DELHI |
| 3 | CALYE | 87 | GOA |
| 4 | DAVE | 90 | CHENNAI |
| 10 | HARISH | 80 | VZM |
| 11 | HARI | 89 | RJY |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> ALTER TABLE STUDENT DROP SID;
Query OK, 0 rows affected (0.15 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> SELECT * FROM STUDENT;
+-----+-----+-----+
| SNAME | SMARKS | ADDRESS |
+-----+-----+-----+
| YOUTH ICON | 98 | VIZAG |
| BROOK | 92 | DELHI |
| CALYE | 87 | GOA |
| DAVE | 90 | CHENNAI |
| HARISH | 80 | VZM |
| HARI | 89 | RJY |
+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> ALTER TABLE STUDENT ADD SID INTEGER(5) NULL;
Query OK, 0 rows affected, 1 warning (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> SELECT * FROM STUDENT;
+-----+-----+-----+-----+
| SNAME | SMARKS | ADDRESS | SID |
+-----+-----+-----+-----+
| YOUTH ICON | 98 | VIZAG | NULL |
| BROOK | 92 | DELHI | NULL |
| CALYE | 87 | GOA | NULL |
| DAVE | 90 | CHENNAI | NULL |
| HARISH | 80 | VZM | NULL |
| HARI | 89 | RJY | NULL |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

```

MySQL 8.0 Command Line Client
mysql> /* ALTER COMMAND TO MODIFY A COLUMN*/
mysql> DESC STUDENT;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SNAME | varchar(20) | YES | | NULL | |
| SMARKS | int | YES | | NULL | |
| SADDRESS | varchar(30) | YES | | NULL | |
| SID | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

mysql> ALTER TABLE STUDENT MODIFY COLUMN SID VARCHAR(5);
Query OK, 6 rows affected (0.07 sec)
Records: 6 Duplicates: 0 Warnings: 0

mysql> DESC STUDENT;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SNAME | varchar(20) | YES | | NULL | |
| SMARKS | int | YES | | NULL | |
| SADDRESS | varchar(30) | YES | | NULL | |
| SID | varchar(5) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

```

```

MySQL 8.0 Command Line Client
mysql> /* SQL VIEWS */
mysql> CREATE VIEW FIRST AS SELECT SNAME,SMARKS FROM STUDENT;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM FIRST;
+-----+-----+
| SNAME | SMARKS |
+-----+-----+
| YOUTH ICON | 98 |
| BROOK | 92 |
| CALYE | 87 |
| DAVE | 90 |
| HARISH | 80 |
| HARI | 89 |
+-----+-----+
6 rows in set (0.00 sec)

mysql> CREATE VIEW SECOND AS SELECT SNAME,SADDRESS FROM STUDENT;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM SECOND;
+-----+-----+
| SNAME | SADDRESS |
+-----+-----+
| YOUTH ICON | VIZAG |
| BROOK | DELHI |
| CALYE | GOA |
| DAVE | CHENNAI |
| HARISH | VVM |
| HARI | RJY |
+-----+-----+
6 rows in set (0.00 sec)

```



```
mysql> ALTER TABLE STUDENT MODIFY COLUMN SNAME VARCHAR(20) PRIMARY KEY;
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE STUDENT MODIFY COLUMN SMARKS INTEGER(5) NOT NULL;
Query OK, 0 rows affected, 1 warning (0.12 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> DESC STUDENT;
```

Field	Type	Null	Key	Default	Extra
SNAME	varchar(20)	NO	PRI	NULL	
SMARKS	int	NO		NULL	
SADDRESS	varchar(30)	YES		NULL	
SID	int	NO		1	

```
4 rows in set (0.01 sec)
```

```
mysql> INSERT INTO STUDENT VALUES('DAVE',90,'ENG',2);
ERROR 1062 (23000): Duplicate entry 'DAVE' for key 'student.PRIMARY'
mysql> INSERT INTO STUDENT VALUES('RAM',NULL,'ENG',3);
ERROR 1048 (23000): Column 'SMARKS' cannot be null
mysql>
```

```
mysql> CREATE INDEX INDEX1 ON STUDENT(SNAME,SMARKS);
Query OK, 0 rows affected, 1 warning (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 1
```

```
mysql> ALTER TABLE STUDENT DROP INDEX INDEX1;
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0
```


EXPERIMENT NO 8:

AIM: Queries on Joins and Correlated Sub-Queries

DESCRIPTION:

JOIN:

- A JOIN clause is used to combine rows from two or more tables, based on a related column between them.

CO-RELATED QUERIES:

- Correlated subqueries are used for row-by-row processing. Each subquery is executed once for every row of the outer query.
- A correlated subquery is evaluated once for each row processed by the parent statement. The parent statement can be a SELECT, UPDATE, or DELETE statement.

SUB-QUERY:-

- A Subquery or Inner query or a Nested query is a query within another SQL query and embedded within the WHERE clause.
- A subquery is used to return data that will be used in the main query as a condition to further restrict the data to be retrieved.
- Subqueries can be used with the SELECT, INSERT, UPDATE, and DELETE statements
along with the operators like =, <, >, >=, <=, IN, BETWEEN, etc

QUERY:

1. Create two tables

```
create table foodorder(orderno intger, cid integer, order date);
insert into foodorder values(1,123,'01-02-2002');
insert into foodorder values(2,456,'15-02-2002');
insert into foodorder values(3,789,'21-02-2002');
insert into foodorder values(4,101,'16-03-2002');
insert into foodorder values(5,121,'17-02-2002');
select * from foodorder;
```

```
create table foodcustomer(cid number(10), cname varchar(10));
insert into foodcustomer values(123,'jhon');
insert into foodcustomer values(456,'david');
insert into foodcustomer values(789,'joe');
insert into foodcustomer values(101,'emma');
insert into foodcustomer values(121,'hill');
select * from foodcustomer;
```

sql queries for joining and performing sub queries:

1. joining tables

```
select * from foodcustomer join foodorder on foodcustomer.cid = foodorder.cid;
```

2. performing sub queries

```
select * from foodcustomer join foodorder on foodcustomer.cid = foodorder.cid where  
cnmae = 'emma';
```

```
select * from foodcustomer join foodorder on foodcustomer.cid = foodorder.cid where  
orderno < 5;
```

OUTPUT:

```
mysql> use dbms_record_work;  
Database changed  
mysql> desc foodorder;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| ORDERNO | int | YES | | NULL | |  
| CID | int | YES | | NULL | |  
| ORDERS | date | YES | | NULL | |  
+-----+-----+-----+-----+-----+-----+  
3 rows in set (0.08 sec)
```

```
mysql> INSERT INTO foodorder VALUES(1,123,'2002-02-01');  
Query OK, 1 row affected (0.01 sec)  
  
mysql> INSERT INTO foodorder VALUES(2,456,'2003-02-15');  
Query OK, 1 row affected (0.01 sec)  
  
mysql> INSERT INTO foodorder VALUES(3,789,'2002-02-21');  
Query OK, 1 row affected (0.01 sec)  
  
mysql> INSERT INTO foodorder VALUES(4,101,'2003-03-16');  
Query OK, 1 row affected (0.01 sec)  
  
mysql> INSERT INTO foodorder VALUES(5,121,'2002-03-17');  
Query OK, 1 row affected (0.01 sec)  
  
mysql> SELECT * FROM foodorder;  
+-----+-----+-----+  
| ORDERNO | CID | ORDERS |  
+-----+-----+-----+  
| 1 | 123 | 2002-02-01 |  
| 2 | 456 | 2003-02-15 |  
| 3 | 789 | 2002-02-21 |  
| 4 | 101 | 2003-03-16 |  
| 5 | 121 | 2002-03-17 |  
+-----+-----+-----+  
5 rows in set (0.00 sec)
```

```

mysql> CREATE TABLE foodcustomer(cid NUMBER(10),cname VARCHAR(10));
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual
L server version for the right syntax to use near 'NUMBER(10),cname VAR
mysql> CREATE TABLE foodcustomer(cid INTEGER(10),cname VARCHAR(10));
Query OK, 0 rows affected, 1 warning (0.05 sec)

mysql> INSERT INTO foodcustomer VALUES(123,'john');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO foodcustomer VALUES(456,'David');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO foodcustomer VALUES(789,'joe');
    ' > '
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual
L server version for the right syntax to use near '?joe';
'' at line 1
mysql> INSERT INTO foodcustomer VALUES(789,'joe');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO foodcustomer VALUES(101,'emma');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO foodcustomer VALUES(121,'hill');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM foodcustomer;;
+-----+-----+
| cid | cname |
+-----+-----+
| 123 | john  |
| 456 | David |
| 789 | joe   |
| 101 | emma  |
| 121 | hill  |
+-----+-----+
5 rows in set (0.00 sec)

```

```

MySQL 8.0 Command Line Client
mysql> SELECT * FROM FOODCUSTOMER JOIN FOODORDER ON FOODCUSTOMER.CID=FOODORDER.CID
-> ;
+-----+-----+-----+-----+-----+
| cid | cname | ORDERNO | CID | ORDERS |
+-----+-----+-----+-----+-----+
| 123 | john  | 1 | 123 | 2002-02-01 |
| 456 | David | 2 | 456 | 2003-02-15 |
| 789 | joe   | 3 | 789 | 2002-02-21 |
| 101 | emma  | 4 | 101 | 2003-03-16 |
| 121 | hill  | 5 | 121 | 2002-03-17 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> SELECT * FROM foodcustomer JOIN foodorder ON foodcustomer.cid = foodorder.cid where cname = 'emma';
+-----+-----+-----+-----+-----+
| cid | cname | ORDERNO | CID | ORDERS |
+-----+-----+-----+-----+-----+
| 101 | emma  | 4 | 101 | 2003-03-16 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM FOODCUSTOMER JOIN FOODORDER ON FOODCUSTOMER.CID = FOODORDER.CID WHERE ORDERNO<5;
+-----+-----+-----+-----+-----+
| cid | cname | ORDERNO | CID | ORDERS |
+-----+-----+-----+-----+-----+
| 123 | john  | 1 | 123 | 2002-02-01 |
| 456 | David | 2 | 456 | 2003-02-15 |
| 789 | joe   | 3 | 789 | 2002-02-21 |
| 101 | emma  | 4 | 101 | 2003-03-16 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```