

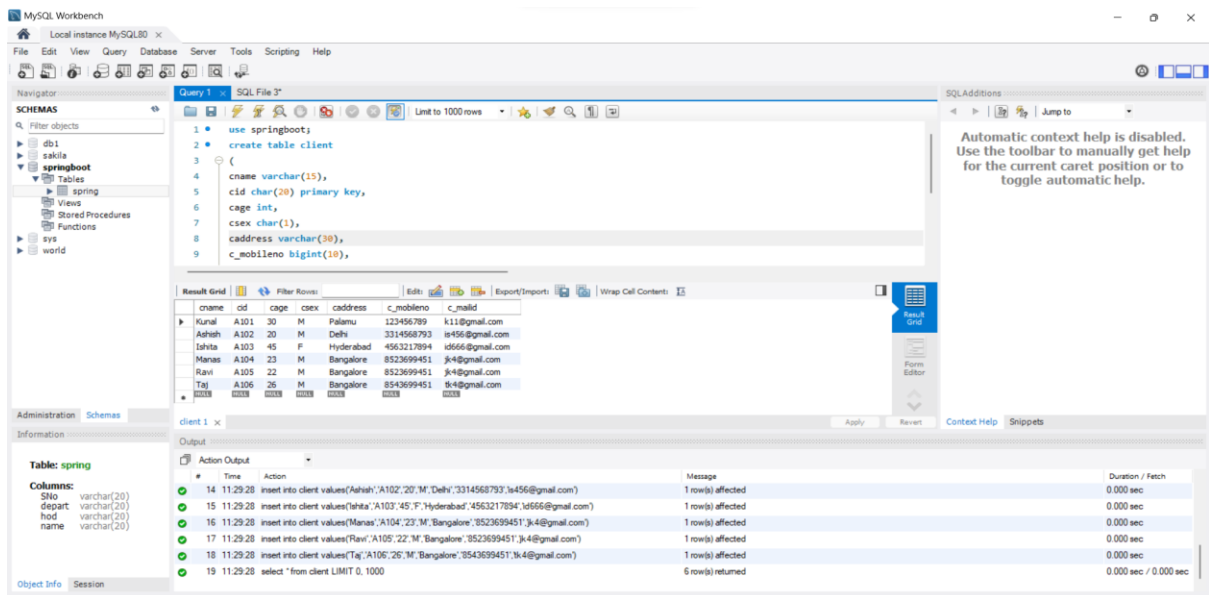
Name – Harshit Jaiswal

Reg no – 20BCE2157

Modern Application Development (Java Spring Boot)

Q1. Create , update , delete commands in mysql.

```
use springboot;
create table client
(
cname varchar(15),
cid char(20) primary key,
cage int,
csex char(1),
caddress varchar(30),
c_mobilen no bigint(10),
c_mailid varchar(25)
);
insert into client values( 'Kunal','A101','30','M','Palamu','123456789','k11@gmail.com');
insert into client values('Ashish','A102','20','M','Delhi','3314568793','is456@gmail.com');
insert into client values('Ishita','A103','45','F','Hyderabad','4563217894','id666@gmail.com');
insert into client values('Manas','A104','23','M','Bangalore','8523699451','jk4@gmail.com');
insert into client values('Ravi','A105','22','M','Bangalore','8523699451','jk4@gmail.com');
insert into client values('Taj','A106','26','M','Bangalore','8543699451','tk4@gmail.com');
select * from client;
```

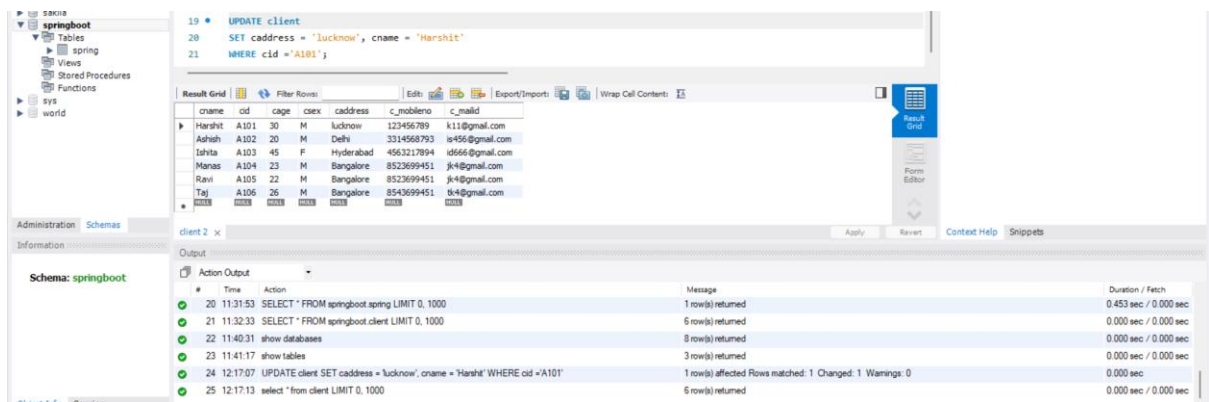


UPDATE client

SET address = 'lucknow', cname = 'Harshit'

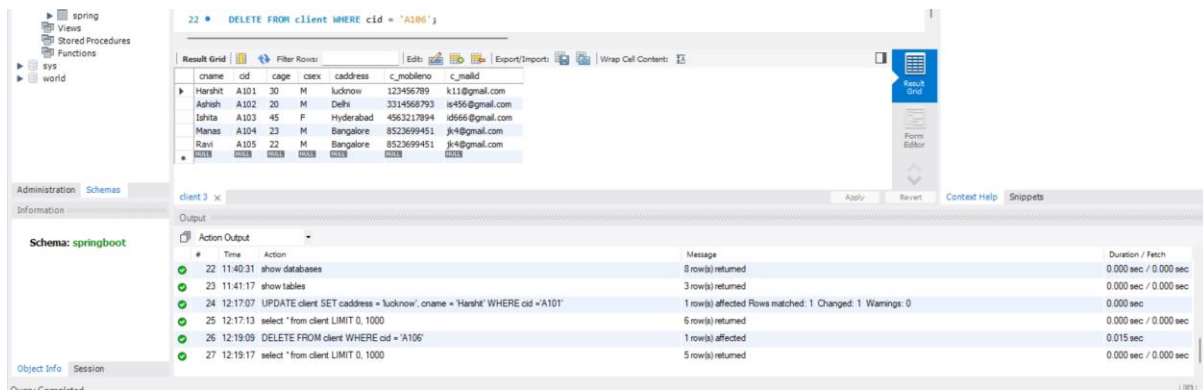
WHERE cid='A101';

select * from client;



DELETE FROM client WHERE cid = 'A106';

select * from client;



Q2. Create tables and perform joins in mysql

create table customer

(

cname varchar(15),

cid char(20) primary key,

cage int,

csex char(1),

caddress varchar(30),

c_mobilen bigint(10),

c_mailid varchar(25)

);

insert into customer values('Kunal','A101','30','M','Palamu','123456789','k11@gmail.com');

insert into customer values('Ashish','A102','20','M','Delhi','3314568793','is456@gmail.com');

insert into customer values('Ishita','A103','45','F','Hyderabad','4563217894','id666@gmail.com');

insert into customer values('Manas','A104','23','M','Bangalore','8523699451','jk4@gmail.com');

insert into customer values('Ravi','A105','22','M','Bangalore','8523699451','jk4@gmail.com');

```

insert into customer values('Taj','A106','26','M','Bangalore','8543699451','tk4@gmail.com');
insert into customer values('Hemanshi','A107','27','F','Mangalore','8943699451','hk5@gmail.com');
insert into customer values('raj','A108','27','F','Mangalore','9943699451','tk5@gmail.com');
insert into customer values('Jenny','A109','27','F','Mangalore','9943699451','jkk5@gmail.com');
insert into customer values('Pina','A110','27','F','Mangalore','9943699451','jkk5@gmail.com');

create table bank
(
  bname varchar(15),
  bid char(20) primary key,
  branch varchar(15),
  cid char(10),
  location varchar(20),
  assets int,
  foreign key (cid) references customer(cid) );

insert into bank values('Indian Bank','I101','Hyderabad','A101','Hyderabad',45632178);
insert into bank values('ICICI Bank','I102','Hyderabad','A102','Hyderabad',456982);
insert into bank values('SBI','S103','Bangalore','A103','Bangalore',456328);
insert into bank values('SBI','S104','Bangalore','A104','Vellore',896523);
insert into bank values('SBI','S105','Bangalore','A105','Bangalore',886523);
insert into bank values('SBI','S106','Bangalore','A106','Bangalore',886523);
insert into bank values('SBI','S107','Vellore','A106','Vellore',886523);
insert into bank values('SBI','S108','Vellore','A108','Vellore',886523);
insert into bank values('SBI','S109','Vellore','A108','Vellore',886523);
insert into bank values('ICICI','C102','Hyderabad','A108','Hyderabad',886523);

create table account
(
  ano char(15) primary key,
  cid char(10),
  branch varchar(15),
  a_amount int,
  acc_type varchar(15),

```

```

opening_date date,
foreign key (cid) references customer(cid) );

insert into account values('7896542130','A101','Hyderabad','45632178','saving','2001-10-12');
insert into account values('8796542581','A102','Hyderabad','456982','saving','2005-08-8');
insert into account values('8523697412','A103','Bangalore','456328','current','2007-05-4');
insert into account values('1236548521','A104','Bangalore','896523','saving','2012-05-8');
insert into account values('1236548522','A105','Bangalore','996523','saving','2012-06-8');
insert into account values('1236548523','A105','Bangalore','8996523','saving','2013-06-8');
insert into account values('1236548532','A104','Bangalore','6596523','saving','2011-06-8');
insert into account values('1236548585','A109','Hyderabad','6596523','saving','2011-06-8');

create table loan (
lno char(10) primary key,
cid char(20),
l_amount int,
age int,
opening_date date,
due_date date,
loan_type varchar(15),
interest_rate decimal (4 , 1),
branch varchar(15),
foreign key (cid) references customer(cid) );

ALTER TABLE loan
ADD CONSTRAINT chk CHECK ((age >= 25 AND loan_type ='home') or (age < 25 and loan_type =
'education'));

insert into loan values('1012','A101','36000000','30','2010-2-5','2025-2-5','home',0.9,'Hyderabad');
insert into loan values('1013','A102','90000000','20','2015-3-5','2025-5-
6','education',0.7,'Hyderabad');

insert into loan values('1014','A103','1700000','45','2018-5-3','2028-8-9','home','0.9','Bangalore');
insert into loan values('1015','A104','500000','23','2020-7-8','2030-9-8','education',0.7,'Bangalore');
insert into loan values('1016','A104','800000','23','2020-7-8','2030-9-8','education',0.7,'Mangalore');
insert into loan values('1017','A105','900000','22','2020-7-8','2030-9-8','education',0.7,'Bangalore');
insert into loan values('1018','A105','700000','22','2020-8-8','2030-8-8','education',0.7,'Mangalore');

```

```

insert into loan values('1019','A106','800000','26','2020-8-8','2030-8-8','home',0.7,'Vellore');
insert into loan values('1020','A106','700000','26','2020-8-8','2030-8-8','home',0.7,'Nellore');
insert into loan values('1021','A107','700000','26','2020-8-8','2030-8-8','home',0.7,'Vellore');
insert into loan values('1011','A108','700000','26','2020-8-8','2030-8-8','home',0.7,'Hyderabad');
insert into loan values('1025','A109','500000','26','2020-8-8','2030-8-8','home',0.7,'Hyderabad');
select * from customer;

select * from loan;

select * from account;

select * from bank;

```

Customer table

The screenshot shows the MySQL Workbench interface. The SQL editor contains several INSERT statements for the 'loan' table and three SELECT statements for 'customer', 'loan', and 'account'. The 'Result Grid' displays the data from the 'customer' table.

name	cid	cage	csex	caddress	c_mobleno	c_mailid
Kunal	A101	30	M	Palamu	122456789	k11@gmail.com
Adhish	A102	20	M	Delli	331456789	ad456@gmail.com
Ishta	A103	45	F	Hyderabad	4563217894	is456@gmail.com
Manas	A104	23	M	Bangalore	8523699451	jk4@gmail.com
Ravi	A105	22	M	Bangalore	8523699451	jk4@gmail.com
Taj	A106	26	M	Bangalore	8543699451	jk4@gmail.com
Hemanshi	A107	27	F	Mangalore	8943699451	hk5@gmail.com
Raj	A108	27	F	Mangalore	9943699451	jk5@gmail.com
Jenny	A109	27	F	Mangalore	9943699451	jk5@gmail.com
Pina	A110	27	F	Mangalore	9943699451	jk5@gmail.com

Loan Table

The screenshot shows the MySQL Workbench interface with the 'Result Grid' displaying the data from the 'loan' table.

lno	cid	l_amount	age	opening_date	due_date	loan_type	interest_rate	branch
1011	A108	700000	26	2020-08-08	2030-08-08	home	0.7	Hyderabad
1012	A101	3600000	30	2020-02-05	2025-02-05	home	0.9	Hyderabad
1013	A102	9000000	20	2015-03-05	2025-03-05	education	0.7	Hyderabad
1014	A103	1700000	45	2018-05-03	2028-05-03	home	0.9	Bangalore
1015	A104	500000	23	2020-07-08	2030-09-08	education	0.7	Bangalore
1016	A104	800000	23	2020-07-08	2030-09-08	education	0.7	Mangalore
1017	A105	900000	22	2020-07-08	2030-09-08	education	0.7	Bangalore
1018	A105	700000	22	2020-08-08	2030-08-08	education	0.7	Mangalore
1019	A106	800000	26	2020-08-08	2030-08-08	home	0.7	Vellore
1020	A106	700000	26	2020-08-08	2030-08-08	home	0.7	Nellore
1021	A107	700000	26	2020-08-08	2030-08-08	home	0.7	Vellore
1025	A109	500000	26	2020-08-08	2030-08-08	home	0.7	Hyderabad

Account Table

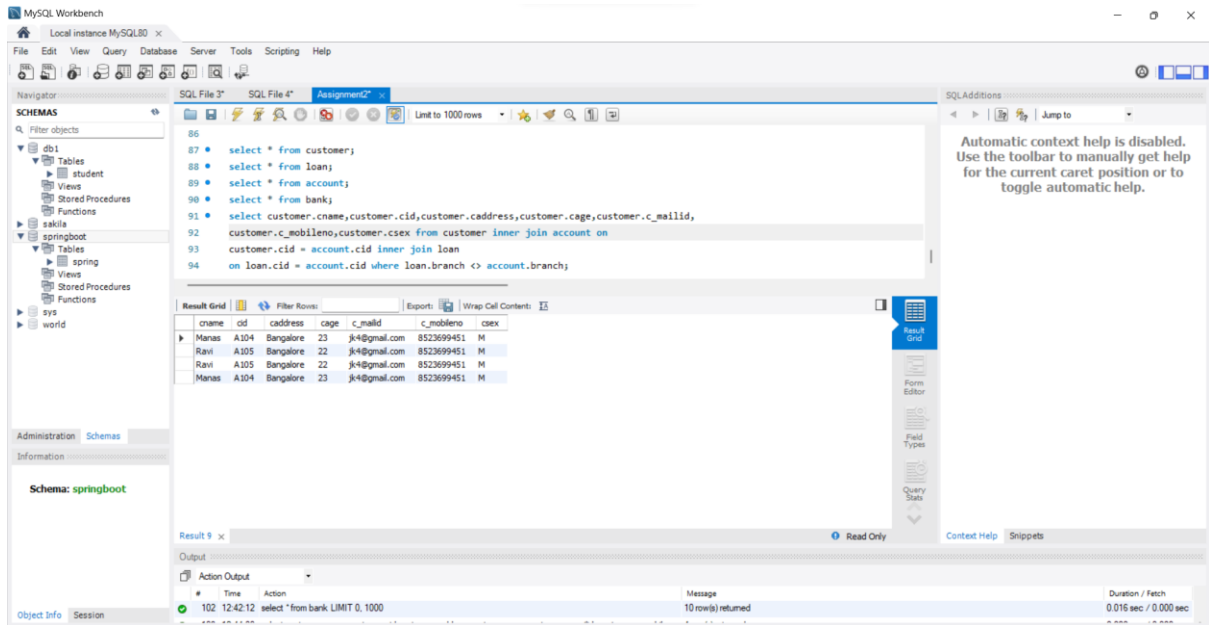
ano	cid	branch	a_amount	acc_type	opening_date
1236548521	A104	Bangalore	896523	saving	2012-05-08
1236548522	A105	Bangalore	996523	saving	2012-06-08
1236548523	A105	Bangalore	8996523	saving	2013-06-08
1236548532	A104	Bangalore	6596523	saving	2011-06-08
1236548585	A109	Hyderabad	6596523	saving	2011-06-08
7896542130	A101	Hyderabad	45632178	saving	2001-10-12
8523697412	A103	Bangalore	456328	current	2007-05-04
8796542581	A102	Hyderabad	456982	saving	2005-08-08
NULL	NULL	NULL	NULL	NULL	NULL

Bank Table

bname	bid	branch	cid	location	assets
ICICI	C102	Hyderabad	A108	Hyderabad	886523
Indian Bank	I101	Hyderabad	A101	Hyderabad	45632178
ICICI Bank	I102	Hyderabad	A102	Hyderabad	456982
SBI	S103	Bangalore	A103	Bangalore	456328
SBI	S104	Bangalore	A104	Vellore	896523
SBI	S105	Bangalore	A105	Bangalore	886523
SBI	S106	Bangalore	A106	Bangalore	886523
SBI	S107	Vellore	A106	Vellore	886523
SBI	S108	Vellore	A108	Vellore	886523
SBI	S109	Vellore	A108	Vellore	886523
NULL	NULL	NULL	NULL	NULL	NULL

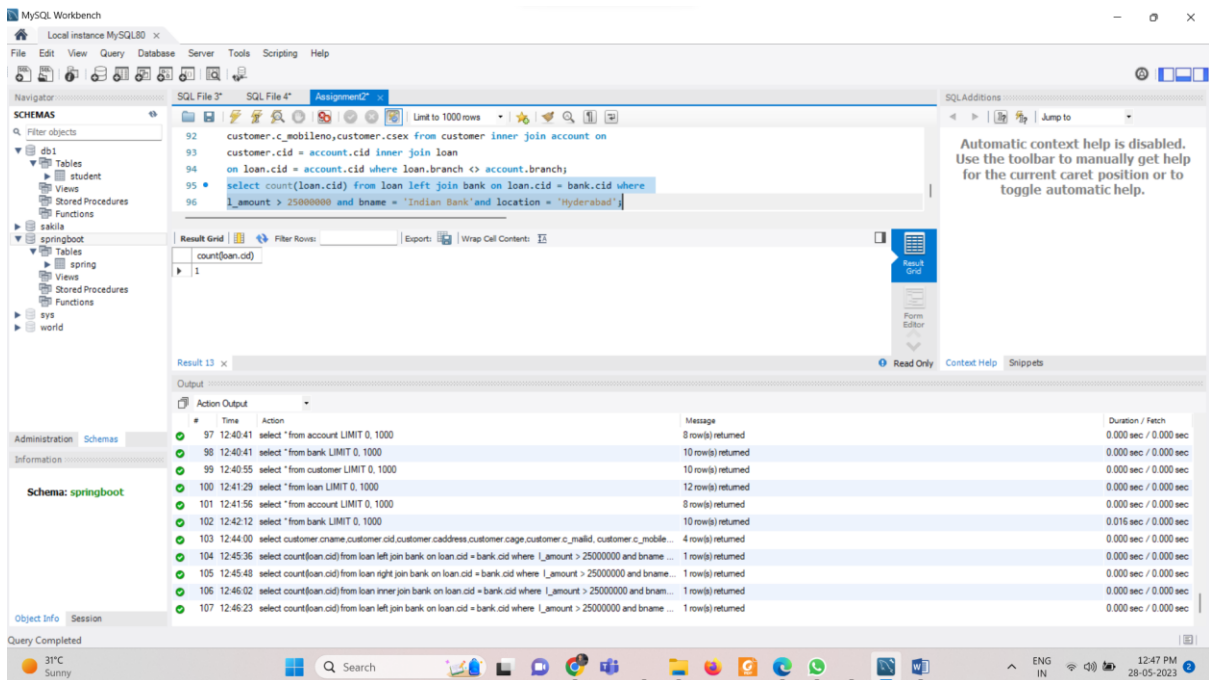
Inner join

```
select customer.cname,customer.cid,customer.caddress,customer.cage,customer.c_mailid,  
customer.c_mobileno,customer.csex from customer inner join account on  
customer.cid = account.cid inner join loan  
on loan.cid = account.cid where loan.branch <> account.branch;
```



LEFT JOIN

select count(loan.cid) from loan left join bank on loan.cid = bank.cid where
l_amount > 25000000 and bname = 'Indian Bank' and location = 'Hyderabad';

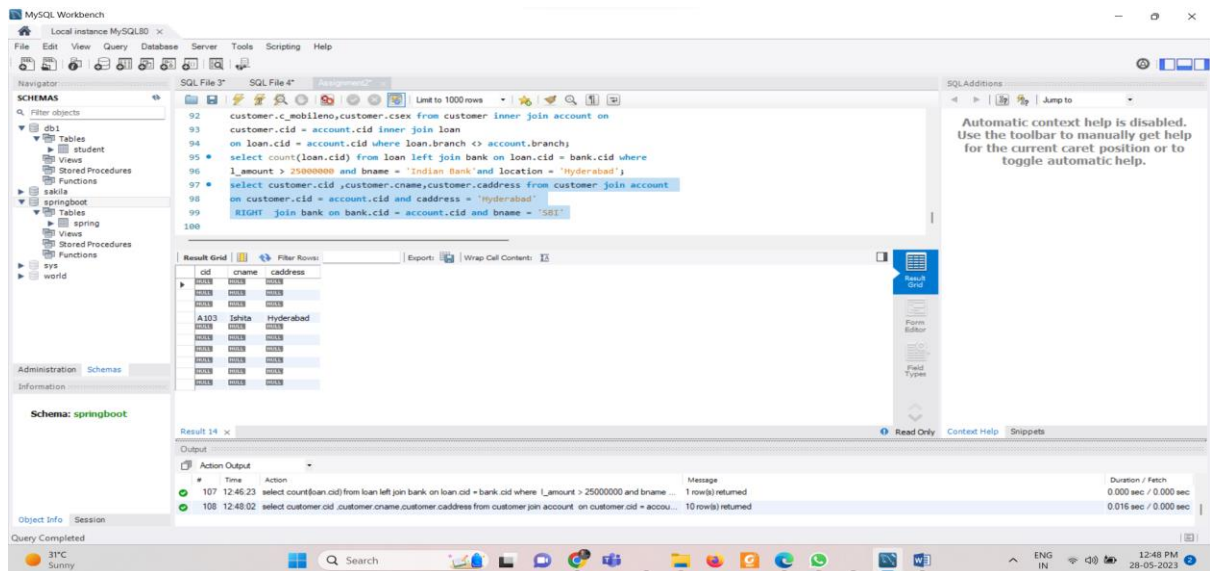


Right join

select customer.cid ,customer.cname,customer.caddress from customer join account

on customer.cid = account.cid and caddress = 'Hyderabad'

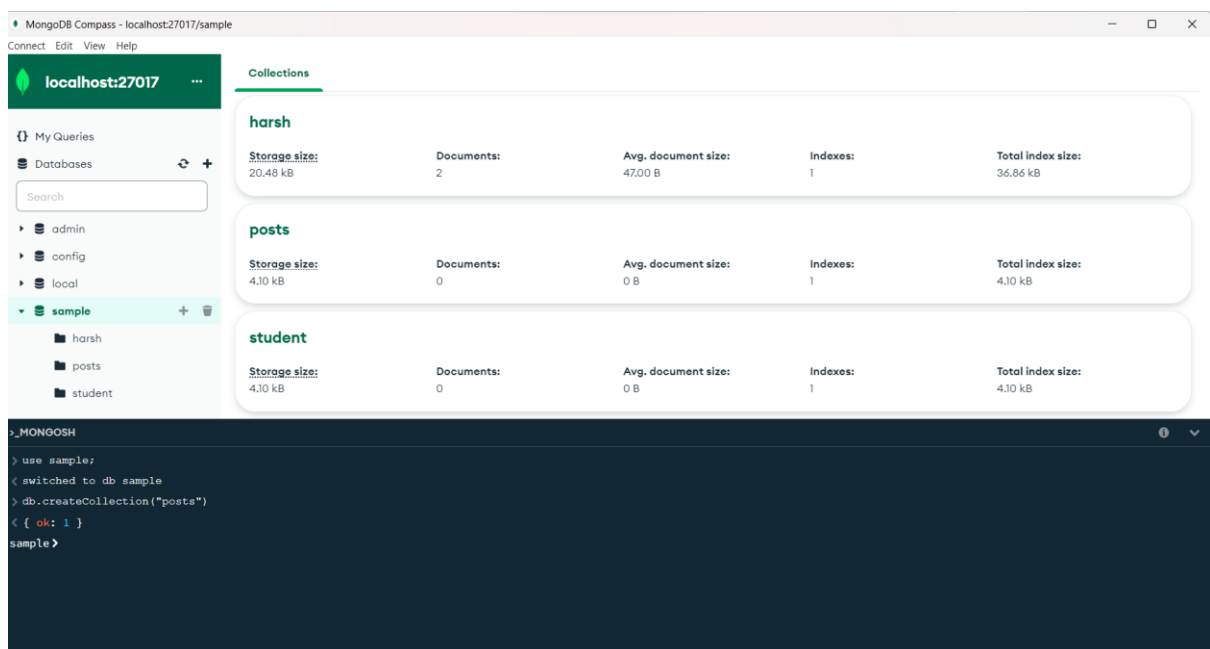
RIGHT join bank on bank.cid = account.cid and bname = 'SBI'



Q3. Create, update, delete commands in mongo

use sample;

db.createCollection("posts")



Insert

```
db.posts.insertOne({
  title: "Post Title 1",
  body: "Body of post.",
  category: "News",
  likes: 1,
  tags: ["news", "events"],
  date: Date()
})
```

```
db.posts.insertMany([
  {
    title: "Post Title 2",
    body: "Body of post.",
    category: "Event",
    likes: 2,
    tags: ["news", "events"],
    date: Date()
  },
  {
    title: "Post Title 3",
    body: "Body of post.",
    category: "Technology",
    likes: 3,
    tags: ["news", "events"],
    date: Date()
  },
  {
    title: "Post Title 4",
    body: "Body of post.",
```

```

category: "Event",

likes: 4,

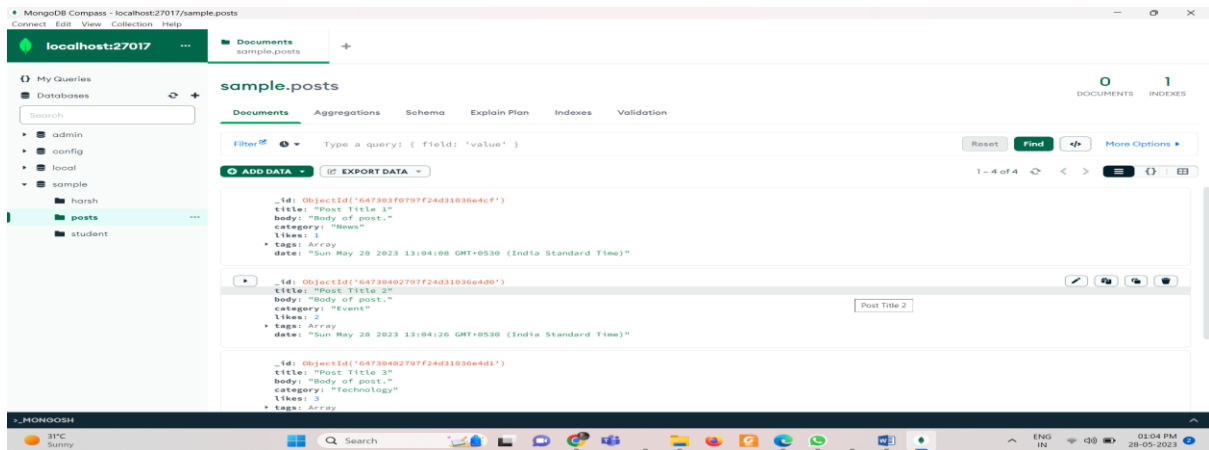
tags: ["news", "events"],

date: Date()

}

])

```



Update

```
db.posts.updateOne( { title: "Post Title 1" }, { $set: { likes: 2 } } )
```

```

db.posts.updateOne(
  { title: "Post Title 5" },
  {
    $set:
    {
      title: "Post Title 5",
      body: "Body of post.",
      category: "Event",
      likes: 5,
      tags: ["news", "events"],
      date: Date()
    }
  }
)

```

```
}
},
{ upsert: true }
)
```

The screenshot shows the MongoDB Compass interface for the 'sample.posts' collection. The left sidebar shows the database structure with 'posts' selected. The main area displays 5 documents in a list view. Each document has the following structure:

- `body`: "Body of post."
- `category`: "Event" or "Technology"
- `likes`: 2, 3, 4, or 5
- `tags`: Array
- `date`: "Sun May 28 2023 13:04:26 GMT+0530 (India Standard Time)"

The bottom status bar shows the system time as 01:10 PM on 28-05-2023.

Delete

`db.posts.deleteOne({ title: "Post Title 5" })`

The screenshot shows the MongoDB Compass interface after deleting a document. The document count is now 4. The terminal window at the bottom shows the following command and output:

```
sample> db.posts.deleteOne({ title: "Post Title 5" })
{
  acknowledged: true,
  deletedCount: 1
}
```

```
db.posts.deleteMany({ category: "Technology" })
```

The screenshot displays the MongoDB Compass interface for a local database at localhost:27017. The left sidebar shows a list of databases and collections, with 'posts' selected under the 'sample' database. The main panel shows the 'sample.posts' collection with 0 documents and 1 index. A filter bar at the top allows for querying documents. Below the filter, two document snippets are visible, each containing fields like '_id', 'title', 'body', 'category', 'likes', and 'tags'. At the bottom, a terminal window shows the execution of the command `db.posts.deleteMany({ category: "Technology" })`, which returns an acknowledgment of 1 document deleted. The Windows taskbar at the bottom shows the system clock as 01:12 PM on 28-05-2023.