

PRACTICAL : 1

SQL-

- SQL stands for Structured Query Language
- SQL lets you access and manipulate databases
- SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987.

Functions:

- SQL can execute queries against a database
- SQL can retrieve data from a database
- SQL can insert records in a database
- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database
- SQL can create stored procedures in a database
- SQL can create views in a database

- SQL can set permissions on tables, procedures, and views.

Types of SQL Commands:

1. DDL (Data Definition Language):

- o DDL changes the structure of the table like creating a table, deleting a table, altering a table, etc.
- o All the command of DDL are auto-committed that means it permanently save all the changes in the database.

Commands Under DDL:

- o CREATE
- o ALTER
- o DROP

2. DML (Data Manipulation Language):

- o DML commands are used to modify the database. It is responsible for all form of changes in the database.
- o The command of DML is not auto-committed that means it can't permanently save all the changes in the database. They can be rollback.

Commands Under DML:

- o INSERT
- o UPDATE
- o DELETE

3. DQL (Data Query Language):

DQL is used to fetch the data from the database.

Commands Under DQL:

- o SELECT

PRACTICAL : 2

(21.01.2021)

1. Insert rows in each table.

```
mysql> insert into employee(e_id,name,age,Dob,salary,city,dno) values('kual@','kunal',21,'2000-08-13',25000,'kolikata',96355),('kun@','kuntal',31,'2000-11-23',100000,'delhi',90355),('puc@','puchu',20,'2001-05-14',50000,'pune',8238),('bap@','bapan',30,'1995-12-03',60000,'gurgaon',89674),('mou@','priya',19,'2000-07-13',20000,'mumbai',34566);
Query OK, 5 rows affected (0.00 sec)
Records: 5 Duplicates: 0 Warnings: 0

mysql> select * from employee;
+-----+-----+-----+-----+-----+-----+-----+
| E_id | name | age | Dob | salary | city | dno |
+-----+-----+-----+-----+-----+-----+-----+
| kual@ | kunal | 21 | 2000-08-13 | 25000 | kolikata | 96355 |
| kun@ | kuntal | 31 | 2000-11-23 | 100000 | delhi | 90355 |
| puc@ | puchu | 20 | 2001-05-14 | 50000 | pune | 8238 |
| bap@ | bapan | 30 | 1995-12-03 | 60000 | gurgaon | 89674 |
| mou@ | priya | 19 | 2000-07-13 | 20000 | mumbai | 34566 |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> insert into department values( 'A001','set','gurgaon'),('A002','ssb','delhi'),('A003','ssd','noida');
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select * from department;
+-----+-----+-----+
| d_id | name | location |
+-----+-----+-----+
| A001 | set | gurgaon |
| A002 | ssb | delhi |
| A003 | ssd | noida |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> create table works(e_id varchar(5),d_id varchar(5),hours integer);
Query OK, 0 rows affected (0.02 sec)

mysql> desc works;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| e_id | varchar(5) | YES | | NULL | |
| d_id | varchar(5) | YES | | NULL | |
| hours | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> insert into works values ('oj@12','A12',8),('ip@13','A13',9),('hi@34','A14',7);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select * from works;
+-----+-----+-----+
| e_id | d_id | hours |
+-----+-----+-----+
| oj@12 | A12 | 8 |
| ip@13 | A13 | 9 |
| hi@34 | A14 | 7 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> select * from student;
```

s_id	sname
AU001	kunal
AU002	vinayak
AU003	akshat
AU004	tushar
AU005	randi

```
5 rows in set (0.00 sec)
```

2. Increase the size of the column city to 10 in the table employee.

```
mysql> alter table employee modify city char(10);
Query OK, 5 rows affected (0.07 sec)
Records: 5 Duplicates: 0 Warnings: 0

mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
E_id	varchar(5)	YES		NULL	
name	char(7)	YES		NULL	
age	int	YES		NULL	
Dob	date	YES		NULL	
salary	int	YES		NULL	
city	char(10)	YES		NULL	
dno	varchar(5)	YES		NULL	

```
7 rows in set (0.00 sec)
```

3. Rename the table department to office.

```
mysql> alter table deperment rename to office;
Query OK, 0 rows affected (0.02 sec)

mysql> desc office;
```

Field	Type	Null	Key	Default	Extra
d_id	varchar(5)	YES		NULL	
name	char(7)	YES		NULL	
location	char(7)	YES		NULL	

```
3 rows in set (0.00 sec)
```

4. Drop the table student.

```
mysql> drop table student;
Query OK, 0 rows affected (0.01 sec)

mysql> desc student;
ERROR 1146 (42S02): Table 'db.student' doesn't exist
```

5.

```
mysql> delete from employee where dno=12;
Query OK, 2 rows affected (0.01 sec)

mysql> select * from employee;
```

e_id	ename	age	dob	salary	city	dno
a1011	neerj	23	1998-05-05	40000	gurgaon	11
a1013	varun	28	1993-09-24	25000	delhi	11
a1015	disha	31	1990-08-19	50000	noida	11

```
3 rows in set (0.00 sec)
```

6. Increase the salary by 10% of those employees having a salary less than 10000.

```
mysql> update employee set salary= salary + (salary * 10 / 100) where salary < 10000;
Query OK, 3 rows affected (0.02 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> select * from employee;
```

E_id	name	Dob	salary	address	dno	state
kual@	kunal	2000-08-13	25000	kolkata	96355	westbengal
kun@	kuntal	2000-11-23	110000	delhi	90355	delhi
puc@	puchu	2001-05-14	55000	pune	8238	maharashtra
bap@	bapan	1995-12-03	66000	gurgaon	89674	haryana
mou@	priya	2000-07-13	20000	mumbai	34566	maharashtra

```
5 rows in set (0.00 sec)
```

7. Add a column state with char (10) in the table employee.

```
mysql> alter table employee modify city char(10);  
Query OK, 5 rows affected (0.07 sec)  
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
E_id	varchar(5)	YES		NULL	
name	char(7)	YES		NULL	
age	int	YES		NULL	
Dob	date	YES		NULL	
salary	int	YES		NULL	
city	char(10)	YES		NULL	
dno	varchar(5)	YES		NULL	

7 rows in set (0.00 sec)

8. Insert values for the newly added column state in the employee table.

```
mysql> select * from employee;
```

E_id	name	Dob	salary	address	dno	state
kual@	kunal	2000-08-13	25000	kolkata	96355	NULL
kun@	kuntal	2000-11-23	100000	delhi	90355	NULL
puc@	puchu	2001-05-14	50000	pune	8238	NULL
bap@	bapan	1995-12-03	60000	gurgaon	89674	NULL
mou@	priya	2000-07-13	20000	mumbai	34566	NULL

```
5 rows in set (0.00 sec)
```

```
mysql> update employee set state='westbengal' where name = 'kunal';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> update employee set state='delhi' where name = 'kuntal';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> update employee set state='maharashtra' where name = 'puchu';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> update employee set state='haryana' where name = 'bapan';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> update employee set state='maharashtra' where name = 'priya';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> select * from employee;
```

E_id	name	Dob	salary	address	dno	state
kual@	kunal	2000-08-13	25000	kolkata	96355	westbengal
kun@	kuntal	2000-11-23	100000	delhi	90355	delhi
puc@	puchu	2001-05-14	50000	pune	8238	maharashtra
bap@	bapan	1995-12-03	60000	gurgaon	89674	haryana
mou@	priya	2000-07-13	20000	mumbai	34566	maharashtra

```
5 rows in set (0.00 sec)
```

9. Change the name of the city in the employee to the address.


```
mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
E_id	varchar(5)	YES		NULL	
name	char(7)	YES		NULL	
age	int	YES		NULL	
Dob	date	YES		NULL	
salary	int	YES		NULL	
city	char(10)	YES		NULL	
dno	varchar(5)	YES		NULL	
state	char(10)	YES		NULL	

```
8 rows in set (0.00 sec)

mysql> alter table employee change city to address;
ERROR 1064 (42000): You have an error in your SQL syntax; check the
mysql> alter table employee change city address(10);
ERROR 1064 (42000): You have an error in your SQL syntax; check the
mysql> alter table employee change city address;
ERROR 1064 (42000): You have an error in your SQL syntax; check the
mysql> alter table employee change city address char(10);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
E_id	varchar(5)	YES		NULL	
name	char(7)	YES		NULL	
age	int	YES		NULL	
Dob	date	YES		NULL	
salary	int	YES		NULL	
address	char(10)	YES		NULL	
dno	varchar(5)	YES		NULL	
state	char(10)	YES		NULL	

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

10. Add a column manager_id with varchar(5) in the table department.

```
mysql> alter table office add column manager_id varchar(5);
ERROR 1146 (42S02): Table 'db.office' doesn't exist
mysql> alter table department add column manager_id varchar(5);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc department;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| d_id       | varchar(5) | YES  |     | NULL    |       |
| name       | char(7)    | YES  |     | NULL    |       |
| location   | char(7)    | YES  |     | NULL    |       |
| manager_id | varchar(5) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from department;
+-----+-----+-----+-----+
| d_id | name | location | manager_id |
+-----+-----+-----+-----+
| A001 | set  | gurgaon  | NULL       |
| A002 | ssb  | delhi    | NULL       |
| A003 | ssd  | noida    | NULL       |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

11. Now insert values for the newly added column `manager_id` in the department.

```
mysql> update department set manager_id='Yo@54' where d_id='A001';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update department set manager_id='jo@12' where d_id='A002';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update department set manager_id='rt@90' where d_id='A003';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from department;
+-----+-----+-----+-----+
| d_id | name | location | manager_id |
+-----+-----+-----+-----+
| A001 | set  | gurgaon  | Yo@54      |
| A002 | ssb  | delhi    | jo@12      |
| A003 | ssd  | noida    | rt@90      |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

PRACTICAL : 3

(28.02.2021)

1. List down the employees who are having salaries greater than equal to 20000 and less than equal to 50000.

```
mysql> select * from employee where salary>=20000 and salar<=50000;
ERROR 1054 (42S22): Unknown column 'salar' in 'where clause'
mysql> select * from employee where salary>=20000 and salary<=50000;
```

E_id	name	Dob	salary	address	dno	state
kual@	kunal	2000-08-13	25000	kolkata	96355	westbengal
puc@	puchu	2001-05-14	50000	pune	8238	maharashtra
mou@	priya	2000-07-13	20000	mumbai	34566	maharashtra

3 rows in set (0.00 sec)

2. List down the departments located in Noida.

```
mysql> select * from department;
```

d_id	name	location	manager_id
A001	set	gurgaon	Yo@54
A002	ssb	delhi	jo@12
A003	ssd	noida	rt@90

3 rows in set (0.00 sec)

```
mysql> select * from department where location='noida';
```

d_id	name	location	manager_id
A003	ssd	noida	rt@90

1 row in set (0.00 sec)

3. List the eid, name of those employees whose name starts with the letter k and ends with the letter l.

```
Database changed
mysql> select e_id from employee where name like 'k%l';
+-----+
| e_id |
+-----+
| kunal |
| kunal |
+-----+
2 rows in set (0.00 sec)

mysql> select e_id from employee where name like 'a%a';
Empty set (0.00 sec)
```

4. List those employees who are having an age greater than 30 yrs and are residing in Delhi.

```
mysql> select name from employee where age > 30;
+-----+
| name |
+-----+
| kunal |
| bapan |
+-----+
2 rows in set (0.00 sec)
```

5. List the eid, age, name of employees living in either Noida, Delhi, or Gurgaon using in the predicate.

```
mysql> select e_id, name, age from employee where address in ('noida', 'gurgaon', 'delhi');
+-----+-----+-----+
| e_id | name | age |
+-----+-----+-----+
| kun@ | kunal | 31 |
| bap@ | bapan | 38 |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

6. List the details of the departments whose location name has the second character as a and second last character as a.

```
mysql> select e_id from employee where name like '__a%_a';  
Empty set (0.00 sec)
```

```
mysql> select e_id from employee where name like '__n%_l';
```

e_id
kual@
kun@

```
2 rows in set (0.00 sec)
```

PRACTICAL : 4

Q2- Insert the following data into these respective tables:

Data for Client Master:

<u>Clientno</u>	<u>Name</u>	<u>city</u>	<u>state</u>	<u>bal.due</u>
0001	Ivan	Bombay	Maharashtra	15000
0002	Vandana	Madras	Tamilnadu	20000
0003	Pramada	Bombay	Maharashtra	5000
0004	Basu	Bombay	Maharashtra	30000
0005	Ravi	Delhi	null	20000
0006	Rukmini	Bombay	Maharashtra	10000

Data for Product Master:

<u>Product No.</u>	<u>Description</u>	<u>Qty</u>	<u>Sell price</u>
P00001	1.44floppies	100	500
P03453	Monitors	10	11200
P06734	Mouse	20	1050
P07865	1.22 floppies	100	525
P07868	Keyboards	10	3050
P07885	CD Drive	10	5100
P07965	540 HDD	10	8000
P07975	1.44 Drive	10	1050
P08865	1.22 Drive	5	1050

```
mysql> select * from ProductMaster;
```

ProductNo	Description	Qty	Sellprice
P0001	1.44floppies	100	500
P03453	Monitors	10	11200
P06734	Mouse	20	1050
P07865	1.22floppies	100	525
P07868	Keyboards	10	3050
P07885	CDdrive	10	5100
P07965	540HDD	10	8000
P07975	1.44Drive	10	1050
P08865	1.22Drive	10	1050

```
9 rows in set (0.00 sec)
```

```
mysql> select * from client_master;
```

clientno	name	city	state	bal_due
C0001	Ivan	Bombay	Maharashtra	15000
C0002	Vandana	Madras	Tamilnadu	200000
C0003	Pramada	Bombay	Maharashtra	50000
C0004	Basu	Bombay	Maharashtra	30000
C0005	Ravi	Delhi	null	20000
C0006	Rukmini	Bombay	Maharashtra	10000

```
6 rows in set (0.00 sec)
```

1.Find the names of all the clients.

```
mysql> select name from client_master;
```

name
Ivan
Vandana
Pramada
Basu
Ravi
Rukmini

```
6 rows in set (0.00 sec)
```

2.Retrieve the list of names and cities all the clients.

```
mysql> select name,city from client_master;
```

name	city
Ivan	Bombay
Vandana	Madras
Pramada	Bombay
Basu	Bombay
Ravi	Delhi
Rukmini	Bombay

```
6 rows in set (0.00 sec)
```

3.List the various product available from the ProductMaster table.

```
mysql> select Description from ProductMaster;
```

Description
1.44floppies
Monitors
Mouse
1.22floppies
Keyboards
CDdrive
540HDD
1.44Drive
1.22Drive

```
9 rows in set (0.00 sec)
```

4. List all the clients who are located in Bombay.

```
mysql> select name from client_master where city='Bombay';
```

name
Ivan
Pramada
Basu
Rukmini

```
4 rows in set (0.00 sec)
```

5. Display the information for the client no C0001 and C0002.

```
mysql> select * from client_master where clientno in('C0001','C0002');
```

clientno	name	city	state	bal_due
C0001	Ivan	Bombay	Maharashtra	15000
C0002	Vandana	Madras	Tamilnadu	200000

```
2 rows in set (0.00 sec)
```

6. Find the products with a description as '1.44dDrive' and '1.22Drive'.


```
mysql> select * from ProductMaster where Description in('1.44Drive','1.22Drive');
```

ProductNo	Description	Qty	Sellprice
P07975	1.44Drive	10	1050
P08865	1.22Drive	10	1050

```
2 rows in set (0.00 sec)
```

7.Find all products whose sell price is greater than 5000.

```
mysql> select * from ProductMaster where Sellprice>5000;
```

ProductNo	Description	Qty	Sellprice
P03453	Monitors	10	11200
P07885	CDdrive	10	5100
P07965	540HDD	10	8000

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

8.Find the products whose selling price is greater than 2000 and less than or equal to 5000.

```
mysql> select * from ProductMaster where Sellprice>2000 and Sellprice<=5000;
```

ProductNo	Description	Qty	Sellprice
P07868	Keyboards	10	3050

```
1 row in set (0.00 sec)
```

9.List the name, city, and state of the client not in the state of Maharashtra.

```
mysql> select name,city,state from client_master where state not in('Maharashtra');
```

name	city	state
Vandana	Madras	Tamilnadu
Ravi	Delhi	null

```
2 rows in set (0.00 sec)
```

10.Change the city of a client no C0005 to Bombay.

```
mysql> update client_master set city='Bombay' where clientno='C0005';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from client_master;
```

clientno	name	city	state	bal_due
C0001	Ivan	Bombay	Maharashtra	15000
C0002	Vandana	Madras	Tamilnadu	200000
C0003	Pramada	Bombay	Maharashtra	50000
C0004	Basu	Bombay	Maharashtra	30000
C0005	Ravi	Bombay	null	20000
C0006	Rukmini	Bombay	Maharashtra	10000

```
6 rows in set (0.00 sec)
```

11.Delete all the details of the product where quantity is equal to 100.

```
mysql> delete from ProductMaster where Qty=100;
ERROR 2006 (HY000): MySQL server has gone away
No connection. Trying to reconnect...
Connection id: 20
Current database: db

Query OK, 2 rows affected (0.16 sec)

mysql> select * from ProductMaster;
```

ProductNo	Description	Qty	Sellprice
P03453	Monitors	10	11200
P06734	Mouse	20	1050
P07868	Keyboards	10	3050
P07885	CDdrive	10	5100
P07965	540HDD	10	8000
P07975	1.44Drive	10	1050
P08865	1.22Drive	10	1050

```
7 rows in set (0.00 sec)
```

12.Add a column telephone to the table client_master with the domain number(10,0) and add the values for it.

```
mysql> alter table client_master add column telephone integer;
Query OK, 0 rows affected (0.09 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc client_master;
```

Field	Type	Null	Key	Default	Extra
clientno	varchar(6)	YES		NULL	
name	char(8)	YES		NULL	
city	char(8)	YES		NULL	
state	char(13)	YES		NULL	
bal_due	int	YES		NULL	
telephone	int	YES		NULL	

```
6 rows in set (0.03 sec)
```

13. Change the size of the selling price to (10,2).

```
mysql> alter table ProductMaster modify Sellprice decimal(10,2);
Query OK, 7 rows affected (0.05 sec)
Records: 7 Duplicates: 0 Warnings: 0

mysql> desc ProductMaster;
```

Field	Type	Null	Key	Default	Extra
ProductNo	varchar(10)	YES		NULL	
Description	varchar(20)	YES		NULL	
Qty	int	YES		NULL	
Sellprice	decimal(10,2)	YES		NULL	

```
4 rows in set (0.00 sec)
```

14. Find the names of the clients having "a" as the second letter in their names.

```
mysql> select name from client_master where name like '__a%';
```

name
Ivan
Pramada

```
2 rows in set (0.00 sec)
```

PRACTICAL : 5

1.Count the total number of orders placed.

```
mysql> select sum(Qty) from ProductMaster;
```

sum(Qty)
80

```
1 row in set (0.02 sec)
```

2.Calculate the average price of all the products.

```
mysql> select avg(Sellprice) "Avg_sellprice" from ProductMaster;
```

Avg_sellprice
4357.142857

```
1 row in set (0.00 sec)
```

3.Determine the maximum and minimum product prices.

```
mysql> select min(Sellprice) Min_sellprice,max(Sellprice) Max_sellprice from ProductMaster;
```

Min_sellprice	Max_sellprice
1050.00	11200.00

```
1 row in set (0.00 sec)
```

4.Count the number of products having prices greater than equal to 1500.

```
mysql> select count(Sellprice) "No of products > 1500" from ProductMaster where Sellprice>=1500;
```

No of products > 1500
4

```
1 row in set (0.00 sec)
```

5.Find all the products whose quantity is less than 50.

```
mysql> select count(Qty) "Qty less 50" from ProductMaster where Qty<50;
```

Qty less 50
7

```
1 row in set (0.00 sec)
```

6.Retrieve the city of all clients in lower case letters.

```
mysql> select name,lower(city) from client_master;
```

name	lower(city)
Ivan	bombay
Vandana	madras
Pramada	bombay
Basu	bombay
Ravi	bombay
Rukmini	bombay

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

7.Update the table client by changing the first letter of the client in caps.

```
mysql> update client_master set name=concat(ucase(left(name,1)),substring(name,2));
Query OK, 0 rows affected (0.01 sec)
Rows matched: 6 Changed: 0 Warnings: 0
```

```
mysql> select * from client_master;
```

clientno	name	city	state	bal_due	telephone
C0001	Ivan	Bombay	Maharashtra	15000	NULL
C0002	Vandana	Madras	Tamilnadu	200000	NULL
C0003	Pramada	Bombay	Maharashtra	50000	NULL
C0004	Basu	Bombay	Maharashtra	30000	NULL
C0005	Ravi	Bombay	null	20000	NULL
C0006	Rukmini	Bombay	Maharashtra	10000	NULL

```
6 rows in set (0.00 sec)
```

8.Display the state of client no C0002 in upper case.

```
mysql> select clientno,upper(state) STATE from client_master where clientno="C0002";
```

clientno	STATE
C0002	TAMILNADU

```
1 row in set (0.00 sec)
```

9.Retrieve the city of client by extracting substring from the city column with m=1,n=4

```
mysql> select name,substring(city,1,4) City from client_master;
```

name	City
Ivan	Bomb
Vandana	Madr
Pramada	Bomb
Basu	Bomb
Ravi	Bomb
Rukmini	Bomb

```
6 rows in set (0.00 sec)
```

10.Calculate the length of names of clients having client_no=C0004

```
mysql> select name,length(name) Length from client_master where clientno="C0004";
```

name	Length
Basu	4

```
1 row in set (0.00 sec)
```

11.Update the table product by adding Rs 1000 to the sell price of all products

```
mysql> select * from ProductMaster;
```

ProductNo	Description	Qty	Sellprice
P03453	Moniters	10	12200.00
P06734	Mouse	20	2050.00
P07868	Keyboards	10	4050.00
P07885	CDdrive	10	6100.00
P07965	540HDD	10	9000.00
P07975	1.44Drive	10	2050.00
P08865	1.22Drive	10	2050.00

```
7 rows in set (0.00 sec)
```

PRACTICAL : 6

1. Create a table Emp with attributes ename, ecity, salary, enumber with following constraints: enumber is the primary key and should start with 'E', ecity should have default values as Delhi.

```
mysql> create table Emp(Ename varchar(10),Ecity varchar(5) default 'Delhi',Enumber varchar(5) primary key check(Enumber like 'E%'));
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> desc Emp;
```

Field	Type	Null	Key	Default	Extra
Ename	varchar(10)	YES		NULL	
Ecity	varchar(5)	YES		Delhi	
Enumber	varchar(5)	NO	PRI	NULL	

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

```
mysql> alter table Emp add salary integer;
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc Emp;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Ename | varchar(10) | YES  |     | NULL    |       |
| Ecity | varchar(5)  | YES  |     | Delhi   |       |
| Enumber | varchar(5) | NO   | PRI | NULL    |       |
| salary | int         | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

2. Create another Table Department with attributes did, budget and manager_id, dname with following constraints: budget should not be less than 10000.

```
mysql> create table Department(Did varchar(5),Budget int check(Budget<=10000),Manager_id varchar(5),Dname varchar(10));
Query OK, 0 rows affected (0.02 sec)

mysql> desc Department;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Did   | varchar(5) | YES  |     | NULL    |       |
| Budget | int        | YES  |     | NULL    |       |
| Manager_id | varchar(5) | YES  |     | NULL    |       |
| Dname | varchar(10) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

3. Create another Table Works with attributes did and eid & combination of both as primary key.

```
mysql> create table works(did varchar(5),eid varchar(5),primary key(did,eid));
Query OK, 0 rows affected (0.01 sec)

mysql> desc works;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| did   | varchar(5) | NO   | PRI | NULL    |       |
| eid   | varchar(5) | NO   | PRI | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

4. Alter table Department and make did as the primary key.


```
mysql> alter table Department add constraint pk primary key(Did);
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc Department;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Did   | varchar(5) | NO   | PRI | NULL    |       |
| Budget | int       | YES  |     | NULL    |       |
| Manager_id | varchar(5) | YES  |     | NULL    |       |
| Dname  | varchar(10) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

5.Alter table Department and make manager_id as the foreign key.

```
mysql> alter table Department add constraint fg foreign key(Manager_id) references Emp (Enumber);
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc Department;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Did   | varchar(5) | NO   | PRI | NULL    |       |
| Budget | int       | YES  |     | NULL    |       |
| Manager_id | varchar(5) | YES  | MUL | NULL    |       |
| Dname  | varchar(10) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

6.Alter table Works and make did and eid as foreign key.

```
mysql> alter table works add constraint fk foreign key(did) references Department(Did);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table works add constraint fk foreign key(eid) references Emp(Enumber);
ERROR 1826 (HY000): Duplicate foreign key constraint name 'fk'
mysql> alter table works add constraint fk1 foreign key(eid) references Emp(Enumber);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc works;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| did   | varchar(5) | NO   | PRI | NULL    |       |
| eid   | varchar(5) | NO   | PRI | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

7. Create table book with attributes ISBN, title, price, pid with following constraints: ISBN is the primary key, price should be ≥ 20 / ≤ 200 , title should not have null values.

```
mysql> create table book(ISBN integer primary key,title char (15) default 'not null',price integer check(price>=20 and price<=200),pid varchar(15));
Query OK, 0 rows affected (0.02 sec)

mysql> desc book;
```

Field	Type	Null	Key	Default	Extra
ISBN	int	NO	PRI	NULL	
title	char(15)	YES		not null	
price	int	YES		NULL	
pid	varchar(15)	YES		NULL	

4 rows in set (0.00 sec)

8. Create table publisher with attributes pid, pname, state with following constraints: pname should not have null values; state should be either Haryana, Tamil nadu, Maharashtra.

```
mysql> create table publisher(pid varchar(15),pname char(20) default 'NOT NULL',state char(20) check(state in ('Haryana','Tamilnadu','Maharastra')));
Query OK, 0 rows affected (0.02 sec)

mysql> desc publisher;
```

Field	Type	Null	Key	Default	Extra
pid	varchar(15)	YES		NULL	
pname	char(20)	YES		NOT NULL	
state	char(20)	YES		NULL	

3 rows in set (0.01 sec)

9. Alter the table publisher by making pid as primary key.

```
mysql> alter table publisher add constraint primary key (pid);
Query OK, 0 rows affected (0.06 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc publisher;
```

Field	Type	Null	Key	Default	Extra
pid	varchar(15)	NO	PRI	NULL	
pname	char(20)	YES		NOT NULL	
state	char(20)	YES		NULL	

3 rows in set (0.00 sec)

10. Alter the table book by making pid as foreign key.

```
mysql> desc book;
```

Field	Type	Null	Key	Default	Extra
ISBN	int	NO	PRI	NULL	
title	char(15)	YES		not null	
price	int	YES		NULL	
pid	varchar(15)	YES	MUL	NULL	

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

PRACTICAL : 7

1. Create table sailors with attributes-s_id, s_name, s_age, s_rating with s_id as primary key.

```
mysql> desc sailors;
```

Field	Type	Null	Key	Default	Extra
s_id	varchar(20)	NO	PRI	NULL	
s_name	char(20)	YES		NULL	
s_age	int	YES		NULL	
s_rating	int	YES		NULL	

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

```
mysql> select * from sailors;
```

s_id	s_name	s_age	s_rating
S001	biplab	25	5
S002	rahim	28	6
S003	arindam	21	6
S004	tapu	38	8
S005	bapan	30	9

```
5 rows in set (0.00 sec)
```

2. Create table reserves with attributes-s_id, b_id, day (date) with (s_id+b_id) as primary key and s_id and b_id as foreign key.

```
mysql> desc reserves;
```

Field	Type	Null	Key	Default	Extra
s_id	varchar(20)	NO	PRI	NULL	
b_id	varchar(20)	NO	PRI	NULL	
day	date	YES		NULL	

```
3 rows in set (0.00 sec)
```

```
mysql> select * from reserves;
```

s_id	b_id	day
S002	B001	2021-04-02
S002	B003	2021-04-06
S003	B002	2021-05-23
S004	B005	2021-04-12
S005	B002	2021-04-17

```
5 rows in set (0.00 sec)
```

3. Create table boats with attributes-b_id, bname, bcolour with b_id as primary key.

```
mysql> desc boats;
```

Field	Type	Null	Key	Default	Extra
b_id	varchar(20)	NO	PRI	NULL	
b_name	char(20)	YES		NULL	
b_color	char(20)	YES		NULL	

```
3 rows in set (0.00 sec)
```

```
mysql> select * from boats;
```

b_id	b_name	b_color
B001	radha	red
B002	radha	green
B003	bazigar	green
B004	tismar	red
B005	khalnyak	red

```
5 rows in set (0.00 sec)
```

4. Find the sid of the sailors who have reserved red boat but not green boat.

```
mysql> select R.s_id from reserves R,boats B where R.b_id=B.b_id and B.b_color='red' and R.s_id not in (select R.s_id from reserves R,boats B where R.b_id=B.b_id and B.b_color='green');
```

s_id
S004

```
1 row in set (0.01 sec)
```

5.Find the sid of the sailors who have reserved red boat or green boat.

```
mysql> select R.s_id from reserves R,boats B where R.b_id=B.b_id and B.b_color in ('red','green');
+-----+
| s_id |
+-----+
| S002 |
| S003 |
| S005 |
| S002 |
| S004 |
+-----+
5 rows in set (0.01 sec)
```

6.Find the sid of the sailors who have reserved both red boat and green boat.

```
mysql> select R.s_id from reserves R,boats B where R.b_id=B.b_id and B.b_color='red' and B.b_color='green';
Empty set (0.00 sec)
```

PRACTICAL : 9

1.Find the age of youngest sailors for each rating level.

```
mysql> select s_rating,min(s_age) from sailors group by (s_rating);
+-----+-----+
| s_rating | min(s_age) |
+-----+-----+
| 5 | 25 |
| 6 | 28 |
| 4 | 21 |
| 8 | 38 |
| 9 | 30 |
+-----+-----+
5 rows in set (0.01 sec)
```

2.Find the age of youngest sailor who is eligible to vote for each rating level with at least two such sailor.

3. For each red boat, find the number of reservations for this boat.

```
mysql> select B.b_id ,B.b_name,B.b_color, count(R.b_id) from boats B natural join reserves R group by B.b_id having B.b_color="red";
```

b_id	b_name	b_color	count(R.b_id)
B001	radha	red	1
B005	khalnyak	red	1

2 rows in set (0.00 sec)

4. Find average age of sailor for each rating level that has at least two such sailors.

```
mysql> select s_rating,avg(s_age) average_age from sailors group by s_rating having count(s_rating)>1;
```

s_rating	average_age
6	24.5000

1 row in set (0.01 sec)

5. Write a query to print the following sentence :

Sailor named with sailor id.....has reserved boat number.....on date.....

```
mysql> select concat ("Sailor named ",S.s_name,"with sailor id ",S.s_id,"has reserved boat number ",R.b_id,"on date", R.day) detailed_information from sailors:
$ natural join reserves R;
```

detailed_information
Sailor named rahimwith sailor id S002has reserved boat number B001on date2021-04-02
Sailor named rahimwith sailor id S002has reserved boat number B003on date2021-04-06
Sailor named arindamwith sailor id S003has reserved boat number B002on date2021-05-23
Sailor named tapuwith sailor id S004has reserved boat number B005on date2021-04-12
Sailor named bapanwith sailor id S005has reserved boat number B002on date2021-04-17

5 rows in set (0.01 sec)

PRACTICAL : 8

Using natural join :

Find names of sailors who have reserved boat number B003.

```
mysql> select sailors.s_name from sailors natural join reserves where reserves.b_id="B003";
```

s_name
rahim

```
1 row in set (0.01 sec)
```

2. Find names of sailors who have reserved at least one boat.

```
mysql> select sailors.s_name from sailors natural join reserves R group by s_id;
```

s_name
rahim
arindam
bapan
tapu

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

3. Find names of sailors who have reserved a red boat.

```
mysql> select s_name from sailors S natural join reserves R natural join boats B where B.b_color="red";
```

s_name
rahim
tapu

```
2 rows in set (0.00 sec)
```

4. Find the colors of boat reserved by bapan.

```
mysql> select B.b_color from sailors S natural join reserves R natural join boats B where S.s_name="bapan";
```

b_color
green

```
1 row in set (0.00 sec)
```

Using Equi join :

5. Find names of sailors who have reserved boat number B003.

```
mysql> select S.s_name from sailors S join reserves R on S.s_id = R.s_id where R.b_id = "B003";
```

s_name
rahim

6. Find names of sailors who have reserved at least one boat.

```
mysql> select S.s_name from sailors S join reserves R on S.s_id = R.s_id group by S.s_name;
```

s_name
rahim
arindam
bapan
tapu

```
4 rows in set (0.00 sec)
```

7. Find names of sailors who have reserved a red boat

```
mysql> select S.s_name from sailors S join reserves R on S.s_id = R.s_id join boats B on B.b_id = R.b_id where B.b_color = 'red' group by S.s_name;
```

s_name
rahim
tapu

```
2 rows in set (0.00 sec)
```

8. Find the colors of boat reserved by bapan.

```
mysql> select B.b_color from sailors S join reserves R on S.s_id = R.s_id join boats B on B.b_id = R.b_id where S.s_name = "bapan";
```

b_color
green

```
1 row in set (0.01 sec)
```

Nested queries :

1. Find the sailors whose rating is better than some sailor called tapu.

```
mysql> select * from sailors where s_rating > (select s_rating from sailors where s_name = "tapu");
```

s_id	s_name	s_age	s_rating
S005	bapan	30	9

```
1 row in set (0.00 sec)
```

2. Find the sailors whose rating is better than every sailor called biplab.

```
mysql> select * from sailors where s_rating > (select s_rating from sailors where s_name = "biplab");
```

s_id	s_name	s_age	s_rating
S002	rahim	28	6
S003	arindam	21	6
S004	tapu	38	8
S005	bapan	30	9

```
4 rows in set (0.00 sec)
```

3. Find the sailor with highest rating.

```
mysql> select s_id, max(s_rating) from sailors group by (s_id);
```

s_id	max(s_rating)
S001	5
S002	6
S003	6
S004	8
S005	9

```
5 rows in set (0.00 sec)
```

Self join :

Create a table employee (emp_no, name, manager_no) where emp_no is the primary key and manager_no is the foreign key.

```
mysql> insert into employee values ('E01','basu','E02'),('E02','rukman','E05'),('E03','carol','E04'),('E04','bayross','NULL'),('E05','ivan','NULL');
Query OK, 5 rows affected (0.01 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql> select * from employee;
```

emp_no	name	manager_no
E01	basu	E02
E02	rukman	E05
E03	carol	E04
E04	bayross	NULL
E05	ivan	NULL

```
5 rows in set (0.00 sec)
```

1. Execute a query to retrieve names of the employees and the name of their respective managers from the employee table.

```
mysql> select E.name E_name ,M.name M_name from employee E join employee M on E.manager_no = M.emp_no;
```

E_name	M_name
basu	rukman
rukman	ivan
carol	bayross

```
3 rows in set (0.00 sec)
```

Outer join :

1.Create tables and insert these rows:

Students

Student_ID	Student_Name	Advisor_ID
1	Student_1	1
2	Student_2	8
4	Student_4	2
5	Student_5	3
7	Student_7	3
9	Student_9	1
10	Student_10	3

Advisors

Advisor_ID	Advisor_Name
1	Advisor 1
3	Advisor 3
5	Advisor 5

```
mysql> create table students(student_id int primary key,student_name char(20),advisor_id int);
Query OK, 0 rows affected (0.07 sec)
```

```
mysql> insert into students values(1,'student_1',1),(2,'student_2',8),(4,'student_4',2),(5,'student_5',3),(7,'student_7',3),(9,'student_9',1),(10,'student_10',3);
Query OK, 7 rows affected (0.01 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

```
mysql> select * from students;
```

student_id	student_name	advisor_id
1	student_1	1
2	student_2	8
4	student_4	2
5	student_5	3
7	student_7	3
9	student_9	1
10	student_10	3

```
7 rows in set (0.00 sec)
```

```
mysql> create table advisors(advisor_id int primary key,advisor_name varchar(20));
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> insert into advisors values(1,'advisor_1'),(3,'advisor_3'),(5,'advisor_5');
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

```
mysql> select * from advisors;
```

advisor_id	advisor_name
1	advisor_1
3	advisor_3
5	advisor_5

```
3 rows in set (0.00 sec)
```

2.Query for retrieving the student details along with their advisory detail.

```
mysql> select * from students S right join advisors A on S.advisor_id = A.advisor_id union select * from students S left join advisors A on S.advisor_id = A.advisor_id;
```

student_id	student_name	advisor_id	advisor_id	advisor_name
1	student_1	1	1	advisor_1
9	student_9	1	1	advisor_1
5	student_5	3	3	advisor_3
7	student_7	3	3	advisor_3
10	student_10	3	3	advisor_3
NULL	NULL	NULL	5	advisor_5
2	student_2	8	NULL	NULL
4	student_4	2	NULL	NULL

```
8 rows in set (0.06 sec)
```

3.Query for retrieving the details of all the student along with details of advisory (using outer join).

```
mysql> select * from students S left join advisors A on S.advisor_id = A.advisor_id ;
```

student_id	student_name	advisor_id	advisor_id	advisor_name
1	student_1	1	1	advisor_1
2	student_2	8	NULL	NULL
4	student_4	2	NULL	NULL
5	student_5	3	3	advisor_3
7	student_7	3	3	advisor_3
9	student_9	1	1	advisor_1
10	student_10	3	3	advisor_3

7 rows in set (0.01 sec)

4. Query for retrieving the details of student along with details of all the advisory (using outer join).

```
mysql> select * from students S right join advisors A on S.advisor_id = A.advisor_id ;
```

student_id	student_name	advisor_id	advisor_id	advisor_name
1	student_1	1	1	advisor_1
9	student_9	1	1	advisor_1
5	student_5	3	3	advisor_3
7	student_7	3	3	advisor_3
10	student_10	3	3	advisor_3
NULL	NULL	NULL	5	advisor_5

6 rows in set (0.00 sec)

PRACTICAL : 10

1. Basic definition and advantages of views.

Create a table staff with attributes staff_no, firstname, lastname, position, DOB, salary, branch_no with staff_no as primary key.

Insert six rows in it.

```
mysql> create table staff(staff_no varchar(20),firstname varchar(20),lastname varchar(20),position varchar(20),DOB date,salary int,branch_no varchar(20),primary key(staff_no));
Query OK, 0 rows affected (0.08 sec)
```

```
mysql> insert into staff values('S001','vinnet','negi','developer','2000-01-01',30000,'B01'),('S002','brotin','sharma','manager','2000-04-12',32000,'B03'),('S003','arindam','de','junior employee','2000-03-24',23000,'B02'),('S004','kuntal','acharya','ceo','1990-05-29',200000,'B02'),('S005','vinayak','vajpayee','employee','2000-04-12',35000,'B04'),('S006','raja','das','photographer','2000-07-27',20000,'B02');
Query OK, 6 rows affected (0.01 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

```
mysql> select * from staff;
+-----+-----+-----+-----+-----+-----+-----+
| staff_no | firstname | lastname | position | DOB | salary | branch_no |
+-----+-----+-----+-----+-----+-----+-----+
| S001 | vinnet | negi | developer | 2000-01-01 | 30000 | B02 |
| S002 | brotin | sharma | manager | 2000-04-12 | 32000 | B03 |
| S003 | arindam | de | junior employee | 2000-03-24 | 23000 | B02 |
| S004 | kuntal | acharya | ceo | 1990-05-29 | 200000 | B02 |
| S005 | vinayak | vajpayee | employee | 2000-04-12 | 35000 | B04 |
| S006 | raja | das | photographer | 2000-07-27 | 20000 | B02 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

2.Create a view so that manager at branch 200 can see only the details for staff who work in his/her branch office.

```
mysql> create view report as select * from staff where branch_no='B02';
Query OK, 0 rows affected (0.02 sec)

mysql> select * from report;
+-----+-----+-----+-----+-----+-----+-----+
| staff_no | firstname | lastname | position | DOB | salary | branch_no |
+-----+-----+-----+-----+-----+-----+-----+
| S001 | vinnet | negi | developer | 2000-01-01 | 30000 | B02 |
| S003 | arindam | de | junior employee | 2000-03-24 | 23000 | B02 |
| S004 | kuntal | acharya | ceo | 1990-05-29 | 200000 | B02 |
| S006 | raja | das | photographer | 2000-07-27 | 20000 | B02 |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

3.Create a view staffdetails200 that excludes salary info so that only managers can access the salary details for the staff who work at their branch.

```
mysql> create view staffdetails200 as select staff_no,firstname,lastname,position,DOB,branch_no from staff where branch_no='B02';
Query OK, 0 rows affected (0.01 sec)

mysql> select * from staffdetails200;
+-----+-----+-----+-----+-----+-----+
| staff_no | firstname | lastname | position | DOB | branch_no |
+-----+-----+-----+-----+-----+-----+
| S001 | vinnet | negi | developer | 2000-01-01 | B02 |
| S003 | arindam | de | junior employee | 2000-03-24 | B02 |
| S004 | kuntal | acharya | ceo | 1990-05-29 | B02 |
| S006 | raja | das | photographer | 2000-07-27 | B02 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```