

03.08.23

1.a) Write down the SQL commands to create a table of students with Name, Roll no, Department and year as fields.

sudo mysql;

create Database Disha;

use Disha;

create table student-14 (Name varchar(20),  
 Roll Int(10),  
 Dept varchar(10),  
 Year Int(5));

insert into student-14 values ('Riya', 003, 'CSE', 1);

insert into student-14 values ('Akash', 004, 'BCA', 3);

insert into student-14 values ('Joy', 013, 'MCA', 2);

insert into student-14 values ('Diya', 009, 'CSE', 2);

Select \* from student-14;

Name	Roll	Dept	Year
Riya	003	CSE	1
Akash	004	BCA	3
Joy	013	MCA	2
Diya	009	CSE	2

- b. Write down the SQL commands to create a table to store Id, Name, Salary, DOJ (Date of Joining), Address of Employee and -
- Find Id, & name of employee with more than 50K salary.
  - Find name of employee who joined before 01-01-2023.
  - Find Id of employee have salary less than 50K and in Delhi.
  - Update the location of E03 to Jaipur.

sudo mysql;

create Database Disha;

use Disha;

```
create table employee (Id      Varchar(5),
                      Name     varchar(20),
                      Salary   Int(6),
                      DOJ     Date,
                      Address  Varchar(100));
```

insert into employee values('E01', 'Ravi', 50000, '2023-05-01', 'Delhi');

insert into employee values('E02', 'Kishan', 55000, '2020-04-13', 'Kolkata');

insert into employee values('E03', 'Lavi', 60000, '1999-1-1', 'Tripura');

insert into employee values('E04', 'Devi', 45000, '2020-1-1', 'Chennai');

insert into employee values('E05', 'Nupur', 30000, '2015-2-1', 'Delhi');

Select \* from employee;

Select Id, Name from employee where salary >= 50000;

Select Name from employee where DOJ < '2020-01-01';

Select Id from employee where salary <= 50000 and Address = 'Delhi';

Update employee set Address = 'Jaipur' where Id = 'E03';



Id	Name	Salary	DOJ	Address
E01	Ravi	50000	2023-05-01	delhi
E02	Kishan	55000	2020-04-13	kolkata
E03	Lavi	60000	1999-01-01	Tripura
E04	Devi	45000	2020-01-01	chennai
E05	Nupur	30000	2015-02-01	delhi

i. Employee with 50K + salary

Id	Name
E02	Kishan
E03	Lavi
E04	Devi

ii. Employee joined before 2023-01-01

Name
Kishan
Lavi
Devi
Nupur

iii. Employee in Delhi with salary < 50K

ID
E05

iv. Update finished

✓  
S. Far. 23  
10.08.23

- c. Create a Test table with id, name and city. Insert data into the table and display the name & then the entire table.

```

sudo mysql;
create Database Disha;
use Disha;
create table Test(E_id varchar(10), Name varchar(100),
                   city varchar(100));
insert into Test values ('C101', 'Deepak', 'Mumbai');
insert into Test values ('C102', 'Aman', 'Banglore');
insert into Test values ('C103', 'RajKumar', 'Kolkata');
insert into Test values ('C104', 'Aman', 'Delhi');
insert into Test values ('C105', 'Kunal', 'Mumbai');
select * from Test;
select Name from Test;

```

E-id	Name	city
C101	Deepak	Mumbai
C102	Aman	Banglore
C103	RajKumar	Kolkata
C104	Aman	Delhi
C105	Kunal	Mumbai

Name
Deepak
Aman
RajKumar
Aman
Kunal

2. Create two tables customers (cust\_id(Primary key), Name, Age, Address, Salary) and shipment (shipment\_no (Primary key), cust\_id, weight). Insert data into table and then add new column start\_city in shipment table. Add constraints as shipment cust\_id references customers (cust\_id) and declare not null :name. write SQL statements to retrieve the queries -
- i. Describe customer & shipment table
  - ii. Show the tables.
  - iii. Find start-city of shipment table
  - iv. Change Age of Kaushik to 35.
  - v. Change salary of Ramesh to 15,500.00.
  - vi. Delete data of 3202 in shipment table.

sudo mysql;

use Disha;

create table customers (cust\_id varchar(20), Name  
varchar(20), Age int, Address  
varchar(20), salary float);

ALTER TABLE customers ADD PRIMARY KEY(cust\_id);

create table shipment (shipment\_no varchar(20) primary key,  
cust\_id varchar(20), weight int);

insert into customers values ('C001', 'Ramesh', 32, 'Ahmedabad',  
12000.00);

insert into customers values ('C002', 'Kilan', 25, 'Delhi', 14500.00);

insert into customers values ('C003', 'Kaushik', 23, 'Kota', 18000.00);

insert into customers values ('C004', 'Muffy', 25, 'Mumbai', 15000.00);

ALTER TABLE shipments ADD start\_city varchar(20);

ALTER TABLE shipments ADD FOREIGN KEY(cust\_id) REFERENCES  
customers (cust\_id);

ALTER TABLE customers modify name varchar(4) NOT  
NULL;

insert into shipment values ('3201', 'C001', 120, 'Indore');

insert into shipment values ('3202', 'C002', 210, 'Kolkata');

insert into shipment values ('3203', '2003', 320, Mumbai);

desc customers;

desc shipment;

update customers set Age = 35 where Name = 'Kaushik';

update customers set salary = 15500.00 where Name = 'Ramesh';

Delete from shipment where shipment\_no. = '3202';

commit;

exit;

customer table -

cust_id	Name	Age	Address	Salary
C001	Ramesh	32	Ahmedabad	12000.00
C002	Kilan	25	Delhi	14500.00
C003	Kaushik	23	Kota	18000.00
C004	Muffy	25	Mumbai	15000.00

shipment table -

shipment_no	cust_id	weight	start_city
3201	C001	120	Indore
3202	C002	210	Kolkata
3203	C003	320	Mumbai

KPU  
17/8/23

17.08.23

3. Create 3 tables - cust (cust\_id, lname, fname, area, phone\_no), movie (mv\_no, title, type, star), invoice (inv\_no, mv\_no, cust\_id, issue\_date, return\_date). Add the constraints - Invoice (cust\_id), (mv\_no) from cust & movie. Declare fname, type, title → NOT NULL and add Price in movie of float (8,2). And insert respective data in them.

sudo mysql;

use Disha;

create table cust (cust\_id varchar(3) PRIMARY KEY,  
lname varchar(15), fname varchar(15),  
Area varchar(2), phone\_no(8));

create table movie(mv\_no varchar(2) PRIMARY KEY, Title  
varchar(25), Type varchar(10), star  
varchar(25));

create table invoice (inv\_no varchar(3) PRIMARY KEY,  
mv\_no varchar(2), cust\_id varchar(3),  
issue-date date, return-date date));

insert into cust values('a01', 'Bayross', 'Ivan', 'Sa', 6125467);  
insert into cust values('a02', 'Saitwal', 'Vandana', 'mu', 5560379);  
insert into cust values('a03', 'Jagusti', 'Pramada', 'da', 4563891);  
insert into cust values('a04', 'Navindgi', 'Basit', 'ba', 6125401);  
insert into cust values('a05', 'sreedharan', 'Ravi', 'ra', NULL);  
insert into cust values('a06', 'NULL', 'Rukmini', 'gh', 5125274);

alter table invoice add FOREIGN KEY (cust\_id) references  
cust (cust\_id);

alter table invoice add FOREIGN KEY (mv\_no) references  
movie (mv\_no);

alter table cust modify fname varchar(15) NOT NULL;

alter table movie modify Title varchar(25) NOT NULL;

```
alter table movie modify Type varchar(10) NOTNULL;
alter table movie add Price float(8,2);
insert into movie values ('1', 'bloody vengeance', 'action',
    'jackie chan', 180.95);
insert into movie values ('2', 'the firm', 'thriller', 'tom
    cruise', 200.00);
insert into movie values ('3', 'pretty woman', 'romance',
    'richard gere', 150.55);
insert into movie values ('4', 'home alone', 'comedy',
    'macaulay culkin', 150.00);
insert into movie values ('5', 'the fugitive', 'thriller',
    'narrison ford', 200.00);
insert into movie values ('6', 'coma', 'suspense',
    'michael douglas', 100.00);
insert into movie values ('7', 'dracula', 'horror',
    'gary oldman', 150.25);
insert into movie values ('8', 'quick change', 'comedy',
    'bill muray', 100.00);
insert into movie values ('9', 'gone with the wind', 'drama',
    'clarke gable', 200.00);
insert into movie values ('10', 'carry on doctor', 'comedy',
    'leslie phillips', 100.00);
insert into invoice values ('i01', '4', 'a01', 1993-07-23,
    1993-07-25);
insert into invoice values ('i02', '3', 'a02', 1993-08-12, 1993-
    08-15);
insert into invoice values ('i03', '11', 'a02', 1993-08-15, 1993-
    08-18);
```

```
insert into invoice values ('i04','6','a03',1993-09-10,
                           1993-09-12);
insert into invoice values ('i05','7','a04',1993-08-05,
                           1993-08-08);
insert into invoice values ('i06','2','a06',1993-09-18,
                           1993-09-21);
insert into invoice values ('i07','9','a05',1993-07-07,
                           1993-07-10);
insert into invoice values ('i08','9','a01',1993-08-11,
                           1993-08-14);
insert into invoice values ('i09','5','a03',1993-07-06,
                           1993-07-07);
insert into invoice values ('i10','8','a06',1993-09-03,
                           1993-09-06);
select * from cust;
select * from movie;
select * from invoice;
COMMIT;
exit;
```

cust table -

cust_id	lname	fname	Area	Phone-no
a01	Bayross	Ivan	sa	6125467
a02	Saitwal	Vandana	mu	5560379
a03	Jaguste	Pramada	da	4563891
a04	Navindgi	Basu	ba	6125401
a05	Sreedharam	Ravi	ra	NULL
a06	NULL	Rukmini	gh	5125274

2

Movie table -

MV-no	Title	Type	Star	Price
1	Bloody Vengeance	Action	Jackie Chan	180.95
2	The Firm	Thriller	Tom Cruise	200.00
3	Pretty Woman	Romance	Richard Gere	150.55
4	Home Alone	Comedy	Macaulay Culkin	150.00
5	The Fugitive	Thriller	Harrison Ford	200.00
6	Coma	Suspense	Michael Douglas	100.00
7	Dracula	Horror	Gary Oldman	150.25
8	Quick Change	Comedy	Bill Murray	100.00
9	Gone with the Wind	Drama	Clark Gable	200.00
10	Carry On Doctor	Comedy	Leslie Phillips	100.00

Invoice table -

inv-no	MV-no	cust-id	issue-date	return-date
i01	4	a01	1993-07-23	1993-07-25
i02	3	a02	1993-08-12	1993-08-15
i03	1	a02	1993-09-10	1993-09-12
i04	6	a03	1993-08-05	1993-08-08
i05	7	a04	1993-09-18	1993-09-21
i06	2	a06	1993-07-07	1993-07-10
i07	9	a05	1993-08-15	1993-08-18
i08	9	a01	1993-08-11	1993-08-14
i09	5	a03	1993-07-06	1993-07-07
i10	8	a06	1993-09-03	1993-09-06

V.Puri  
24/08/23

24.08.23

4.A) Write SQL statements to retrieve these query :

1. Find out the names of all the customers.

select lname, fname from cust;

lname	fname
Bayross	Ivan
Saitwal	Vandana
Jagusti	Pramada
Navidgi	Basu
Sreedharam	Ravi
NULL	Rukmini

2. Print the entire customer table.

cust_id	lname	fname	Area	Phone_no
a01	Bayross	Ivan	sa	6125467
a02	Saitwal	Vandana	mu	5560379
a03	Jagusti	Pramada	da	4563891
a04	Navidgi	Basu	ba	6125401
a05	Sreedharam	Ravi	ra	NULL
a06	null	Rukmini	gh	5125274

3. Retrieve the list of fname and the area of all the customers. select fname, area from cust;

fname	Area
Ivan	Sa
Vandana	Mu
Pramada	Da
Basu	Ba
Ravi	Va
Rukmini	Gh

4. List the various movies type available from the movie table. select type from movie;

type
Action
comedy
thriller
romance
comedy
thriller
suspense
horror
comedy
drama

5. Print the info of invoice table in the following format for all records.

A) The Invoice No. of cust Id. {cust\_id} is {inv\_no} & Movie No. is {mv\_no}.

select concat('The Invoice No. of cust Id ', cust\_id, ' and Movie No. ', mv\_no) from 'invoice';

B. {cust\_id} has taken Movie No. {mv\_no} on {issue-date} and will return on {return-date}.

select concat(cust\_id, 'has taken Movie No.', mv\_no, 'on', issue\_date, 'and will return on', return\_date) from invoice;

6. change the telephone number of pramada to 466389.

update cust set phone\_no = 466389 where fname = 'Pramada';

select fname, phone\_no from cust;

fname	Phone-no
Iram	6125467
Vandana	5560176
Pramada	466389
Basu	6125401
Ravi	NULL
Rukmini	5125274

7. change the issue-date of cust\_id 'a01' to 24/07/93.  
update invoice set issue\_date = '1993-07-24' where cust\_id = 'a01';

8. change the price of 'gone with the wind' to Rs250.00  
update movie set price = 250.00 where title = 'gone with the wind';

9. Delete the record with invoice number 'i08' from the 'invoice' table.

delete from invoice where inv-no = 'i08';

select \* from invoice;

inv-no	mv-no	cust-id	issue-date	return-date
i01	4	a01	1993-07-24	1993-07-25
i10	8	a06	1993-08-03	1993-09-06
i02	3	a02	1993-08-12	1993-08-15
i03	1	a02	1993-08-15	1993-08-18
i04	6	a03	1993-09-10	1993-09-12
i05	7	a04	1993-08-05	1993-08-08
i06	2	a06	1993-09-18	1993-09-21
i07	9	a05	1993-07-07	1993-07-10
i09	5	a03	1993-07-06	1993-07-07

10. Delete all the records having return date before 10th July '93.

delete from invoice where return-date < '1993-07-10';

select \* from invoice;

inv-no	mv-no	cust-id	issue-date	return-date
i10	8	a06	1993-09-03	1993-09-06
i02	3	a02	1993-08-12	1993-08-15
i03	1	a02	1993-08-15	1993-08-18
i04	6	a03	1993-09-10	1993-09-12
i05	7	a04	1993-08-05	1993-08-08
i06	2	a06	1993-09-18	1993-09-21
i07	9	a05	1993-07-07	1993-07-10

31.08.23

4. B) 11. change the area of cust\_id 'a05' to 'vs'.  
update cust set area='vs' where cust\_id='a05'.
12. change the return\_date of invoice number 'i08' to  
16-08-93.  
update invoice set return\_date = <1993-08-16> where  
inv\_no = 'i08';
13. Find the names of all customers having 'a' as  
the second letter in their names.  
select fname from cust where fname like  
'\_a%' ;
14. Find the lnames of all customers that begin with  
'is' or 'ij'.  
select lname from cust where lname like 'i%' or  
lname like 's%' ;
15. Find the list of all customers who stay in area 'da' or  
area 'mu' or area 'gh'.  
select \* from cust where area = 'da' or area = 'mu' or  
area = 'gh' ;
16. Find out the customers who stay in an area  
whose second letter is 'a'.  
select fname, lname from cust where area like  
'\_a%' ;
17. Print the list of employees whose phone numbers  
are greater than the value 5550000.

select \* from cust where phone-no = 555 0000;

18. Print the info from invoice table of customers who have been issued movies in the month of september.

select \* from invoice where extract(month of issue-date) = 9;

19. Display the invoice table info for cust-id 'a01' & 'a02'.

select \* from invoice where cust-id = 'a01' and cust-id = 'a02';

inv-no	mv-no	cust-id	issue-date	return-date
i01	4	a01	1993-07-23	1993-07-25
i02	3	a02	1993-08-12	1993-08-15
i03	1	a02	1993-09-10	1993-09-12
i08	9	a01	1993-08-11	1993-08-14

20. Find the movies of type 'action' and 'comedy'.  
select \* from movie where type = 'action' or type = 'comedy';

mv-no	title	type	star	price
1	bloody vengeance	action	Jackie chan	180.95
4	home alone	comedy	Macaulay Culkin	150.00
8	quick change	comedy	Bill Murray	100.00
10	carry on doctor	comedy	Leslie Phillips	100.00

07.09.23

- 4.c) 21. Find the movies whose price is greater than 150 and less than or equal to 200.

select \* from movie where price > 150 and  
price <= 200;

mv_no	title	type	star	Price
1	bloody revenge	action	jackie chan	180.95
2	the firm	thriller	tom cruise	200.00
3	pretty woman	romance	richard gene	150.55
5	the fugitive	thriller	harrison ford	200.00
7	dracula	horror	gary oldman	150.25
9	go with the wind	drama	clarke gable	200.00

22. Find the movies that cost more than 150 and also find the new cost as original cost + 15.  
select title, price, price + 15 as ori\_price from movie where price > 150;

title	Price	ori_price
bloody revenge	180.95	214.25
the firm	200.00	3000.00
pretty woman	150.55	2258.25
the fugitive	200.00	3000.00
dracula	150.25	2253.75
go with the wind	200.00	3000.00

23. Rename the new column in the above query as new-price.

select distinct pri-price as new-price from movie;

new-price

2714.25

3000.00

2258.25

3000.00

2253.75

3000.00

24. List the movies in sorted order of their titles  
select \* from movie order by title;

25. Print the names & types of all the movie except horror movie.

select title, type from movie where type

<> 'horror';

26. Divide the cost of movie 'home alone' by difference between its price and 100.

select price / (price - 100) as result from movie where title = 'home alone';

result

3.000000

27. List the names, areas and cust-id of customers without phone numbers.

select lname, fname, area, cust\_id from cust where phone\_no is NULL or phone\_no = '';

28. List the names of customers without lname.  
select fname from cust where lname is NULL  
or lname = ' ';
29. List the mv-no, title, type of movies whose star begin with letter 'm'.  
select mv-no, title, type from movie where star like 'm%';

mv-no	title	type
4	home alone	comedy
6	coma	suspense

30. List the mv-no & inv-no of customers having inv-no less than 'i05' from the invoice transaction table.  
select mv-no, inv-no from invoice where inv-no < 'i05';