राष्ट्रीय प्रौद्यौगिकी संस्थान दिल्ली
National Institute of Technology Delhi
(An autonomous Institute under the aegis of Ministry of HRD, Govt. of India)

Database management system project

(*BANK MANAGEMENT SYSTEM*)

By

Ahan Bandyopadhyay (211210008)

Gaurav Singh (211210024)

Kanha Madan (211210031)

CONTENTS

PROJECT

1 INTRODUCTION:

1.1 PROJECT OBJECTIVE

1.2 PROJECT BENEFITS

1.3 PROJECT SCOPE

- The bank management system is a set of essential tools and processes that allow banks and their credit institutions to carry out their functions.

- The components of the bank management system may differ depending on the bank, but generally, the system includes core banking to manage basic transactions, loans, mortgages, and payments accessible via ATM, mobile banking, and branches.

Database Management is the core of modern data, as handling and managing data is the key to making exponential progress in today's world.

Here, the undersigned students of 2nd year:

1. AHAN BANDYOPADHYAY (211210008)

2. GAURAV SINGH (211210024)

3. KANHA MADAN (211210031)

Have completed a project on the Database of the Bank Management System which cites all the information.

1.1 Project objective

. To allow only authorized user to access various function and processed available in the system.

. Locate any A/C wanted by the user.

. Reduced clerical work as most of the work done by computer.

. Provide greater speed & reduced time consumption.

1.2 Project benefits

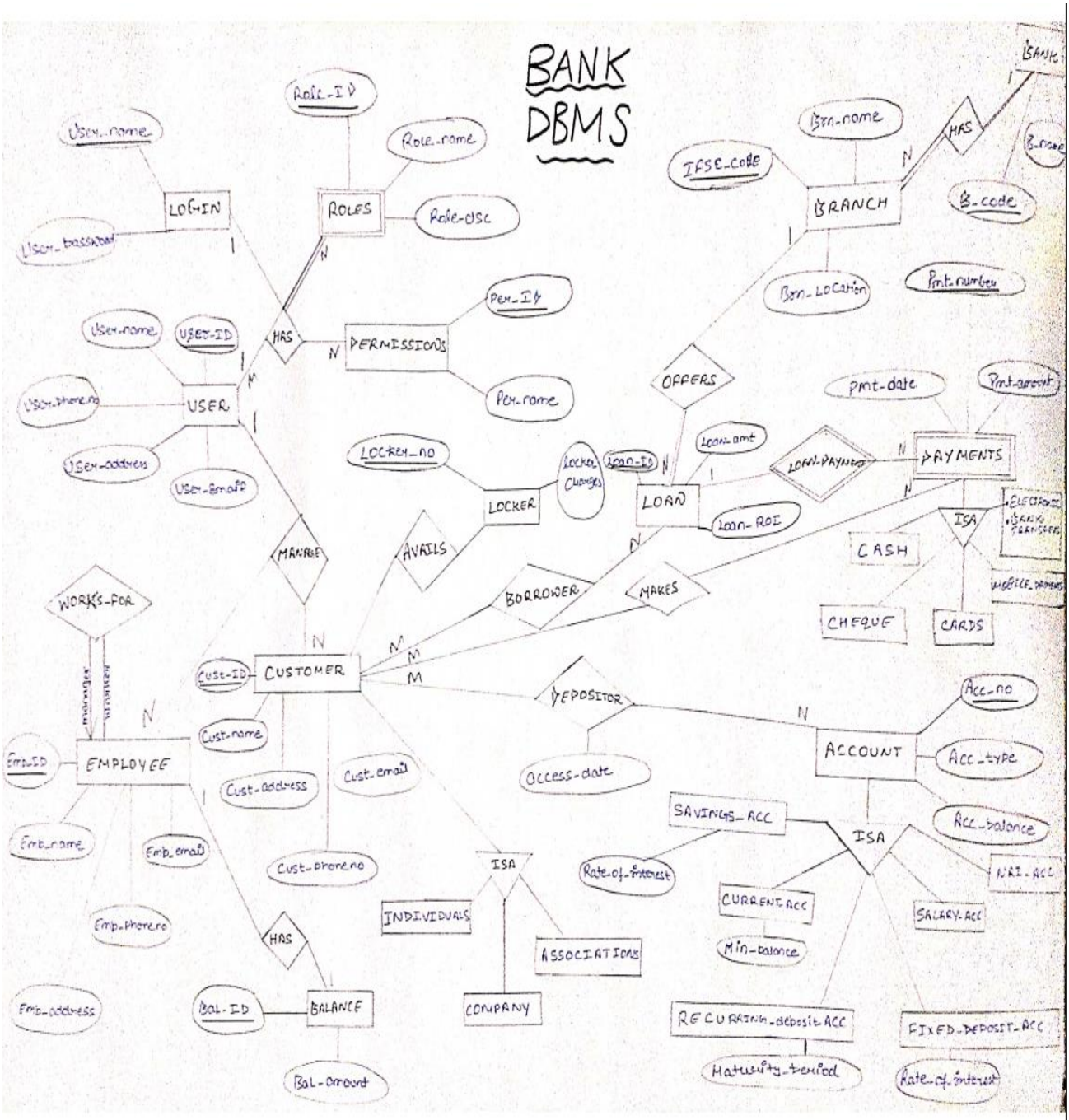
1.3 Project scope

Banking activities are considered to be the life blood of national economy. Without banking services, trading and business activities cannot be carried on smoothly. Banks are the distributors and protectors of liquid capital which is of vital significance to a developing country.
Efficient administration of the banking system helps in the economic Growth of the nation. Banking is useful to trade and commerce.

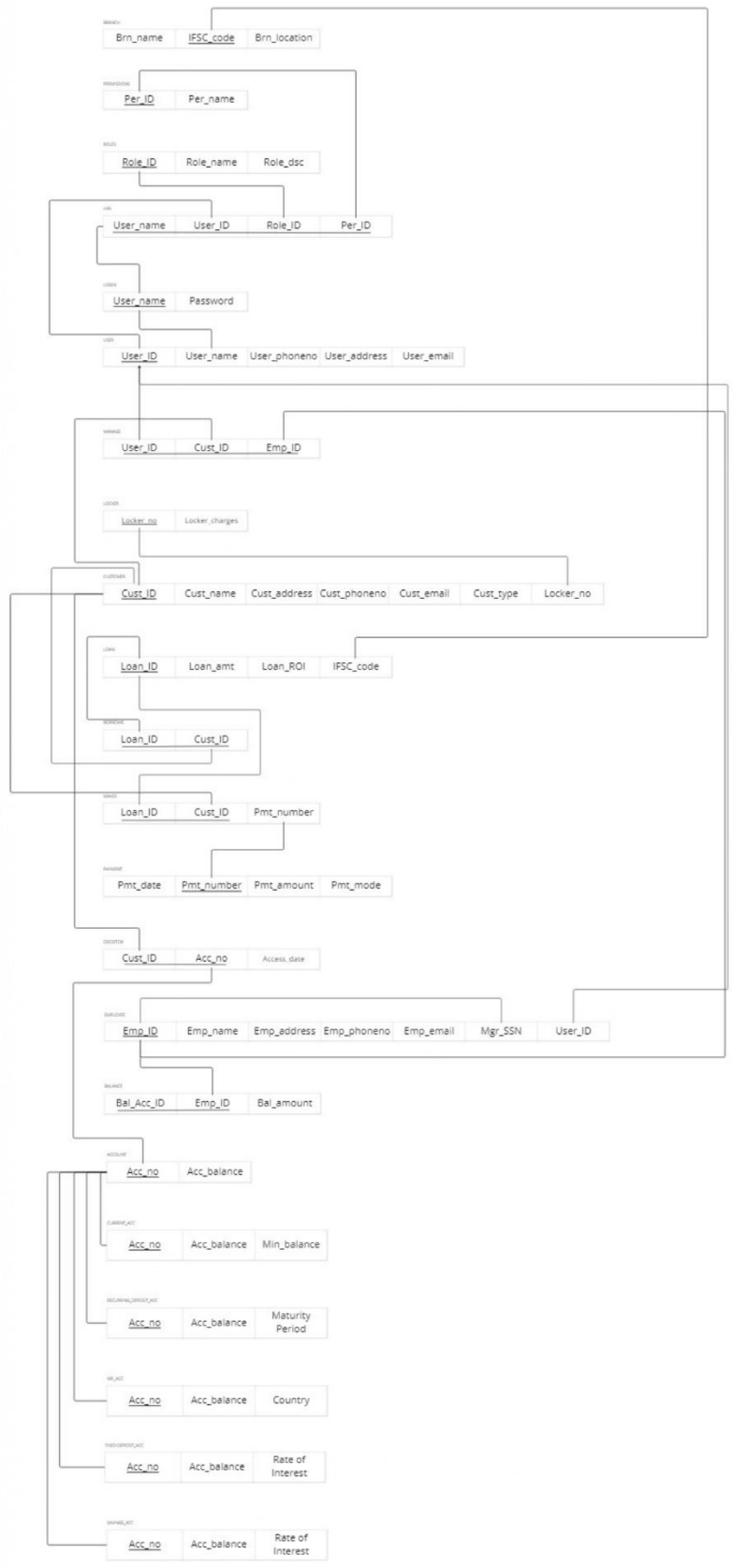
Software requirements

In the development of a project the selection of an appropriate DBMS Software and a platform is of primary importance. With many software options available a developer has to consider the various features and functionalities and ease of handling the software, keeping an account of such things we decided to use Bootstrap Studio for designing the front-end, the front-end has been developed by the use of HTML, CSS and JS. MySQL has been used as a back-end query language. PHP has been chosen as a scripting language. The server chosen is the localhost which would be hosting the website on the machine itself.

ER DIAGRAM


BANK DBMS
Role_ID
Role_name
Role_desc
ROLES
User_name
LOGIN
User_password
HAS
PERMISSIONS
Per_ID
Per_name
User_name
USER_ID
User_phone_no
USER
User_address
User_email
MANAGE
LOCKER_no
LOCKER
Locker_charges
AVAILS
WORKS-FOR
manages
supervision
EMPLOYEE
Emp_ID
Emp_name
Emp_email
Emp_phone_no
Emp_address
HAS
BAL_ID
BALANCE
Bal_amount
Cust_ID
CUSTOMER
Cust_name
Cust_address
Cust_email
Cust_phone_no
BORROWER
MAKES
OFFERS
Loan_ID
Loan_amt
LOAN
Loan_ROI
Locker_charges
LOAN_PAYING
pmt_date
Pmt_amount
PAYMENTS
ISA
CASH
CHEQUE
CARDS
DEPOSITOR
Access_date
ISA
INDIVIDUALS
ASSOCIATIONS
COMPANY
IFSC_code
Brn_name
BRANCH
HAS
Brn_Location
Pmt_number
ACCOUNT
Acc_no
Acc_type
Acc_balance
ISA
SAVINGS_ACC
Rate_of_interest
CURRENT_ACC
Min_balance
SALARY_ACC
RECURRING_deposit_ACC
Maturity_period
FIXED_DEPOSIT_ACC
Rate_of_interest

MAPPING ER -> MODEL

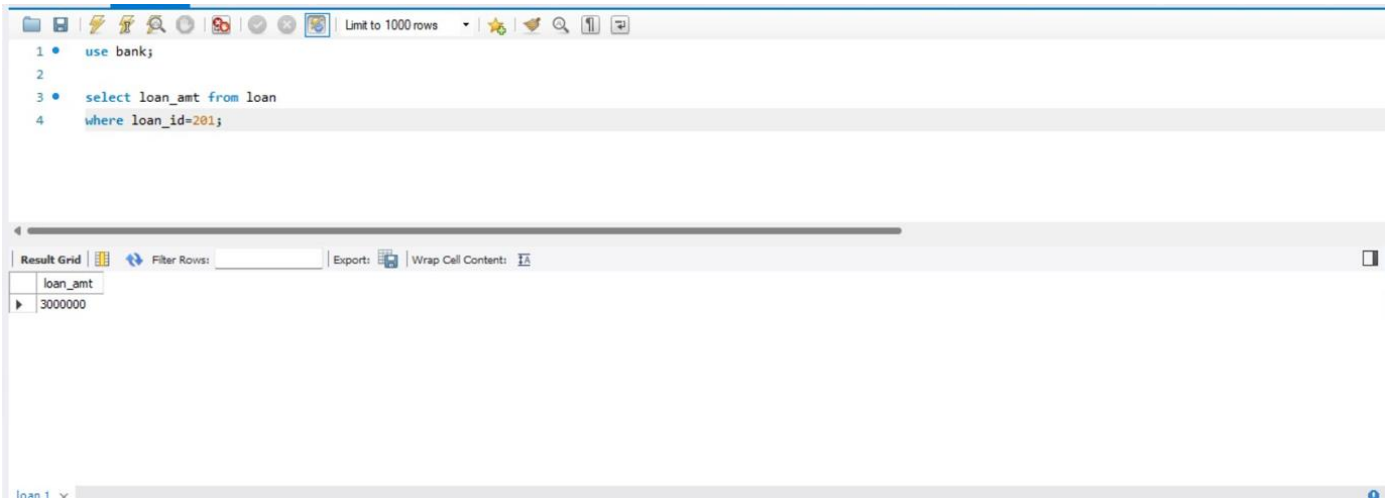

BRANCH
Brn_name
IFSC_code
Brn_location
PERMISSIONS
Per_ID
Per_name
ROLES
Role_ID
Role_name
Role_dsc
HAS
User_name
User_ID
Role_ID
Per_ID
LOGIN
User_name
Password
USER
User_ID
User_name
User_phoneno
User_address
User_email
MANAGE
User_ID
Cust_ID
Emp_ID
LOCKER
Locker_no
Locker_charges
CUSTOMER
Cust_ID
Cust_name
Cust_address
Cust_phoneno
Cust_email
Cust_type
Locker_no
LOAN
Loan_ID
Loan_amt
Loan_ROI
IFSC_code
BORROWS
Loan_ID
Cust_ID
MAKES
Loan_ID
Cust_ID
Pmt_number
PAYMENT
Pmt_date
Pmt_number
Pmt_amount
Pmt_mode
ACCESSION
Cust_ID
Acc_no
Access_date
EMPLOYEE
Emp_ID
Emp_name
Emp_address
Emp_phoneno
Emp_email
Mgr_SSN
User_ID
BALANCE
Bal_Acc_ID
Emp_ID
Bal_amount
ACCOUNT
Acc_no
Acc_balance
CURRENT_ACC
Acc_no
Acc_balance
Min_balance
SECURING_DEPOSIT_ACC
Acc_no
Acc_balance
Maturity Period
SAV_ACC
Acc_no
Acc_balance
Country
FIXED-DEPOSIT_ACC
Acc_no
Acc_balance
Rate of Interest
SAVINGS_ACC
Acc_no
Acc_balance
Rate of Interest

SQL queries

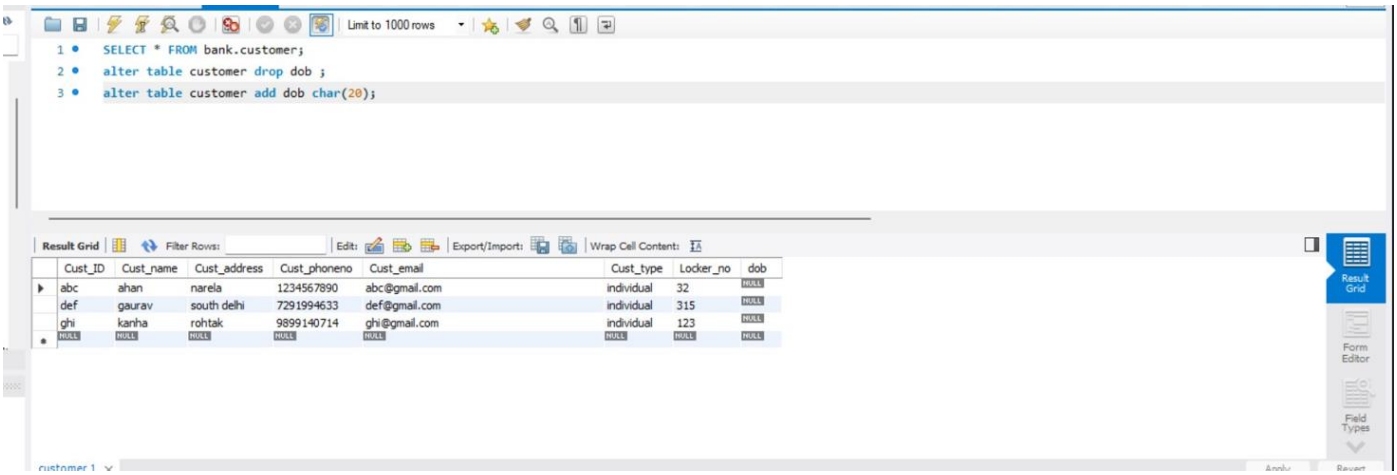
QUERIES TO CREATE THE RELATIONS AND POPULATING THE DATABASE:

Basic queries

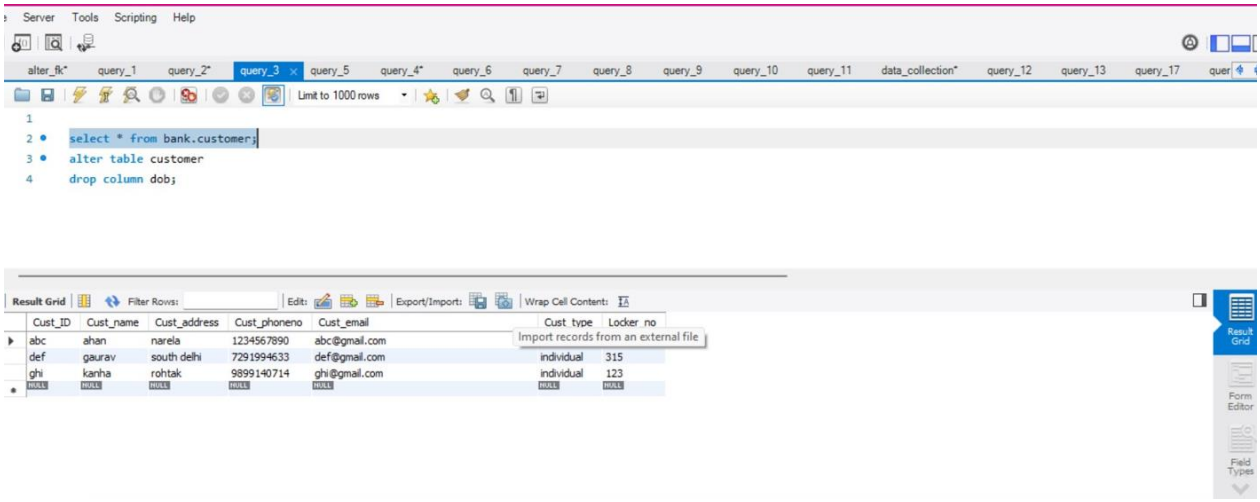
Question 1: Retrieve loan details where loan id = 201



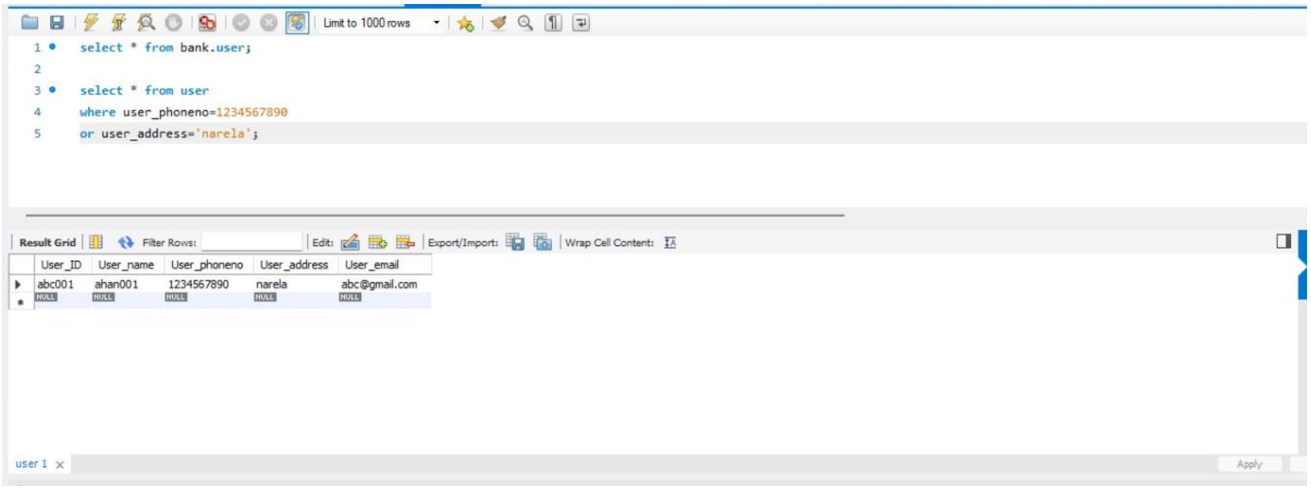
Question 2: Add attribute date of birth (dob) for customer



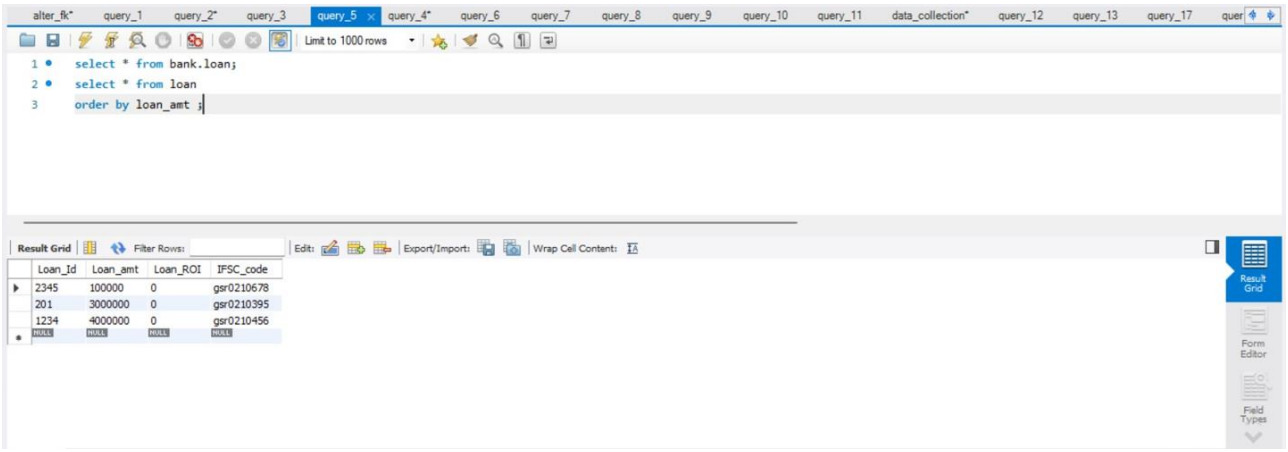
Question 3: Drop attribute dob for customer



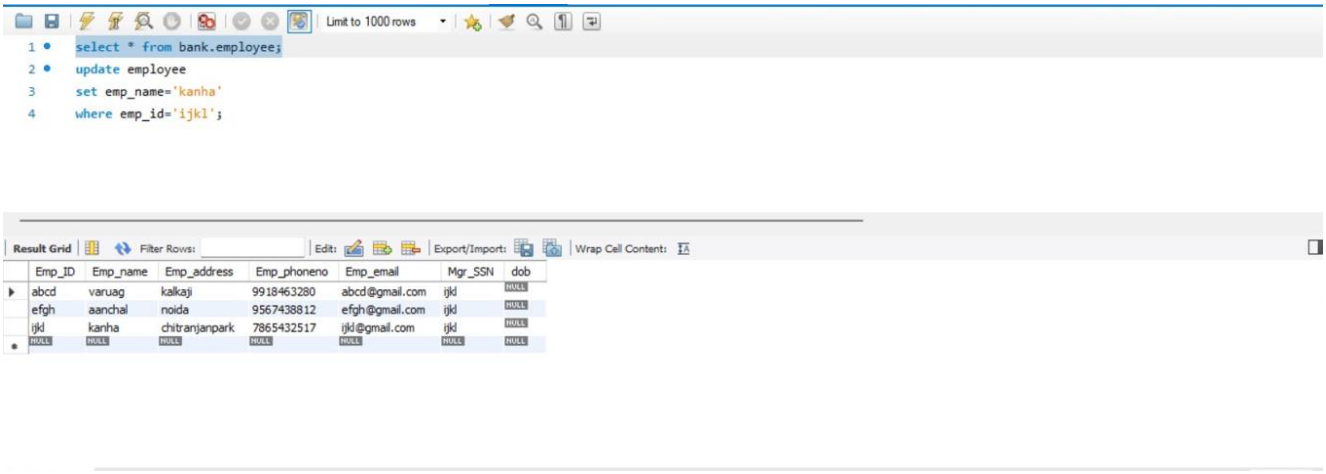
Question 4: Retrieve user details for user having mobile no. = 1234567890 and address = 'Narela'



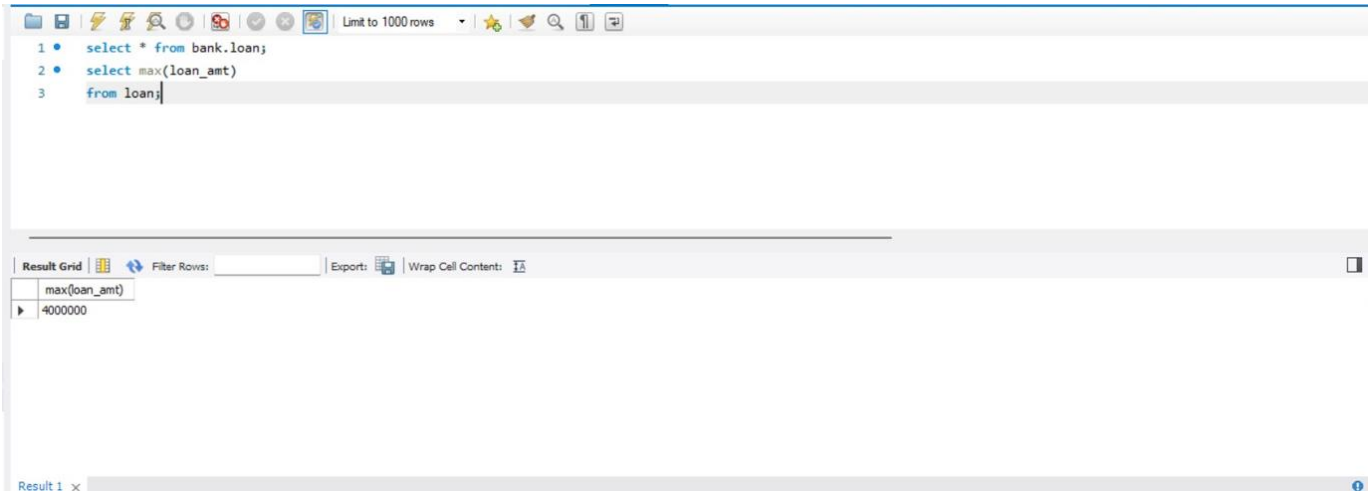
Question 5: Retrieve loan details order by ascending loan amount



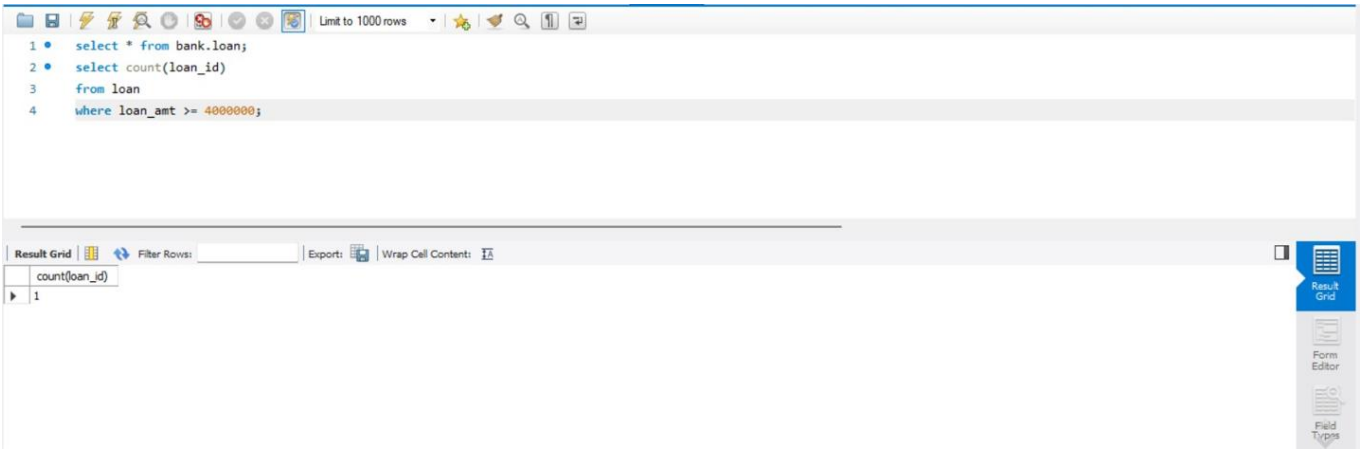
Question 6:Update name of employee having employee ID = 'ijkl'



Question 7: Retrieve maximum amount of loan lend by any customer from bank



Question 8: Count the distinct users that have taken loan of and above 4000000



Question 9: Retrieve loan details of user who have lend a loan between 1000000 to 100000000

```
select * from bank.loan;

use bank;

select *
from loan
where loan_amt between 1000000 and 10000000;
```

Result Grid

Loan_Id	Loan_amt	Loan_ROI	IFSC_code
1234	4000000	0	gsr0210456
201	3000000	0	gsr0210395
NULL	NULL	NULL	NULL

Question 10: Retrieve depositor details for user having 7 in their account no. anywhere

```
select * from bank.depositor;
select * from depositor
where acc_no
like'__7%';
```

Result Grid

Cust_ID	Acc_no	Access_date
abc	5037720309	2023-03-12
def	5147756908	2024-04-13
NULL	NULL	NULL

depositor1

Question 11: Retrieve combine data of employee data and customer table as employee ID , employee names, employee address

```
1  select emp_id, emp_name, emp_address
2  from employee
3  union
4  select cust_id, cust_name, cust_address
5  from customer;
```

Result Grid

emp_id	emp_name	emp_address
abcd	varuag	kalkaji
efgh	aanchal	noida
ijkl	dhruv	chitranjanpark
abc	ahan	narela
def	gaurav	south delhi
ghi	kanha	rohtak

Question 12: Find total amount of payments performed in each payment mode

```
1  select sum(pmt_amount)
2  from payment
3  group by pmt_mode;
```

Result Grid

sum(pmt_amount)
100000
20000
4000

Result 1

Question 13: Retrieve user details of user having 'a' in their name.

```
1  select user_name, user_email
2  from user
3  group by user_name
4  having user_name like'%a%';
```

Result Grid

user_name	user_email
ahan001	abc@gmail.com
varuag01	abcd@gmail.com
gaurav002	def@gmail.com
aanchal02	efgh@gmail.com
kanha003	ghi@gmail.com

user 1

Question 14: Retrieve user data in ascending order of their names

```
use bank;
select * from user
order by user_name asc;
```

User_ID	User_name	User_phoneno	User_address	User_email
efgh_2	aanchal02	9567438812	noida	efgh@gmail.com
abc001	ahan001	1234567890	narela	abc@gmail.com
def002	gaurav002	7291994633	south delhi	def@gmail.com
ijkl_3	gsr03	7865432517	chitranjanpark	ijkl@gmail.com
ghi003	kanha003	9899140714	rohtak	ghi@gmail.com
abcd_1	varuag01	9918463280	kalkaji	abcd@gmail.com

Advance queries

Question 15: Retrieve names of all customers who have performed payment via net banking

```
select distinct customer.cust_name
from customer, makes, payment
where customer.cust_id = makes.cust_id and makes.pmt_number =
(select payment.pmt_number
from payment
where pmt_mode = 'netbanking');
```

cust_name
kanha

Question 16: retrieve loan amount taken by the customer having locker no. = 123

```
use bank;
select distinct loan.loan_amt
from loan, customer
where loan.loan_id =
    (select borrows.loan_id
    from borrows
    where borrows.cust_id=
        (select customer.cust_id
         from customer
         where locker_no=123));
```

loan_amt
100000

Question 17: Loan details of customer with locker no '123'

```
select borrows.loan_id
    from borrows
    where borrows.cust_id=
(select customer.cust_id
from customer
where locker_no=123)
```

loan_id
2345

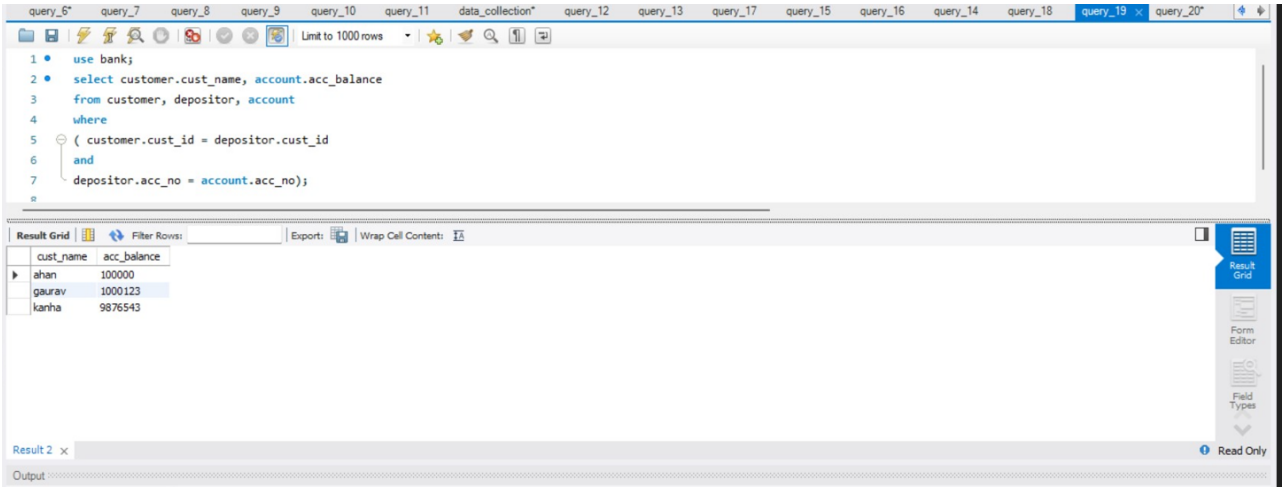
Question 18: Retrieve ID of all customers having account in bank. That are from 'Norway

```
use bank;

select distinct customer.Cust_ID
from customer, account, nri_acc, depositor
where customer.Cust_ID = depositor.Cust_ID and depositor.Acc_no =
  (
   select account.Acc_no
   from account, nri_acc
   where account.Acc_no = nri_acc.Acc_no and country = 'norway' );
```

Cust_ID
abc

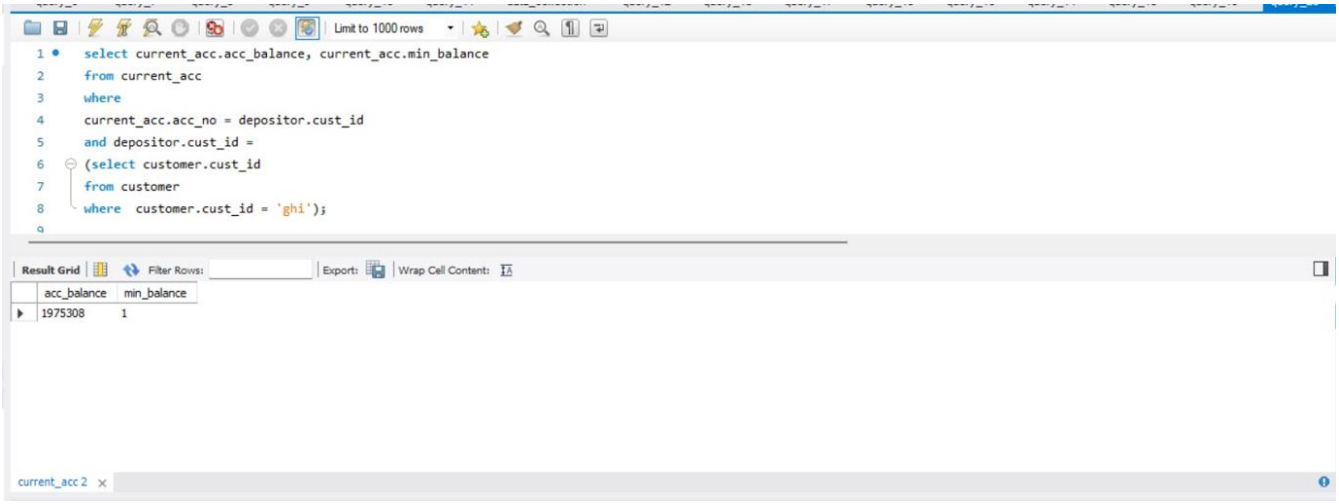
Question 19: Show customer names with their account balance.



```
use bank;
select customer.cust_name, account.acc_balance
from customer, depositor, account
where
( customer.cust_id = depositor.cust_id
and
depositor.acc_no = account.acc_no);
```

cust_name	acc_balance
ahan	100000
gaurav	1000123
kanha	9876543

Question 20: Show account balance and account minimum balance limit for customer having id='ghi'.



```
select current_acc.acc_balance, current_acc.min_balance
from current_acc
where
current_acc.acc_no = depositor.cust_id
and depositor.cust_id =
(select customer.cust_id
from customer
where  customer.cust_id = 'ghi');
```

acc_balance	min_balance
1975308	1