

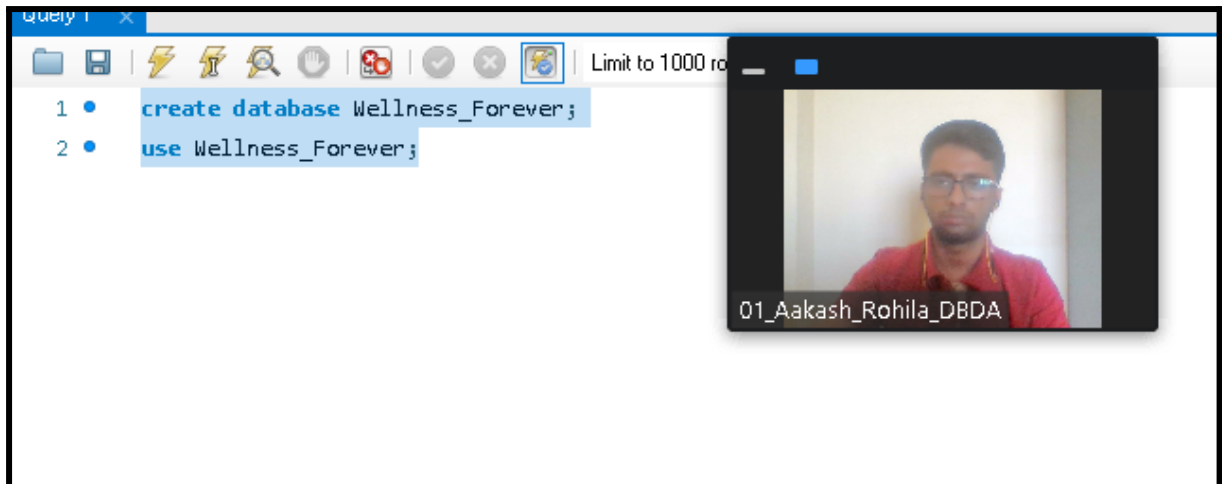
MySQL

Create Database system for Medical Store(Like Wellness Forever)

#####

```
create database Wellness_Forever;  
use Wellness_Forever;
```

#####



1. Create minimum 5 tables(5)

#####

```
create table customer(
```

```
customer_id varchar(20),  
customer_name varchar(20),  
customer_health varchar(20)
```

```
);
```

```
create table medicine(
```

```
medicine_id varchar(20),  
medicine_name varchar(20),  
medicine_cost int
```

```
);
```

```
create table orders(
```

```
order_id varchar(20),  
order_name varchar(20),  
order_status varchar(20)
```

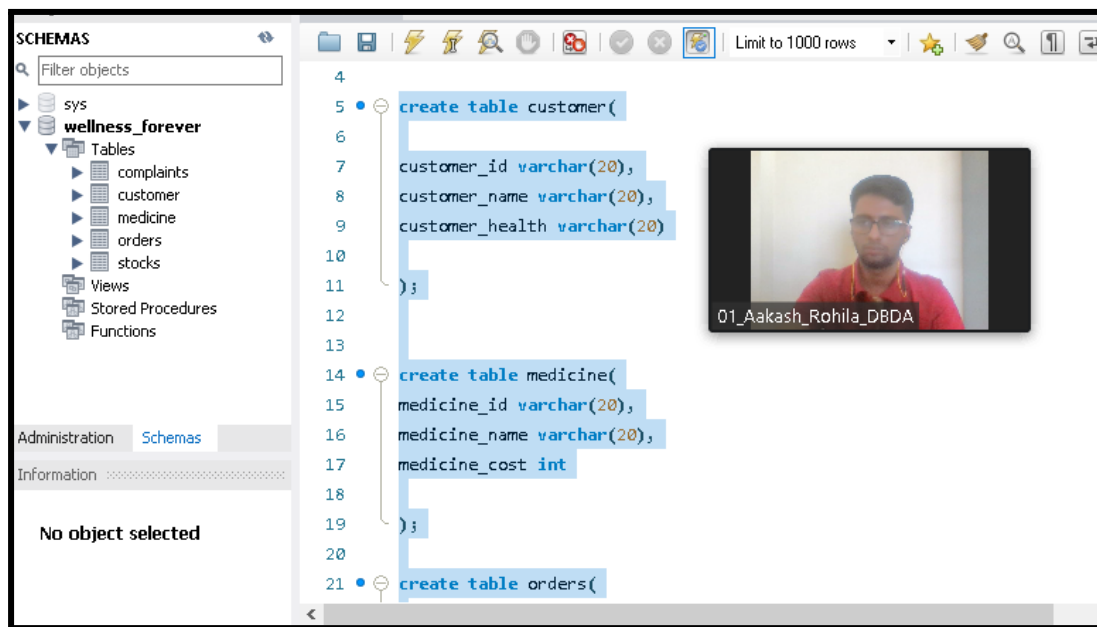
```
);
```

```
create table complaints(
```

```
complaint_id varchar(20),  
customer_id varchar(20),  
complaint_issue varchar(20)
```

```
);
```

#####



2. Insert minimum three records in each of the table(2)

#####

```
insert into customer
values
("C101","Aakash","Good"),
("C102","Aman","Great"),
("C103","Ajay","Good");
```

```
insert into medicine
values
("M101","Crosin",5),
("M102","Vicks",40),
("M103","Saridon",10);
```

```
insert into orders
values
("O101","M101","Shipped"),
("O102","M103","Waiting"),
("O103","M102","Shipped");
```

```
insert into stocks
values
("M101","Crosin",100),
("M102","Vicks",10),
("M103","Saridon",50);
```

```
insert into complaints
values
("T101","C101","Expired"),
("T102","C102","Active"),
("T103","C102","Expired");
```

#####

3. Each table must have Primary Key(2)

#####

```
alter table complaints  
add primary key(complaint_id);
```

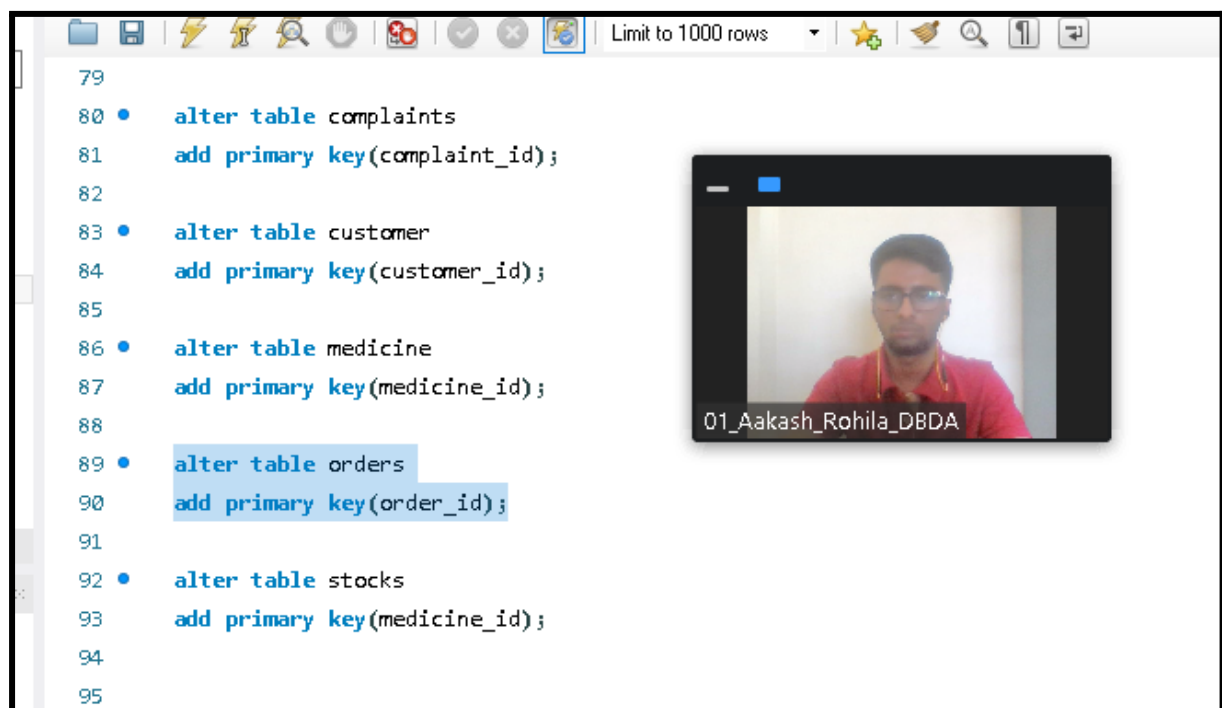
```
alter table customer  
add primary key(customer_id);
```

```
alter table medicine  
add primary key(medicine_id);
```

```
alter table orders  
add primary key(order_id);
```

```
alter table stocks  
add primary key(medicine_id);
```

#####



4. Connect atleast 3 tables using Foreign Key(2)

#####

```
alter table customer
```

```
add medicine_id varchar(20);
```

```
update customer set medicine_id = "M101" where customer_id = "C101";
```

```
update customer set medicine_id = "M103" where customer_id = "C102";
```

```
update customer set medicine_id = "M102" where customer_id = "C103";
```

```
alter table customer
```

```
add foreign key (medicine_id) references medicine(medicine_id);
```

```
alter table complaints
```

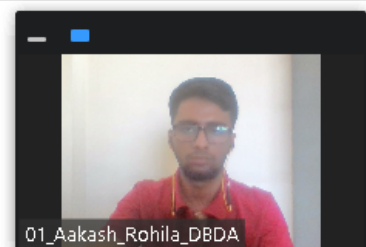
```
add foreign key (customer_id) references customer(customer_id);
```

```
alter table orders
```

```
add foreign key (order_name) references Medicine(medicine_id);
```

#####

```
7  add medicine_id varchar(20);
8
9  •  update customer set medicine_id = "M101" where customer_id = "C101";
10 •  update customer set medicine_id = "M103" where customer_id = "C102";
11 •  update customer set medicine_id = "M102" where customer_id = "C103";
12
13 •  alter table customer
14      add foreign key (medicine_id) references medicine(medicine_id);
15
16
17
18 •  alter table complaints
19      add foreign key (customer_id) references customer(customer_id);
20
21
22 •  alter table orders
23      add foreign key (order_name) references Medicine(medicine_id);
24
```



5. Write an appropriate command to Use of Where with OR clause (2)

#####

```
select *  
from customer  
where customer_health = 'Good' or medicine_id = 'M101';
```

#####

1 • SELECT * FROM wellness_forever.customer;
2
3 • select *
4 from customer
5 where customer_health = 'Good' or medicine_id = 'M101';

01_Aakash_Rohila_DBDA

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

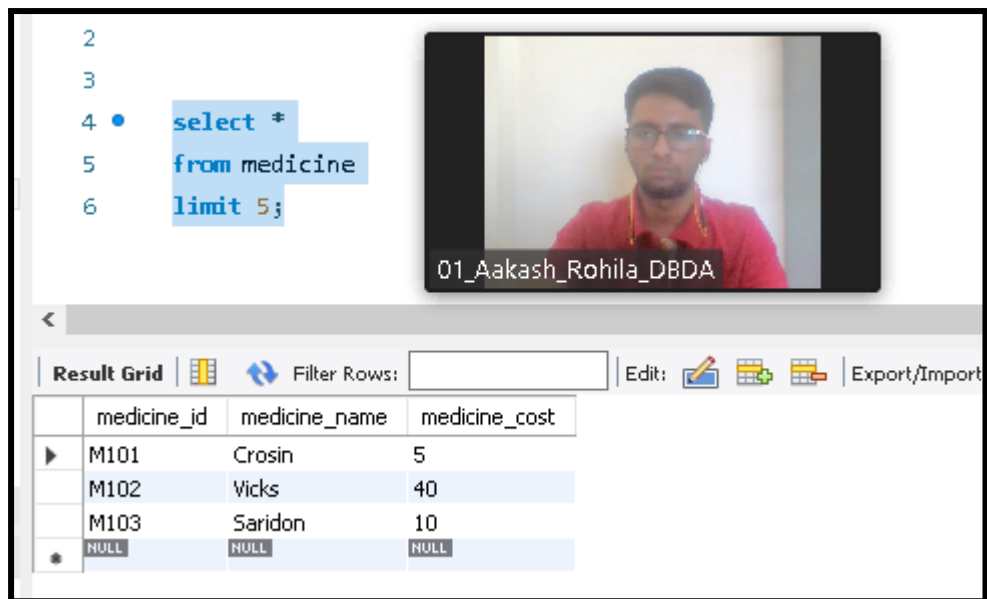
customer_id	customer_name	customer_health	medicine_id
C101	Aakash	Good	M101
C103	Ajay	Good	M102
NULL	NULL	NULL	NULL

6. Write a command to show first five records from any one table (2)

#####

```
select *  
from medicine  
limit 5;
```

#####



```

2
3
4 • select *
5   from medicine
6   limit 5;

```

01_Aakash_Rohila_DBDA

Result Grid | Filter Rows: | Edit: | Export/Import

	medicine_id	medicine_name	medicine_cost
▶	M101	Crosin	5
	M102	Vicks	40
	M103	Saridon	10
*	NULL	NULL	NULL

7. Use of JOIN (left or right, any one types) (4)

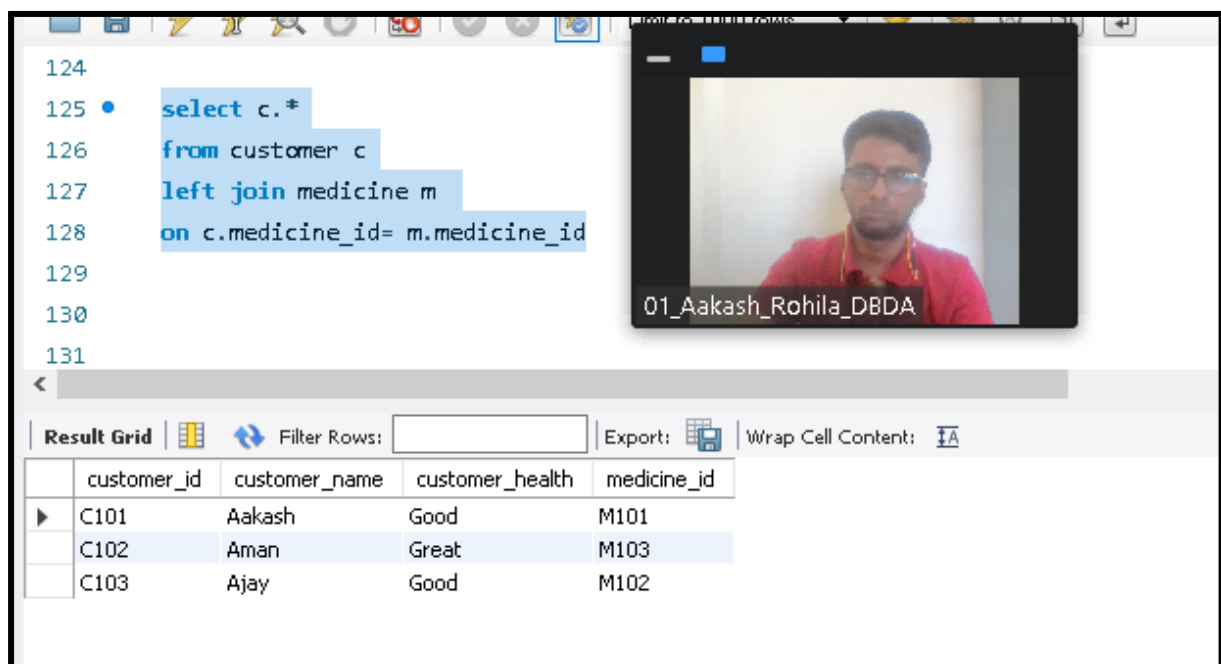
#####

```

select c.*
from customer c
left join medicine m
on c.medicine_id= m.medicine_id

```

#####



```

124
125 • select c.*
126   from customer c
127   left join medicine m
128   on c.medicine_id= m.medicine_id
129
130
131

```

01_Aakash_Rohila_DBDA

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	customer_id	customer_name	customer_health	medicine_id
▶	C101	Aakash	Good	M101
	C102	Aman	Great	M103
	C103	Ajay	Good	M102

8. Create at least one procedure (with one INPUT and one OUTPUT parameter) based on the created tables with appropriate utility (4)

“Below procedure will fetch all the order_id of those orders whose status has been passed as a parameter, it can be useful when you want to know the order_id of those orders which have not been shipped.”

#####

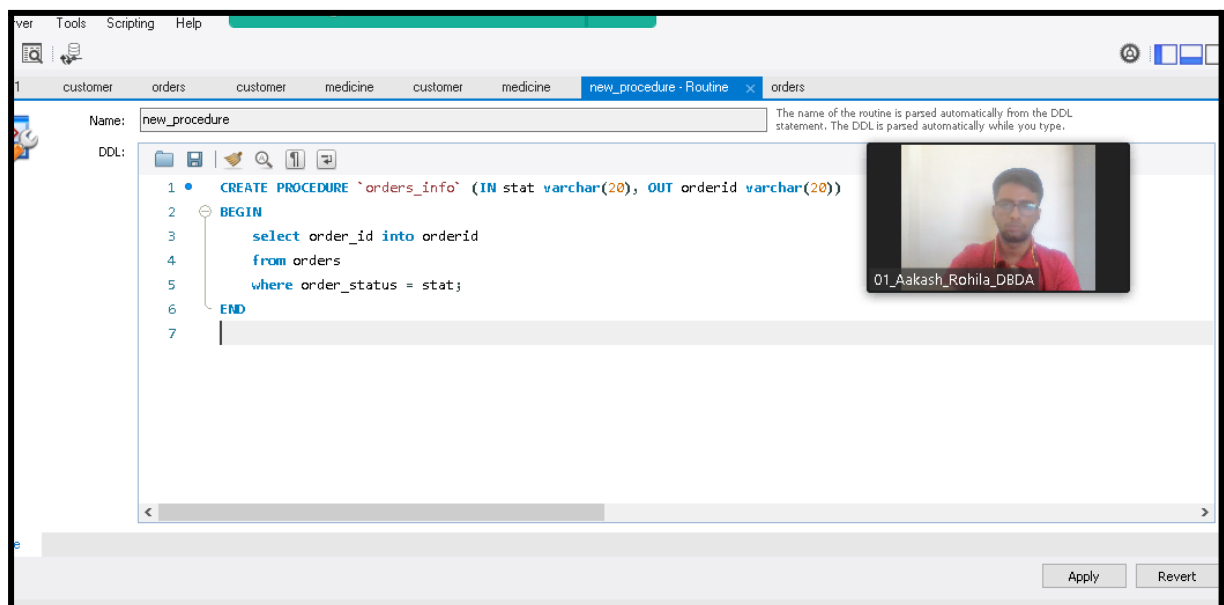
```
CREATE PROCEDURE `orders_info` (IN stat varchar(20), OUT orderid varchar(20))
BEGIN
    select order_id into orderid
    from orders
    where order_status = stat;
END
```

#####

call orders_info('waiting',@orderid);

select @orderid

#####



The screenshot shows a SQL IDE interface. On the left, a query window displays the following SQL code:

```

131
132 • call orders_info('waiting',@orderid);
133
134 • select @orderid
135
136
137

```

Below the query window, a 'Result Grid' is visible, showing a single row with the value 'O102' under the column '@orderid'. To the right of the query window, there is a video overlay of a person with glasses and a red shirt, identified by the text '01_Aakash_Rohila_DBDA'.

9. Create a ONE trigger of BEFORE UPDATE ON any one of the table(4)

```

#####
delimiter $

```

```

create trigger update_complaint_issue before update
on medicine for each row
BEGIN

```

```

insert into medicine_record(m_id , m_name , m_cost)
values
(old.medicine_id , old.medicine_name , old.medicine_cost);

```

```

END $

```

```

delimiter ;

```

```

update medicine set medicine_id = 'M101' , medicine_name = 'Dolo 650', medicine_cost = 2
where medicine_id = 'M101'

```

```

#####

```

```

9
10 delimiter $
11
12 • create trigger update_complaint_issue before update
13 on medicine for each row
14 BEGIN
15
16 insert into medicine_record(m_id , m_name , m_cost)
17 values
18 (old.medicine_id , old.medicine_name , old.medicine_cost);
19
20 END $
21
22 delimiter ;
23
24
25

```

01_Aakash_Rohila_DBDA

Output

#	Time	Action	Message	Duration / Fetch
42	16:02:08	SELECT * FROM wellness_forever.branch LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000
43	16:13:08	create table medicine_record(medicine_id varchar(20) primary key, medicine_name ...	0 row(s) affected	0.032 sec
44	16:14:49	DROP TABLE `wellness_forever`.`medicine_record`	0 row(s) affected	0.047 sec
45	16:14:52	create table medicine_record(m_id varchar(20) primary key, m_name varchar(20), m...	0 row(s) affected	0.031 sec
46	16:15:53	create trigger update_complaint_issue before update on medicine for each row BEGI...	Error Code: 1064. You have an error in your SQL syntax; check the manual that corres...	0.000 sec
47	16:16:54	create trigger update_complaint_issue before update on medicine for each row BEGI...	0 row(s) affected	0.031 sec

You are screen sharing

Stop Share

```

• update medicine set medicine_id = 'M101' , medicine_name = 'Dolo 650', medicine_cost = 2
where medicine_id = 'M101'

```

01_Aakash_Rohila_DBDA

Action Output

#	Time	Action	Message	Duration / Fetch
47	16:16:54	create trigger update_complaint_issue before update on medicine for each row BEGI...	0 row(s) affected	0.031 sec
48	16:18:01	SELECT * FROM wellness_forever.medicine LIMIT 0, 1000	3 row(s) returned	0.015 sec / 0.000

“As one can see when I update my medicine data my old record is recorded automatically in my medicine_record table with the help of trigger command”

```

1 • SELECT * FROM wellness_forever.medicine_record;

```

01_Aakash_Rohila_DBDA

Result Grid

	m_id	m_name	m_cost
▶	M101	Crosin	5
*	NULL	NULL	NULL

MongoDB

Create Database in MongoDB for Medical Store System. - 7 Marks

#####

```
use medicine_store;
db;
```

#####

```
> db
medicine_store
> _
```

1. Write the command for creating collection & for inserting documents in collections (minimum 3 documents) (3)

#####

```
db.medicine.insert([{"name": "Crosin", medicine_cost : 10}, {"name": "Vicks",medicine_cost : 40}, {"name": "Saridon" , medicine_cost : 5}]);
```

#####

```
Command Prompt - MongoDB
> db
medicine_store
> db.medicine.insert([{"name": "Crosin", medicine_cost : 10}, {"name": "Vicks",medicine_cost : 40}, {"name": "Saridon" , medicine_cost : 5}]);
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 3,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
> db.medicine.find().pretty()
{
  "_id" : ObjectId("635f9ec235b54cd618b19440"),
  "name" : "Crosin",
  "medicine_cost" : 10
}
{
  "_id" : ObjectId("635f9ec235b54cd618b19441"),
  "name" : "Vicks",
  "medicine_cost" : 40
}
{
  "_id" : ObjectId("635f9ec235b54cd618b19442"),
  "name" : "Saridon",
  "medicine_cost" : 5
}
>
```



2. For use of AND clause within function (2)

#####

```
db.medicines.find({$and : [{name : "Crosin"},{medicine_cost : {$lt : 15}}]},{_id : 0});
```

#####

```
}
> db.medicines.find({$and : [{name : "Crosin"},{medicine_cost : {$lt : 15}}]},{_id : 0})
{ "name" : "Crosin", "medicine_cost" : 10 }
>
```

3. For sorting the result in descending order using some field (2)

#####

```
db.medicines.find({}, {_id : 0}).sort({medicine_cost : -1}).pretty();
```

#####

```
> db.medicines.find({}, {_id : 0}).sort({medicine_cost : -1}).pretty()
{ "name" : "Vicks", "medicine_cost" : 40 }
{ "name" : "Crosin", "medicine_cost" : 10 }
{ "name" : "Saridon", "medicine_cost" : 5 }
>
```

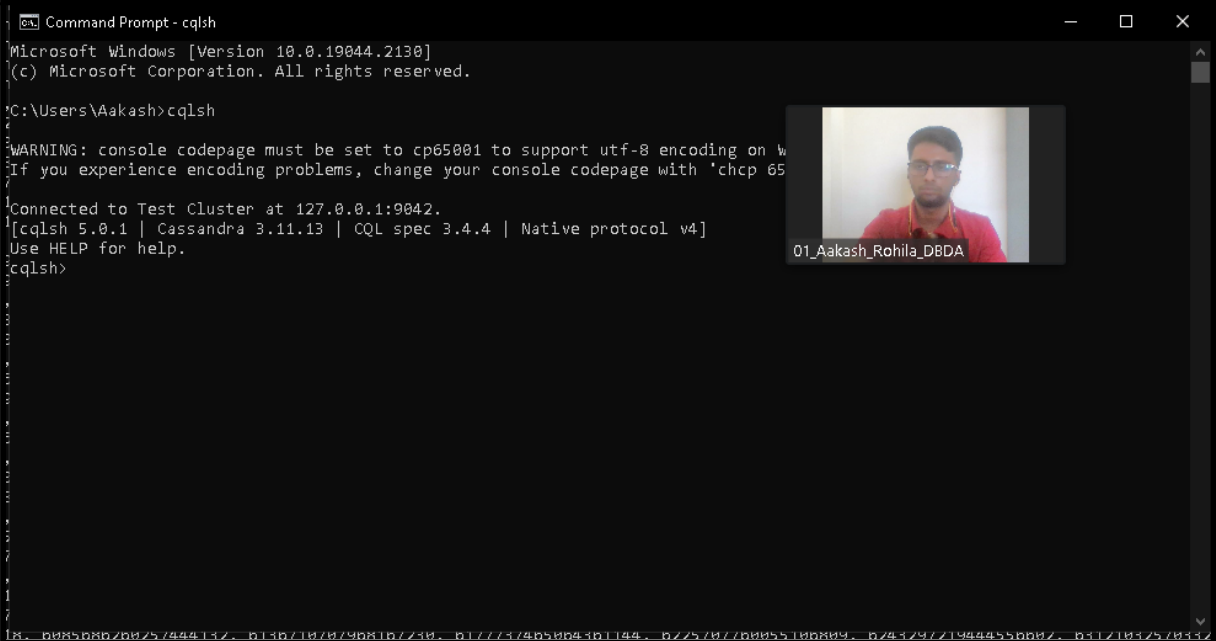
Cassandra

Write the command for following - 6 Marks

First start cassandra server with the following command in cmd:

>cassandra

Then start another cmd command and write 'cqlsh' to start working with cassandra commands.



```
Command Prompt - cqlsh
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Aakash>cqlsh

WARNING: console codepage must be set to cp65001 to support utf-8 encoding on w
If you experience encoding problems, change your console codepage with 'chcp 65

Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.11.13 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh>
```

1. Create a Keyspace with all required parameters (2)

#####

create keyspace medicine with replication = {'class': 'SimpleStrategy' , 'replication_factor': 1};

desc keyspaces;

use medicine;

#####

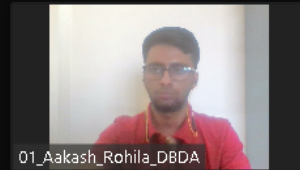
```

cqlsh> create keyspace medicine with replication = {'class': 'SimpleStrategy', 'replication_factor': 1};
cqlsh> show keyspaces
Improper show command.
cqlsh> desc
Improper desc command.
cqlsh> desc keyspaces

system_schema  system  medicine      employee
system_auth    library system_distributed system_traces

cqlsh>

```



2.Create a table 'Branch' with appropriate columns.(2)

#####

```

create table branch(
product_id text primary key,
products text,
product_cost int
);

```

```

desc tables;

```

#####

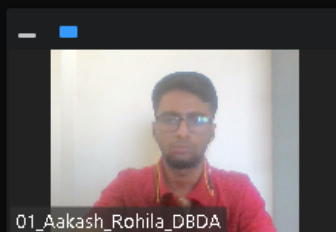
```

cqlsh:medicine> create table branch( product_id text primary key, products text, product_cost int );
cqlsh:medicine> desc tables

branch

cqlsh:medicine>

```



3. Insert values like Medicines, Bisucits, FruitJuices and SoftDrinks in one of the column 'Products' of table 'Branch' (2)

#####

```
insert into branch(product_id,product_cost,products)
values
('P101',50,'Medicines');
```

```
insert into branch(product_id,product_cost,products)
values
('P102',5,'Bisucits');
```

```
insert into branch(product_id,product_cost,products)
values
('P103',90,'FruitJuices');
```

```
insert into branch(product_id,product_cost,products)
values
('P104',40,'SoftDrinks');
```

#####

```
sqlsh:medicine> insert into branch(product_id,product_cost,products)
... values
... ('P102',5,'Bisucits');
sqlsh:medicine> insert into branch(product_id,product_cost,products)
... values
... ('P103',90,'FruitJuices');
sqlsh:medicine>
sqlsh:medicine> insert into branch(product_id,product_cost,products)
... values
... ('P104',40,'SoftDrinks');
sqlsh:medicine> select * from branch;
```

product_id	product_cost	products
P103	90	FruitJuices
P101	50	Medicines
P104	40	SoftDrinks
P102	5	Bisucits

(4 rows)

```
sqlsh:medicine>
```

