

INFO 6210
Data Management and Database Design

Aman Nigam
NUID: 001494916

Bank Management System
FINAL PROJECT

PROBLEM STATEMENT:

- The Bank Management System provides services to manage bank accounts at a bank.
- The bank has many branches and their individual address.
- The bank provides two types of account including Savings and Checking.
- A customer can open an account in any of its branches.
- The bank provides ATM and Internet banking facility to the customer.
- It also has a separate department for loan.
- Three types of loan are provided by the bank namely, Education Loan, Auto Loan and Home Loan.
- The bank also has employees for each of their departments.

DESIGNING THE DATABASE

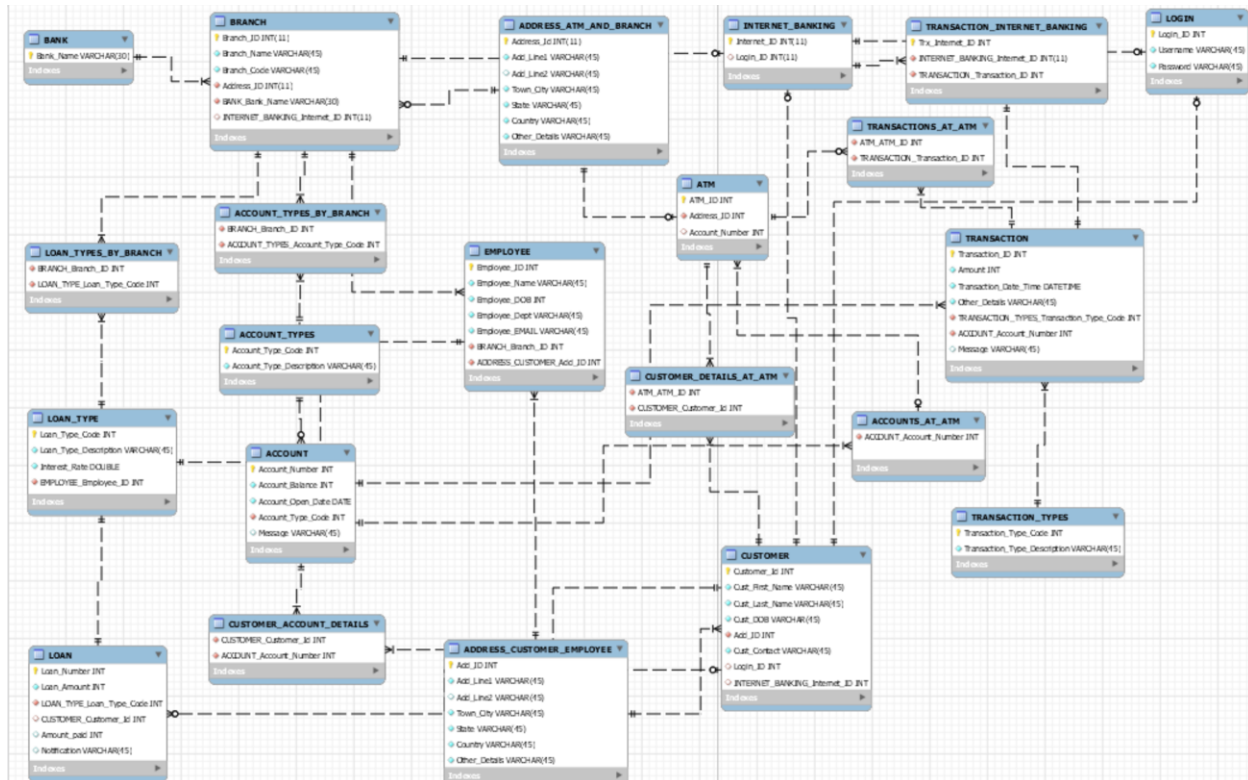
The designing of the database started with the schema. A rough estimate of how many tables would be required to achieve the objective. I created a tentative ER diagram to help me understand how to relate different tables and what all columns each table will hold.

- 1) A pool of columns for a table were made. These columns were Bank_name, Branch_name, branch_id, branch_add_line1, branch_add_line2, branch_city, branch_branch_state, branch_zipcode, branch_code. They all were in a table named bank.
- 2) Next table was of accounts which has columns such as account_number(PK), account_balance, acc_open_date, account_type_code, branch_id(FK), customer_id(FK).
- 3) An account will have two tables of savings and checking account that will have account_number as foreign key.
- 4) An address table to hold address of customer. The columns in this table were address_id(PK), line_1, line_2, town_city, zip_code, state, country, other_details.
- 5) A customer table to hold information about customer_id(PK), customer_first_name, customer_last_name, customer_dob, address_id(FK), branch_id(FK), account_number(FK).
- 6) A loans table that will be connected to the customer table and will have loan_number(PK), Loan_type_code, loan_type_description(Home loan, education loan, auto loan).
- 7) An ATM table that has its own address and atm_id(PK).
- 8) ATM will have relationship with accounts table with an account_number(FK).
- 9) The most important transaction table that will include columns such as transaction_id(PK), transaction_type_code, amount, account_type_code, account_number(FK), transaction_date_time, customer_id, and atm_id(FK).
- 10) A Login table that will be related to customer and to internet banking table. It will store customer's username and password and customer_id(FK).
- 11) The internet banking table will have the transaction_type_code(FK), transaction_id(FK), Internet_id(PK).
- 12) There would be two tables named deposit and withdrawal that will be related to the internet banking table.

These were the tables and its columns that I decided that would be required to attain my project objective. These are all un-normalized tables and many still needs some other attributes to be included.

After a couple of more attempts to find the right tables for each of the columns and to get the tables normalized to atleast to 3rd normal form. I was able to reach to my final design whose Entity Relationship diagram is shown below with its attachment.


Bank ER
Diagram.mwb



The normalized form of the schema included 22 tables. The following major changes were done as compared to first version:

- 1) Bank name was moved to a separate table named BANK.
- 2) BRANCH table included Bank_Name, Address_ID and Internet_Banking_Id as foreign key.
- 3) The address of the branch is now moved to a separate table named ADDRESS_ATM_AND_BRANCH that stores the address of both the branch and the ATM.

- 4) The ACCOUNTS table now hold just 5 columns with account_number as the primary key and account_type_code foreign key.
- 5) Two separate table for saving and checking account is now a single table with account_type_code and account_type_description as its only column.
- 6) The address table for customer now holds address for branch employees and as well as that of customer.
- 7) Few columns were appended to the customer table namely, login_id and internet_banking_id which are the primary key to their respective table.
- 8) Loan_Type_code and Loan_type_description has been now moved to a new table named LOAN_TYPE and include two new columns Interest_Rate and Employee_ID(FK).
- 9) The LOAN table has 6 columns now namely, Loan_Number(PK), Loan_Amount, Loan_Type_Code(FK), Customer_Id(FK), Amount_Paid and Notification.
- 10) The deposit and withdrawal table has been combined to a single table named TRANSACTION_TYPES.
- 11) The TRANSACTION table has a transaction_id(PK), amount, transaction_date_time, other_details, Transaction_Type_Code, Account_number and message columns.
- 12) There are 6 bridge tables between Branch and account_types, account and customer, branch and loan_type, ATM and Customer, ATM and Account, ATM and Transaction.

PROCEDURES, VIEW, TRIGGER, FUNCTIONS

VIEWS

A view named “Auto_Loan_Details_Customer” that display details of the customer who has taken an Auto Loan from the bank branch.

The same view could be used to display the details of customers who have taken an education or home loan. This could be achieved by just changing the hard coded value of auto loans to education or home loan.

```
CREATE VIEW Auto_Loan_Details_Customer AS
```

```
SELECT
```

```
    b.Branch_Name AS 'Branch Name',
```

```
    a.account_number AS 'A/C Number',
```

```
    CONCAT_WS(' ', c.Cust_First_Name, c.Cust_Last_Name) AS 'Customer Name',
```

```
    c.Cust_Contact 'Contact',
```

```
    I3.loan_number AS 'Loan Number',
```

```
    I2.loan_type_description AS 'Loan Type',
```

```
    I2.interest_rate AS 'Interest Rate',
```

```
    I3.Loan_Amount AS 'Loan Amount',
```

```
    I2.loan_type_code AS 'Loan Type Code',
```

```
    e.employee_name AS 'Employee Name'
```

```
FROM
```

```
    EMPLOYEE e,
```

```
    branch b,
```

```
    LOAN_TYPES_BY_BRANCH I1,
```

```
    LOAN_TYPE I2,
```

```
    LOAN I3,
```

```
    CUSTOMER c,
```

```
CUSTOMER_ACCOUNT_DETAILS cad,  
ACCOUNT a  
WHERE  
13.customer_customer_id = c.customer_id  
AND c.customer_id = cad.customer_customer_id  
AND cad.account_account_number = a.account_number  
AND 12.EMPLOYEE_Employee_ID = e.Employee_ID  
AND b.branch_id = 11.branch_branch_id  
AND 11.loan_type_loan_type_code = 12.loan_type_code  
AND 12.loan_type_code = 13.loan_type_loan_type_code  
AND b.branch_id = 201  
AND 12.loan_type_description = 'Auto Loans';
```

```
drop view Loan_Details_Customer;
```

```
SELECT
```

```
*
```

```
FROM
```

```
Auto_Loan_Details_Customer;
```

PROCEDURE AND FUNCTION

A procedure named “loan_amt_paid_func” that updates the loan table column Notification with a message whether a particular customer can apply for another loan or not based on whether he has paid atleast 50% of the previous loan amount. The procedure takes two input parameters customer_id and pay_amt.

The procedure initially updates the amount_paid column of the loan table based on the customer_id.

The same procedure call the function “loan_check” that takes two parameters loan_amount and new_amt_paid and returns the message based on the calculation of previous loan amount paid atleast 50%.

If amount paid is greater than the 50% value it will return a message “You are eligible to apply for another loan”.

While if it is less than that amount it will return a message “First!Please pay atleast 50% of previous loan”.

The returned message is updated on the notification column of loan table that is defined in the procedure.

PROCEDURE:

DELIMITER //

CREATE PROCEDURE loan_amt_paid_func

(IN cust_id INT,IN pay_amt INT)

BEGIN

 SELECT

 l.loan_number,

 @loan_amount:=l.loan_amount,

 c.customer_id,

 @amt_paid:=l.amount_paid

FROM

 customer c,

loan l

WHERE

c.customer_id = l.CUSTOMER_Customer_id
AND l.CUSTOMER_Customer_id = cust_id;

UPDATE loan

SET

amount_paid = (pay_amt + @amt_paid)

WHERE

CUSTOMER_Customer_id = cust_id;

SELECT

@new_amt_paid:=l.amount_paid

FROM

customer c,

loan l

WHERE

c.customer_id = l.CUSTOMER_Customer_id
AND l.CUSTOMER_Customer_id = cust_id;

SELECT @notification:=LOAN_CHECK(@loan_amount, @new_amt_paid);

UPDATE loan

SET

Notification = @notification

WHERE

CUSTOMER_Customer_id = cust_id;

END //

DELIMITER ;

CALL loan_amt_paid_func(903,5000);

FUNCTION:

DELIMITER //

CREATE FUNCTION loan_check(loan_amount int,amount_paid int)

RETURNS text

BEGIN

set @cal=loan_amount*(50/100);

IF amount_paid>@cal then

return 'You are eligible to apply for another loan';

else

return 'First!Please pay atleast 50% of previous loan';

end if;

END //

DELIMITER ;

drop function loan_check;

SELECT LOAN_CHECK(15000, 903, 10000);

TRIGGER

A trigger named “update_accbalance_trg” that updates the account_balance column of the ACCOUNT table whenever an update is performed on the TRANSACTION table for that particular account_number. The balance is updated based on the amount held in the account plus the transaction amount. This is an After Update trigger.

```
DELIMITER //
```

```
CREATE TRIGGER update_accbalance_trg
```

```
After UPDATE ON transaction
```

```
For each row
```

```
BEGIN
```

```
declare acc_bal int;
```

```
    SELECT
```

```
        account_balance
```

```
INTO acc_bal FROM
```

```
    account
```

```
WHERE
```

```
    account_number = new.account_account_number;
```

```
    UPDATE account
```

```
SET
```

```
    account_balance = acc_bal + new.amount
```

```
WHERE
```

```
    account_number = new.account_account_number;
```

```
end //
```

```
delimiter ;
```

EXTRAS

- 1) The following query will give the data regarding customer transaction details. It list all the transaction done by a particular customer based on his account_number.

SELECT

d.CUSTOMER_Customer_Id AS 'Customer ID',
CONCAT_WS(' ', e.Cust_First_Name, e.Cust_Last_Name) AS 'Customer Name',
c.Account_Number AS 'A/C Number',
f.Transaction_Type_Description AS 'Transaction Type',
b.Transaction_Id AS 'Transaction ID',
b.Amount,
b.Transaction_Date_Time AS 'Date',
b.Other_Details AS 'Type'

FROM

TRANSACTION b,
account c,
customer_account_details d,
customer e,
TRANSACTION_TYPES f

WHERE

b.TRANSACTION_TYPES_Transaction_Type_Code = f.Transaction_Type_Code
AND b.ACCOUNT_Account_number = c.Account_Number
AND c.Account_Number = d.ACCOUNT_Account_Number
AND d.CUSTOMER_Customer_Id = e.customer_id
AND d.CUSTOMER_Customer_Id = e.customer_id
AND c.Account_Number = 9995;

- 2) Apart from this, I have created a procedure named “Customer_Transaction_new” that will update the transaction table, message column if the transaction done is more than \$400. So if it is more than \$400 then it will update the message column with “Transaction greater than 400 are not allowed” else “Your recent transaction was successful”.

DELIMITER //

CREATE PROCEDURE Customer_Transaction_new

(IN acc_num INT)

BEGIN

SELECT

d.CUSTOMER_Customer_Id AS 'Customer ID',

CONCAT_WS(' ', e.Cust_First_Name, e.Cust_Last_Name) AS 'Customer Name',

c.Account_Number AS 'A/C Number',

f.Transaction_Type_Description AS 'Transaction Type',

b.Transaction_Id AS 'Transaction ID',

b.Amount,

b.Transaction_Date_Time AS 'Date',

b.Other_Details AS 'Type'

FROM

TRANSACTION b,

account c,

customer_account_details d,

customer e,

TRANSACTION_TYPES f

WHERE

b.TRANSACTION_TYPES_Transaction_Type_Code = f.Transaction_Type_Code

AND b.ACCOUNT_Account_number = c.Account_Number

```
AND c.Account_Number = d.ACCOUNT_Account_Number
AND d.CUSTOMER_Customer_Id = e.customer_id
AND d.CUSTOMER_Customer_Id = e.customer_id
AND c.Account_Number = 9995;
        and b.Transaction_Date_Time<='2018-12-11 21:52:24'
and b.Transaction_Date_Time>'2018-12-11 17:42:17'
```

```
If @amount > 400 then
```

```
        update TRANSACTION set message='Transaction greater than 400 are not
allowed'
```

```
        where Transaction_Id=@trx_id;
```

```
else
```

```
        update account set message='Your recent transaction was successful'
```

```
        where Transaction_Id=@trx_id;
```

```
end if;
```

```
END //
```

```
DELIMITER ;
```

```
CALL Customer_Transaction_new(9995);
```

```
drop procedure Customer_Transaction_new;
```

- 3) Created a trigger “Bal_upd” that will update a column named balance in a temporary table acc_temp after every insertion on the TRANSACTION table. The balance will be the addition amount in the account_balance column in ACCOUNTS table and the amount column in the transaction table.

Temporary Table:

```
CREATE TABLE acc_temp (  
    trx_desc VARCHAR(45),  
    balance INT,  
    old_balance INT,  
    new_balance INT,  
    acc_num INT,  
    amount INT  
);
```

Trigger:

```
DELIMITER //  
  
CREATE TRIGGER Bal_upd  
After INSERT ON transaction  
For each row  
BEGIN  
    declare type int;  
    declare amount int;  
    declare account_num int;  
    declare type_desc varchar(45);  
    declare balance int;  
  
    SELECT  
    tt.Transaction_Type_Code,
```

```

t.Amount,
acc.Account_Number,
tt.Transaction_Type_Description,
acc.account_balance
INTO type , amount , account_num , type_desc , balance FROM
TRANSACTION_TYPES tt,
Account acc,
transaction t
WHERE
t.TRANSACTION_TYPES_Transaction_Type_Code = tt.Transaction_Type_Code
AND t.ACCOUNT_Account_Number = acc.Account_Number
AND acc.Account_Number = 6000;

IF type = 2001 then
Update acc_temp set balance=balance+amount
where acc_num=account_num;

else

Update acc_temp set balance=amount-balance
where acc_num=account_num;

end if;

END //
DELIMITER ;

DROP trigger Bal_upd;

```


PRIVILEGES

```
create user 'cust@localhost' identified by '123456';
```

```
revoke all privileges, grant option from 'cust@localhost';
```

```
grant insert (add_id,add_line1,add_line2,town_city,state,country,other_details)
on nigamaman.ADDRESS_CUSTOMER_EMPLOYEE
to 'cust@localhost';
```

```
grant insert
(Customer_id,cust_first_name,cust_last_name,cust_dob,add_id,cust_contact,login_id,internet_ba
nking_internet_id)
on nigamaman.customer
to 'cust@localhost';
```

```
grant insert (login_id,username,password)
on nigamaman.login
to 'cust@localhost';
```

```
grant update (add_id,add_line1,add_line2,town_city,state,country,other_details)
on nigamaman.ADDRESS_CUSTOMER_EMPLOYEE
to 'cust@localhost';
```

```
grant update
(Customer_id,cust_first_name,cust_last_name,cust_dob,add_id,cust_contact,login_id,internet_ba
nking_internet_id)
on nigamaman.customer
```

```
to 'cust@localhost';
```

```
grant update (login_id,username,password)
```

```
on nigamaman.login
```

```
to 'cust@localhost';
```

```
grant select
```

```
(transaction_id,amount,transaction_date_time,other_details,transaction_types_transaction_type_  
code,account_account_number)
```

```
on nigamaman.transaction
```

```
to 'cust@localhost';
```

```
drop user 'cust@localhost';
```

APPENDIX

MySQL Script

```
-- MySQL dump 10.13  Distrib 8.0.12, for Win64 (x86_64)

--

-- Host: localhost  Database: nigamaman

-- -----

-- Server version      8.0.12


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS
*/;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
SET NAMES utf8 ;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0
*/;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;


--

-- Table structure for table `acc_temp`

--
```

```

DROP TABLE IF EXISTS `acc_temp`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `acc_temp` (
  `trx_desc` varchar(45) DEFAULT NULL,
  `balance` int(11) DEFAULT NULL,
  `old_balance` int(11) DEFAULT NULL,
  `new_balance` int(11) DEFAULT NULL,
  `acc_num` int(11) DEFAULT NULL,
  `amount` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `account`
--

```

```

DROP TABLE IF EXISTS `account`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `account` (
  `Account_Number` int(11) NOT NULL,
  `Account_Balance` double DEFAULT NULL,
  `Account_Open_Date` date NOT NULL,
  `Account_Type_Code` int(11) NOT NULL,
  PRIMARY KEY (`Account_Number`),
  KEY `fk_ACCOUNT_ACCOUNT_TYPES1_idx` (`Account_Type_Code`),
  CONSTRAINT `fk_ACCOUNT_ACCOUNT_TYPES1` FOREIGN KEY
  (`Account_Type_Code`) REFERENCES `account_types` (`account_type_code`)

```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--
```

```
-- Table structure for table `account_types`
```

```
--
```

```
DROP TABLE IF EXISTS `account_types`;
```

```
/*!40101 SET @saved_cs_client = @@character_set_client */;
```

```
SET character_set_client = utf8mb4 ;
```

```
CREATE TABLE `account_types` (
```

```
  `Account_Type_Code` int(11) NOT NULL,
```

```
  `Account_Type_Description` varchar(45) NOT NULL,
```

```
  PRIMARY KEY (`Account_Type_Code`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--
```

```
-- Table structure for table `account_types_by_branch`
```

```
--
```

```
DROP TABLE IF EXISTS `account_types_by_branch`;
```

```
/*!40101 SET @saved_cs_client = @@character_set_client */;
```

```
SET character_set_client = utf8mb4 ;
```

```
CREATE TABLE `account_types_by_branch` (
```

```
  `BRANCH_Branch_ID` int(11) NOT NULL,
```

```
  `ACCOUNT_TYPES_Account_Type_Code` int(11) NOT NULL,
```

```
  KEY `fk_BRANCH_has_ACCOUNT_TYPES_ACCOUNT_TYPES1_idx`  
  (`ACCOUNT_TYPES_Account_Type_Code`),
```

```

KEY `fk_BRANCH_has_ACCOUNT_TYPES_BRANCH1_idx` (`BRANCH_Branch_ID`),
CONSTRAINT `fk_BRANCH_has_ACCOUNT_TYPES_ACCOUNT_TYPES1` FOREIGN
KEY (`ACCOUNT_TYPES_Account_Type_Code`) REFERENCES `account_types`
(`account_type_code`),
CONSTRAINT `fk_BRANCH_has_ACCOUNT_TYPES_BRANCH1` FOREIGN KEY
(`BRANCH_Branch_ID`) REFERENCES `branch` (`branch_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `accounts_at_atm`
--

```

```

DROP TABLE IF EXISTS `accounts_at_atm`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
SET character_set_client = utf8mb4;
CREATE TABLE `accounts_at_atm` (
  `ACCOUNT_Account_Number` int(11) NOT NULL,
  KEY `fk_ACCOUNT_has_ATM_ACCOUNT1_idx` (`ACCOUNT_Account_Number`),
  CONSTRAINT `fk_ACCOUNT_has_ATM_ACCOUNT1` FOREIGN KEY
(`ACCOUNT_Account_Number`) REFERENCES `account` (`account_number`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `address_atm_and_branch`
--

```

```

DROP TABLE IF EXISTS `address_atm_and_branch`;
/*!40101 SET @saved_cs_client = @@character_set_client */;

```

```

SET character_set_client = utf8mb4 ;
CREATE TABLE `address_atm_and_branch` (
  `Address_Id` int(11) NOT NULL,
  `Add_Line1` varchar(45) NOT NULL,
  `Add_Line2` varchar(45) DEFAULT NULL,
  `Town_City` varchar(45) NOT NULL,
  `State` varchar(45) NOT NULL,
  `Country` varchar(45) NOT NULL,
  `Other_Details` varchar(45) NOT NULL,
  PRIMARY KEY (`Address_Id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```
--
```

```
-- Table structure for table `address_customer_employee`
```

```
--
```

```

DROP TABLE IF EXISTS `address_customer_employee`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
SET character_set_client = utf8mb4 ;
CREATE TABLE `address_customer_employee` (
  `Add_ID` int(11) NOT NULL,
  `Add_Line1` varchar(45) NOT NULL,
  `Add_Line2` varchar(45) DEFAULT NULL,
  `Town_City` varchar(45) NOT NULL,
  `State` varchar(45) NOT NULL,
  `Country` varchar(45) NOT NULL,
  `Other_Details` varchar(45) NOT NULL,

```

```

PRIMARY KEY (`Add_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `atm`
--

DROP TABLE IF EXISTS `atm`;

/*!40101 SET @saved_cs_client = @@character_set_client */;
SET character_set_client = utf8mb4 ;
CREATE TABLE `atm` (
  `ATM_ID` int(11) NOT NULL,
  `Address_ID` int(11) NOT NULL,
  `Account_Number` int(11) DEFAULT NULL,
  PRIMARY KEY (`ATM_ID`),
  KEY `fk_Address_ID_idx` (`Address_ID`),
  KEY `fk_Account_Number_idx` (`Account_Number`),
  CONSTRAINT `fk_Account_Number` FOREIGN KEY (`Account_Number`) REFERENCES
`accounts_at_atm` (`account_account_number`),
  CONSTRAINT `fk_Address_ID_ATM` FOREIGN KEY (`Address_ID`) REFERENCES
`address_atm_and_branch` (`address_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Temporary view structure for view `auto_loan_details_customer`
--

```



```

DROP TABLE IF EXISTS `auto_loan_details_customer`;

/*!50001 DROP VIEW IF EXISTS `auto_loan_details_customer`*/;

SET @saved_cs_client = @@character_set_client;

SET character_set_client = utf8mb4;

/*!50001 CREATE VIEW `auto_loan_details_customer` AS SELECT
  1 AS `Branch Name`,
  1 AS `A/C Number`,
  1 AS `Customer Name`,
  1 AS `Contact`,
  1 AS `Loan Number`,
  1 AS `Loan Type`,
  1 AS `Interest Rate`,
  1 AS `Loan Amount`,
  1 AS `Loan Type Code`,
  1 AS `Employee Name`*/;

SET character_set_client = @saved_cs_client;

--
-- Table structure for table `bank`
--

DROP TABLE IF EXISTS `bank`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `bank` (
  `Bank_Name` varchar(30) NOT NULL,
  PRIMARY KEY (`Bank_Name`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

```

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `branch`
--

DROP TABLE IF EXISTS `branch`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
SET character_set_client = utf8mb4 ;
CREATE TABLE `branch` (
  `Branch_ID` int(11) NOT NULL,
  `Branch_Name` varchar(45) NOT NULL,
  `Branch_Code` varchar(45) NOT NULL,
  `Address_ID` int(11) NOT NULL,
  `BANK_Bank_Name` varchar(30) NOT NULL,
  `INTERNET_BANKING_Internet_ID` int(11) DEFAULT NULL,
  PRIMARY KEY (`Branch_ID`),
  KEY `fk_BRANCH_ADDRESS_ATM_BRANCH1_idx` (`Address_ID`) /*!80000
  INVISIBLE */ ,
  KEY `fk_BRANCH_BANK1_idx` (`BANK_Bank_Name`),
  KEY `fk_BRANCH_INTERNET_BANKING1_idx` (`INTERNET_BANKING_Internet_ID`),
  CONSTRAINT `fk_ADDRESS_ID` FOREIGN KEY (`Address_ID`) REFERENCES
  `address_atm_and_branch` (`address_id`),
  CONSTRAINT `fk_BRANCH_BANK1` FOREIGN KEY (`BANK_Bank_Name`)
  REFERENCES `bank` (`bank_name`),
  CONSTRAINT `fk_BRANCH_INTERNET_BANKING1` FOREIGN KEY
  (`INTERNET_BANKING_Internet_ID`) REFERENCES `internet_banking` (`internet_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

```

--

-- Table structure for table `customer`

--

DROP TABLE IF EXISTS `customer`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `customer` (

 `Customer_Id` int(11) NOT NULL,

 `Cust_First_Name` varchar(45) NOT NULL,

 `Cust_Last_Name` varchar(45) NOT NULL,

 `Cust_DOB` date DEFAULT NULL,

 `Add_ID` int(11) NOT NULL,

 `Cust_Contact` varchar(45) NOT NULL,

 `Login_ID` int(11) DEFAULT NULL,

 `INTERNET_BANKING_Internet_ID` int(11) DEFAULT NULL,

 PRIMARY KEY (`Customer_Id`),

 KEY `fk_CUSTOMER_ADDRESS_CUSTOMER1_idx` (`Add_ID`),

 KEY `fk_LOGIN_ID_idx` (`Login_ID`),

 KEY `fk_CUSTOMER_INTERNET_BANKING1_idx`

(`INTERNET_BANKING_Internet_ID`),

 CONSTRAINT `fk_CUSTOMER_ADDRESS_CUSTOMER1` FOREIGN KEY (`Add_ID`)
REFERENCES `address_customer_employee` (`add_id`),

 CONSTRAINT `fk_CUSTOMER_INTERNET_BANKING1` FOREIGN KEY
(`INTERNET_BANKING_Internet_ID`) REFERENCES `internet_banking` (`internet_id`),

 CONSTRAINT `fk_LOGIN_ID` FOREIGN KEY (`Login_ID`) REFERENCES `login`
(`login_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

```
--  
-- Table structure for table `customer_account_details`  
--
```

```
DROP TABLE IF EXISTS `customer_account_details`;  
/*!40101 SET @saved_cs_client = @@character_set_client */;  
SET character_set_client = utf8mb4 ;  
CREATE TABLE `customer_account_details` (  
  `CUSTOMER_Customer_Id` int(11) NOT NULL,  
  `ACCOUNT_Account_Number` int(11) NOT NULL,  
  KEY `fk_CUSTOMER_has_ACCOUNT_ACCOUNT1_idx`  
  (`ACCOUNT_Account_Number`),  
  KEY `fk_CUSTOMER_has_ACCOUNT_CUSTOMER1_idx` (`CUSTOMER_Customer_Id`),  
  CONSTRAINT `fk_CUSTOMER_has_ACCOUNT_ACCOUNT1` FOREIGN KEY  
  (`ACCOUNT_Account_Number`) REFERENCES `account` (`account_number`),  
  CONSTRAINT `fk_CUSTOMER_has_ACCOUNT_CUSTOMER1` FOREIGN KEY  
  (`CUSTOMER_Customer_Id`) REFERENCES `customer` (`customer_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--  
-- Table structure for table `customer_details_at_atm`  
--
```

```
DROP TABLE IF EXISTS `customer_details_at_atm`;  
/*!40101 SET @saved_cs_client = @@character_set_client */;  
SET character_set_client = utf8mb4 ;  
CREATE TABLE `customer_details_at_atm` (  
  `ATM_ATM_ID` int(11) NOT NULL,
```

```

`CUSTOMER_Customer_Id` int(11) NOT NULL,
KEY `fk_ATM_has_CUSTOMER_CUSTOMER1_idx` (`CUSTOMER_Customer_Id`),
KEY `fk_ATM_has_CUSTOMER_ATM1_idx` (`ATM_ATM_ID`),
CONSTRAINT `fk_ATM_has_CUSTOMER_ATM1` FOREIGN KEY (`ATM_ATM_ID`)
REFERENCES `atm` (`atm_id`),
CONSTRAINT `fk_ATM_has_CUSTOMER_CUSTOMER1` FOREIGN KEY
(`CUSTOMER_Customer_Id`) REFERENCES `customer` (`customer_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `employee`
--

```

```

DROP TABLE IF EXISTS `employee`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
SET character_set_client = utf8mb4 ;
CREATE TABLE `employee` (
  `Employee_ID` int(11) NOT NULL,
  `Employee_Name` varchar(45) NOT NULL,
  `Employee_DOB` date DEFAULT NULL,
  `Employee_Dept` varchar(45) NOT NULL,
  `Employee_EMAIL` varchar(45) NOT NULL,
  `BRANCH_Branch_ID` int(11) NOT NULL,
  `ADDRESS_CUSTOMER_Add_ID` int(11) NOT NULL,
  PRIMARY KEY (`Employee_ID`),
  KEY `fk_EMPLOYEE_BRANCH1_idx` (`BRANCH_Branch_ID`),
  KEY `fk_EMPLOYEE_ADDRESS_CUSTOMER1_idx`
  (`ADDRESS_CUSTOMER_Add_ID`),

```

```

    CONSTRAINT `fk_EMPLOYEE_ADDRESS_CUSTOMER1` FOREIGN KEY
(`ADDRESS_CUSTOMER_Add_ID`) REFERENCES `address_customer_employee`
(`add_id`),

    CONSTRAINT `fk_EMPLOYEE_BRANCH1` FOREIGN KEY (`BRANCH_Branch_ID`)
REFERENCES `branch` (`branch_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `internet_banking`
--

DROP TABLE IF EXISTS `internet_banking`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `internet_banking` (
  `Internet_ID` int(11) NOT NULL,
  `Login_ID` int(11) DEFAULT NULL,
  PRIMARY KEY (`Internet_ID`),
  KEY `fk_INTERNET_BANKING_LOGIN1_idx` (`Login_ID`),
  CONSTRAINT `fk_INTERNET_BANKING_LOGIN1` FOREIGN KEY (`Login_ID`)
REFERENCES `login` (`login_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `loan`
--

DROP TABLE IF EXISTS `loan`;

```

```

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `loan` (
  `Loan_Number` int(11) NOT NULL,
  `Loan_Amount` int(11) NOT NULL,
  `LOAN_TYPE_Loan_Type_Code` int(11) NOT NULL,
  `CUSTOMER_Customer_Id` int(11) DEFAULT NULL,
  `Amount_Paid` int(11) DEFAULT NULL,
  `Notification` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`Loan_Number`),
  KEY `fk_LOAN_LOAN_TYPE1_idx` (`LOAN_TYPE_Loan_Type_Code`),
  CONSTRAINT `fk_LOAN_LOAN_TYPE1` FOREIGN KEY
  (`LOAN_TYPE_Loan_Type_Code`) REFERENCES `loan_type` (`loan_type_code`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `loan_type`
--

```

```

DROP TABLE IF EXISTS `loan_type`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `loan_type` (
  `Loan_Type_Code` int(11) NOT NULL,
  `Loan_Type_Description` varchar(45) NOT NULL,
  `Interest_Rate` double NOT NULL,
  `EMPLOYEE_Employee_ID` int(11) NOT NULL,
  PRIMARY KEY (`Loan_Type_Code`),

```

```

    KEY `fk_Employee_ID` (`EMPLOYEE_Employee_ID`),

    CONSTRAINT `fk_Employee_ID` FOREIGN KEY (`EMPLOYEE_Employee_ID`)
REFERENCES `employee` (`employee_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Table structure for table `loan_types_by_branch`

--

DROP TABLE IF EXISTS `loan_types_by_branch`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `loan_types_by_branch` (

  `BRANCH_Branch_ID` int(11) NOT NULL,

  `LOAN_TYPE_Loan_Type_Code` int(11) NOT NULL,

  KEY `fk_BRANCH_has_LOAN_TYPE_LOAN_TYPE1_idx`

(`LOAN_TYPE_Loan_Type_Code`),

  KEY `fk_BRANCH_has_LOAN_TYPE_BRANCH1_idx` (`BRANCH_Branch_ID`),

  CONSTRAINT `fk_BRANCH_has_LOAN_TYPE_BRANCH1` FOREIGN KEY

(`BRANCH_Branch_ID`) REFERENCES `branch` (`branch_id`),

  CONSTRAINT `fk_BRANCH_has_LOAN_TYPE_LOAN_TYPE1` FOREIGN KEY

(`LOAN_TYPE_Loan_Type_Code`) REFERENCES `loan_type` (`loan_type_code`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Table structure for table `login`

--

```



```

DROP TABLE IF EXISTS `login`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `login` (
  `Login_ID` int(11) NOT NULL,
  `Username` varchar(45) NOT NULL,
  `Password` varchar(45) NOT NULL,
  PRIMARY KEY (`Login_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `transaction`
--

DROP TABLE IF EXISTS `transaction`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `transaction` (
  `Transaction_ID` int(11) NOT NULL,
  `Amount` int(11) DEFAULT NULL,
  `Transaction_Date_Time` datetime NOT NULL,
  `Other_Details` varchar(45) NOT NULL,
  `TRANSACTION_TYPES_Transaction_Type_Code` int(11) NOT NULL,
  `ACCOUNT_Account_Number` int(11) NOT NULL,
  `Message` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`Transaction_ID`),
  KEY `fk_TRANSACTION_TRANSACTION_TYPES1_idx`
  (`TRANSACTION_TYPES_Transaction_Type_Code`),

```

```

KEY `fk_TRANSACTION_ACCOUNT1_idx` (`ACCOUNT_Account_Number`),
CONSTRAINT `fk_TRANSACTION_ACCOUNT1` FOREIGN KEY
(`ACCOUNT_Account_Number`) REFERENCES `account` (`account_number`),
CONSTRAINT `fk_TRANSACTION_TRANSACTION_TYPES1` FOREIGN KEY
(`TRANSACTION_TYPES_Transaction_Type_Code`) REFERENCES `transaction_types`
(`transaction_type_code`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

/*!50003 SET @saved_cs_client      = @@character_set_client */;
/*!50003 SET @saved_cs_results    = @@character_set_results */;
/*!50003 SET @saved_col_connection = @@collation_connection */;
/*!50003 SET character_set_client = utf8mb4 */;
/*!50003 SET character_set_results = utf8mb4 */;
/*!50003 SET collation_connection = utf8mb4_0900_ai_ci */;
/*!50003 SET @saved_sql_mode      = @@sql_mode */;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 TRIGGER `Bal_upd`
AFTER INSERT ON `transaction` FOR EACH ROW BEGIN

    declare type int;

    declare amount int;

    declare account_num int;

    declare type_desc varchar(45);

    declare balance int;


    SELECT tt.Transaction_Type_Code, t.Amount,
acc.Account_Number,tt.Transaction_Type_Description,acc.account_balance
    into type,amount,account_num,type_desc,balance
    from TRANSACTION_TYPES tt, Account acc, transaction t

```

```
where t.TRANSACTION_TYPES_Transaction_Type_Code=tt.Transaction_Type_Code
and t.ACCOUNT_Account_Number=acc.Account_Number
AND acc.Account_Number=6000;
```

```
IF type = 2001 then
```

```
    Update acc_temp set balance=balance+amount
    where acc_num=account_num;
```

```
else
```

```
    Update acc_temp set balance=amount-balance
    where acc_num=account_num;
```

```
end if;
```

```
END */;;
```

```
DELIMITER ;
```

```
/*!50003 SET sql_mode          = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 SET @saved_cs_client     = @@character_set_client */ ;
/*!50003 SET @saved_cs_results   = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client = utf8mb4 */ ;
```

```

/*!50003 SET character_set_results = utf8mb4 */ ;

/*!50003 SET collation_connection = utf8mb4_0900_ai_ci */ ;

/*!50003 SET @saved_sql_mode      = @@sql_mode */ ;

/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 TRIGGER `trial`
AFTER UPDATE ON `transaction` FOR EACH ROW BEGIN

    Insert into acc_temp

values(null,old.amount+new.amount,old.amount,new.amount,new.account_account_number,new
.amount);

end */;;

DELIMITER ;

/*!50003 SET sql_mode              = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

/*!50003 SET @saved_cs_client      = @@character_set_client */ ;

/*!50003 SET @saved_cs_results     = @@character_set_results */ ;

/*!50003 SET @saved_col_connection = @@collation_connection */ ;

/*!50003 SET character_set_client = utf8mb4 */ ;

/*!50003 SET character_set_results = utf8mb4 */ ;

/*!50003 SET collation_connection = utf8mb4_0900_ai_ci */ ;

/*!50003 SET @saved_sql_mode      = @@sql_mode */ ;

/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 TRIGGER
`update_accbalance_trg` AFTER UPDATE ON `transaction` FOR EACH ROW BEGIN

```

```

declare acc_bal int;

        select account_balance
into acc_bal
from account
where account_number=new.account_account_number;

update account set account_balance=acc_bal+new.amount
where account_number=new.account_account_number;

end */;;
DELIMITER ;

/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;

--
-- Table structure for table `transaction_internet_banking`
--

DROP TABLE IF EXISTS `transaction_internet_banking`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
SET character_set_client = utf8mb4 ;
CREATE TABLE `transaction_internet_banking` (
  `Trx_Internet_ID` int(11) NOT NULL,
  `INTERNET_BANKING_Internet_ID` int(11) DEFAULT NULL,
  `TRANSACTION_Transaction_ID` int(11) DEFAULT NULL,

```

```

    KEY `fk_TRANSACTION_INTERNET_BANKING_INTERNET_BANKING1`
(`INTERNET_BANKING_Internet_ID`),

    KEY `fk_TRANSACTION_INTERNET_BANKING_TRANSACTION1`
(`TRANSACTION_Transaction_ID`),

    CONSTRAINT `fk_TRANSACTION_INTERNET_BANKING_INTERNET_BANKING1`
FOREIGN KEY (`INTERNET_BANKING_Internet_ID`) REFERENCES `internet_banking`
(`internet_id`),

    CONSTRAINT `fk_TRANSACTION_INTERNET_BANKING_TRANSACTION1`
FOREIGN KEY (`TRANSACTION_Transaction_ID`) REFERENCES `transaction`
(`transaction_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Table structure for table `transaction_types`

--

DROP TABLE IF EXISTS `transaction_types`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `transaction_types` (

    `Transaction_Type_Code` int(11) NOT NULL,

    `Transaction_Type_Description` varchar(45) NOT NULL,

    PRIMARY KEY (`Transaction_Type_Code`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Table structure for table `transactions_at_atm`

--

```

```

DROP TABLE IF EXISTS `transactions_at_atm`;

/*!40101 SET @saved_cs_client      = @@character_set_client */;

SET character_set_client = utf8mb4 ;

CREATE TABLE `transactions_at_atm` (
  `ATM_ATM_ID` int(11) NOT NULL,
  `TRANSACTION_Transaction_ID` int(11) NOT NULL,
  KEY `fk_ATM_has_TRANSACTION_TRANSACTION1_idx`
  (`TRANSACTION_Transaction_ID`),
  KEY `fk_ATM_has_TRANSACTION_ATM1_idx` (`ATM_ATM_ID`),
  CONSTRAINT `fk_ATM_has_TRANSACTION_ATM1` FOREIGN KEY (`ATM_ATM_ID`)
  REFERENCES `atm` (`atm_id`),
  CONSTRAINT `fk_ATM_has_TRANSACTION_TRANSACTION1` FOREIGN KEY
  (`TRANSACTION_Transaction_ID`) REFERENCES `transaction` (`transaction_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping events for database 'nigamaman'

--

--

-- Dumping routines for database 'nigamaman'

--

/*!50003 DROP FUNCTION IF EXISTS `loan_check` */;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results     = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client  = utf8mb4 */;

```

```

/*!50003 SET character_set_results = utf8mb4 */ ;

/*!50003 SET collation_connection = utf8mb4_0900_ai_ci */ ;

/*!50003 SET @saved_sql_mode      = @@sql_mode */ ;

/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` FUNCTION `loan_check`(loan_amount
int,amount_paid int) RETURNS text CHARSET utf8mb4
BEGIN

    set @cal=loan_amount*(50/100);

    IF amount_paid>@cal then

        return 'You are eligible to apply for another loan';

    else

        return 'First!Please pay atleast 50% of previous loan';

    end if;

END ;;

DELIMITER ;

/*!50003 SET sql_mode              = @saved_sql_mode */ ;

/*!50003 SET character_set_client  = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

/*!50003 DROP PROCEDURE IF EXISTS `Customer_Transaction` */;

/*!50003 SET @saved_cs_client      = @@character_set_client */ ;

/*!50003 SET @saved_cs_results    = @@character_set_results */ ;

/*!50003 SET @saved_col_connection = @@collation_connection */ ;

/*!50003 SET character_set_client  = utf8mb4 */ ;

/*!50003 SET character_set_results = utf8mb4 */ ;

/*!50003 SET collation_connection = utf8mb4_0900_ai_ci */ ;

/*!50003 SET @saved_sql_mode      = @@sql_mode */ ;

```



```

/*!50003 SET sql_mode          =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `Customer_Transaction`(IN acc_num
INT)

BEGIN

select d.CUSTOMER_Customer_Id as "Customer ID", concat_ws(' ',e.Cust_First_Name,
e.Cust_Last_Name) as "Customer Name", c.Account_Number As "A/C Number",
f.Transaction_Type_Description as "Transaction Type", b.Transaction_Id as "Transaction
ID",b.Amount, b. Transaction_Date_Time As "Date", b.Other_Details as "Type"

from TRANSACTION b, account c, customer_account_details d,customer
e,TRANSACTION_TYPES f

where b.TRANSACTION_TYPES_Transaction_Type_Code=f.Transaction_Type_Code

and b.ACCOUNT_Account_number=c.Account_Number

and c.Account_Number=d.ACCOUNT_Account_Number

and d.CUSTOMER_Customer_Id=e.customer_id

and d.CUSTOMER_Customer_Id=e.customer_id

and c.Account_Number=@acc_num;

END ;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */;

/*!50003 SET character_set_client = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

/*!50003 DROP PROCEDURE IF EXISTS `Customer_Transaction_new` */;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results    = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

```

```

/*!50003 SET character_set_client = utf8mb4 */ ;
/*!50003 SET character_set_results = utf8mb4 */ ;
/*!50003 SET collation_connection = utf8mb4_0900_ai_ci */ ;
/*!50003 SET @saved_sql_mode      = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `Customer_Transaction_new` (IN
acc_num INT)

BEGIN

select d.CUSTOMER_Customer_Id as "Customer ID", concat_ws(' ',e.Cust_First_Name,
e.Cust_Last_Name) as "Customer Name", @acc_num:=c.Account_Number As "A/C Number",
f.Transaction_Type_Description as "Transaction Type", @trx_id:= b.Transaction_Id as
"Transaction ID", @amount:=b.Amount, b. Transaction_Date_Time As "Date", b.Other_Details
as "Type"

from TRANSACTION b, account c, customer_account_details d,customer
e,TRANSACTION_TYPES f

where b.TRANSACTION_TYPES_Transaction_Type_Code=f.Transaction_Type_Code

and b.ACCOUNT_Account_number=c.Account_Number

and c.Account_Number=d.ACCOUNT_Account_Number

and d.CUSTOMER_Customer_Id=e.customer_id

and d.CUSTOMER_Customer_Id=e.customer_id

and b.Transaction_Date_Time<='2018-12-11 21:52:24' and b.Transaction_Date_Time>'2018-12-
11 17:42:17'

and c.Account_Number=acc_num;

If @amount > 400 then

        update TRANSACTION set message='Transaction greater than 400 are not allowed'

        where Transaction_Id=@trx_id;

else

```

```

        update account set message='Your recent transaction was successful'
        where Transaction_Id=@trx_id;
end if;
END ;;
DELIMITER ;

/*!50003 SET sql_mode            = @saved_sql_mode */;
/*!50003 SET character_set_client = @saved_cs_client */;
/*!50003 SET character_set_results = @saved_cs_results */;
/*!50003 SET collation_connection = @saved_col_connection */;
/*!50003 DROP PROCEDURE IF EXISTS `loan_amt_paid_func` */;
/*!50003 SET @saved_cs_client      = @@character_set_client */;
/*!50003 SET @saved_cs_results     = @@character_set_results */;
/*!50003 SET @saved_col_connection = @@collation_connection */;
/*!50003 SET character_set_client   = utf8mb4 */;
/*!50003 SET character_set_results = utf8mb4 */;
/*!50003 SET collation_connection   = utf8mb4_0900_ai_ci */;
/*!50003 SET @saved_sql_mode       = @@sql_mode */;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */;
DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `loan_amt_paid_func`(IN cust_id
INT,IN pay_amt INT)

BEGIN

        select l.loan_number, @loan_amount:=l.loan_amount, c.customer_id,
        @amt_paid:=l.amount_paid
        from customer c, loan l
        where c.customer_id=l.CUSTOMER_Customer_id
        and l.CUSTOMER_Customer_id=cust_id;

```

```
update loan set amount_paid=(pay_amt+@amt_paid)
```

```
where CUSTOMER_Customer_id=cust_id;
```

```
select @new_amt_paid:=l.amount_paid
```

```
from customer c, loan l
```

```
where c.customer_id=l.CUSTOMER_Customer_id
```

```
and l.CUSTOMER_Customer_id=cust_id;
```

```
select @notification:=loan_check(@loan_amount,@new_amt_paid);
```

```
update loan set Notification=@notification
```

```
where CUSTOMER_Customer_id=cust_id;
```

```
END ;;
```

```
DELIMITER ;
```

```
/*!50003 SET sql_mode          = @saved_sql_mode */;
```

```
/*!50003 SET character_set_client = @saved_cs_client */;
```

```
/*!50003 SET character_set_results = @saved_cs_results */;
```

```
/*!50003 SET collation_connection = @saved_col_connection */;
```

```
/*!50003 DROP PROCEDURE IF EXISTS `new_loan_amt_paid` */;
```

```
/*!50003 SET @saved_cs_client      = @@character_set_client */;
```

```
/*!50003 SET @saved_cs_results     = @@character_set_results */;
```

```
/*!50003 SET @saved_col_connection = @@collation_connection */;
```

```
/*!50003 SET character_set_client  = utf8mb4 */;
```

```
/*!50003 SET character_set_results = utf8mb4 */;
```

```
/*!50003 SET collation_connection  = utf8mb4_0900_ai_ci */;
```

```
/*!50003 SET @saved_sql_mode       = @@sql_mode */;
```

```
/*!50003 SET sql_mode              =  
'STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION' */;
```

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `new_loan_amt_paid`(IN cust_id INT,IN pay_amt INT)

BEGIN

select l.loan_number, l.loan_amount, c.customer_id, @amt_paid:=l.amount_paid

from customer c, loan l

where c.customer_id=l.CUSTOMER_Customer_id

and l.CUSTOMER_Customer_id=cust_id;

update loan set amount_paid=(pay_amt+@amt_paid)

where CUSTOMER_Customer_id=cust_id;

END ;;

DELIMITER ;

/*!50003 SET sql_mode = @saved_sql_mode */;

/*!50003 SET character_set_client = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

--

-- Final view structure for view `auto_loan_details_customer`

--

/*!50001 DROP VIEW IF EXISTS `auto_loan_details_customer`*/;

/*!50001 SET @saved_cs_client = @@character_set_client */;

/*!50001 SET @saved_cs_results = @@character_set_results */;

/*!50001 SET @saved_col_connection = @@collation_connection */;

/*!50001 SET character_set_client = utf8mb4 */;

/*!50001 SET character_set_results = utf8mb4 */;

```

/*!50001 SET collation_connection    = utf8mb4_0900_ai_ci */;

/*!50001 CREATE ALGORITHM=UNDEFINED */

/*!50013 DEFINER=`root`@`localhost` SQL SECURITY DEFINER */

/*!50001 VIEW `auto_loan_details_customer` AS select `b`.`Branch_Name` AS `Branch
Name`,`a`.`Account_Number` AS `A/C Number`,concat_ws('
`,`c`.`Cust_First_Name`,`c`.`Cust_Last_Name`) AS `Customer Name`,`c`.`Cust_Contact` AS
`Contact`,`l3`.`Loan_Number` AS `Loan Number`,`l2`.`Loan_Type_Description` AS `Loan
Type`,`l2`.`Interest_Rate` AS `Interest Rate`,`l3`.`Loan_Amount` AS `Loan
Amount`,`l2`.`Loan_Type_Code` AS `Loan Type Code`,`e`.`Employee_Name` AS `Employee
Name` from ((((((`employee` `e` join `branch` `b`) join `loan_types_by_branch` `l1`) join
`loan_type` `l2`) join `loan` `l3`) join `customer` `c`) join `customer_account_details` `cad`) join
`account` `a`) where ((`l3`.`CUSTOMER_Customer_Id` = `c`.`Customer_Id`) and
(`c`.`Customer_Id` = `cad`.`CUSTOMER_Customer_Id`) and
(`cad`.`ACCOUNT_Account_Number` = `a`.`Account_Number`) and
(`l2`.`EMPLOYEE_Employee_ID` = `e`.`Employee_ID`) and (`b`.`Branch_ID` =
`l1`.`BRANCH_Branch_ID`) and (`l1`.`LOAN_TYPE_Loan_Type_Code` =
`l2`.`Loan_Type_Code`) and (`l2`.`Loan_Type_Code` =
`l3`.`LOAN_TYPE_Loan_Type_Code`) and (`b`.`Branch_ID` = 201) and
(`l2`.`Loan_Type_Description` = 'Auto Loans')) */;

/*!50001 SET character_set_client    = @saved_cs_client */;

/*!50001 SET character_set_results    = @saved_cs_results */;

/*!50001 SET collation_connection    = @saved_col_connection */;

/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;

/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;

/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

```

INSERT STATEMENTS

-- MySQL dump 10.13 Distrib 8.0.12, for Win64 (x86_64)

--

-- Host: localhost Database: nigramaman

-- Server version 8.0.12

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

SET NAMES utf8 ;

/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;

/*!40103 SET TIME_ZONE='+00:00' */;

/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;

/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;

/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;

/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--

-- Dumping data for table `acc_temp`

--

LOCK TABLES `acc_temp` WRITE;

```

/*!40000 ALTER TABLE `acc_temp` DISABLE KEYS */;

INSERT INTO `acc_temp` VALUES
(NULL,NULL,NULL,NULL,1596,NULL),(NULL,NULL,NULL,NULL,5000,NULL),(NULL,1
700,NULL,NULL,6000,NULL),(NULL,1200,1000,200,6000,200),(NULL,600,200,400,6000,40
0),(NULL,1200,800,400,5000,400),(NULL,1250,250,1000,1596,1000);

/*!40000 ALTER TABLE `acc_temp` ENABLE KEYS */;

UNLOCK TABLES;


--
-- Dumping data for table `account`
--


LOCK TABLES `account` WRITE;

/*!40000 ALTER TABLE `account` DISABLE KEYS */;

INSERT INTO `account` VALUES (1596,1000,'2018-12-04',301),(5000,920,'2018-12-
12',301),(5555,0,'2018-12-11',301),(6000,1100,'2018-12-12',301),(6587,0,'2018-12-
11',301),(7415,0,'2018-12-11',301),(9995,0,'2018-12-11',301);

/*!40000 ALTER TABLE `account` ENABLE KEYS */;

UNLOCK TABLES;


--
-- Dumping data for table `account_types`
--


LOCK TABLES `account_types` WRITE;

/*!40000 ALTER TABLE `account_types` DISABLE KEYS */;

INSERT INTO `account_types` VALUES (301,'Savings Account'),(302,'Checkings Account');

/*!40000 ALTER TABLE `account_types` ENABLE KEYS */;

UNLOCK TABLES;

```



```
--  
-- Dumping data for table `account_types_by_branch`  
--
```

```
LOCK TABLES `account_types_by_branch` WRITE;  
/*!40000 ALTER TABLE `account_types_by_branch` DISABLE KEYS */;  
INSERT INTO `account_types_by_branch` VALUES  
(201,301),(201,302),(202,301),(202,302),(208,301),(208,302);  
/*!40000 ALTER TABLE `account_types_by_branch` ENABLE KEYS */;  
UNLOCK TABLES;
```

```
--  
-- Dumping data for table `accounts_at_atm`  
--
```

```
LOCK TABLES `accounts_at_atm` WRITE;  
/*!40000 ALTER TABLE `accounts_at_atm` DISABLE KEYS */;  
INSERT INTO `accounts_at_atm` VALUES (1596),(9995);  
/*!40000 ALTER TABLE `accounts_at_atm` ENABLE KEYS */;  
UNLOCK TABLES;
```

```
--  
-- Dumping data for table `address_atm_and_branch`  
--
```

```
LOCK TABLES `address_atm_and_branch` WRITE;  
/*!40000 ALTER TABLE `address_atm_and_branch` DISABLE KEYS */;  
INSERT INTO `address_atm_and_branch` VALUES (101,'30 Tremont  
Street',NULL,'Boston','MA','USA','Branch Address'),(102,'35 Cambridgepark
```

```
Dr',NULL,'Boston','MA','USA','Branch Address'),(108,'305 Boylston
Street',NULL,'Boston','MA','USA','Branch Address'),(110,'309 Boylston
Street',NULL,'Boston','MA','USA','ATM Address');
```

```
/*!40000 ALTER TABLE `address_atm_and_branch` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `address_customer_employee`
```

```
--
```

```
LOCK TABLES `address_customer_employee` WRITE;
```

```
/*!40000 ALTER TABLE `address_customer_employee` DISABLE KEYS */;
```

```
INSERT INTO `address_customer_employee` VALUES (401,'1012 Huntington
Avenue',NULL,'Boston','MA','USA','Employee address'),(402,'115 Mission
Main',NULL,'Boston','MA','USA','Employee address'),(403,'410 Boylston
Street',NULL,'Boston','MA','USA','Employee address'),(404,'479 Boylston
Street',NULL,'Boston','MA','USA','Customer address'),(405,'40 Parker
Hill',NULL,'Boston','MA','USA','Customer address'),(406,'1196 St Alphonsus
Street',NULL,'Boston','MA','USA','Customer address'),(407,'1140 Huntington
Ave',NULL,'Boston','MA','USA','Customer address'),(408,'853 Mission
Hill',NULL,'Boston','MA','USA','Customer address');
```

```
/*!40000 ALTER TABLE `address_customer_employee` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `atm`
```

```
--
```

```
LOCK TABLES `atm` WRITE;
```

```
/*!40000 ALTER TABLE `atm` DISABLE KEYS */;
```

```
INSERT INTO `atm` VALUES
(1001,110,NULL),(1002,110,1596),(1003,110,9995),(1004,110,9995);
```

```
/*!40000 ALTER TABLE `atm` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `bank`
```

```
--
```

```
LOCK TABLES `bank` WRITE;
```

```
/*!40000 ALTER TABLE `bank` DISABLE KEYS */;
```

```
INSERT INTO `bank` VALUES ('State Bank of Massachusetts');
```

```
/*!40000 ALTER TABLE `bank` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `branch`
```

```
--
```

```
LOCK TABLES `branch` WRITE;
```

```
/*!40000 ALTER TABLE `branch` DISABLE KEYS */;
```

```
INSERT INTO `branch` VALUES (201,'Tremont Street branch','A201',101,'State Bank of  
Massachusetts',NULL),(202,'Cambridge Park branch','A202',102,'State Bank of  
Massachusetts',NULL),(208,'Boylston Street branch','A208',108,'State Bank of  
Massachusetts',NULL);
```

```
/*!40000 ALTER TABLE `branch` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `customer`
```

```
--
```

```
LOCK TABLES `customer` WRITE;
```

```
/*!40000 ALTER TABLE `customer` DISABLE KEYS */;
```

```
INSERT INTO `customer` VALUES (901,'Samuel','Fernandez','1990-12-19',404,'2017417896',NULL,701),(902,'Casey','Williams','1994-10-29',405,'8567411896',801,702),(903,'Fredrick','D\'Souza','1989-10-19',406,'8159711896',NULL,NULL),(904,'Dion','Nash','1985-11-25',407,'8156489657',NULL,NULL),(905,'Shane','Bond','1984-12-21',407,'8156159632',NULL,NULL);
```

```
/*!40000 ALTER TABLE `customer` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `customer_account_details`
```

```
--
```

```
LOCK TABLES `customer_account_details` WRITE;
```

```
/*!40000 ALTER TABLE `customer_account_details` DISABLE KEYS */;
```

```
INSERT INTO `customer_account_details` VALUES  
(901,1596),(902,9995),(903,5555),(904,6587),(905,7415);
```

```
/*!40000 ALTER TABLE `customer_account_details` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `customer_details_at_atm`
```

```
--
```

```
LOCK TABLES `customer_details_at_atm` WRITE;
```

```
/*!40000 ALTER TABLE `customer_details_at_atm` DISABLE KEYS */;
```

```
INSERT INTO `customer_details_at_atm` VALUES (1002,901);
```

```
/*!40000 ALTER TABLE `customer_details_at_atm` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `employee`
```

```
--
```

```
LOCK TABLES `employee` WRITE;
```

```
/*!40000 ALTER TABLE `employee` DISABLE KEYS */;
```

```
INSERT INTO `employee` VALUES (501,'John Johnson','1985-10-24','Accounts  
Department','jooohnson.john@sbm.com',201,401),(502,'Ricky Martin','1983-08-21','Loans  
Department','martin.ricky@sbm.com',201,402),(503,'Peter Parker','1987-05-19','Internet Banking  
Department','parker.peter@sbm.com',201,403);
```

```
/*!40000 ALTER TABLE `employee` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `internet_banking`
```

```
--
```

```
LOCK TABLES `internet_banking` WRITE;
```

```
/*!40000 ALTER TABLE `internet_banking` DISABLE KEYS */;
```

```
INSERT INTO `internet_banking` VALUES (701,NULL),(702,801);
```

```
/*!40000 ALTER TABLE `internet_banking` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `loan`
```

```
--
```

```
LOCK TABLES `loan` WRITE;
```

```
/*!40000 ALTER TABLE `loan` DISABLE KEYS */;
```

```
INSERT INTO `loan` VALUES (10001,15000,602,903,10000,'You are eligible to apply for another loan'),(10002,90000,603,904,NULL,NULL),(10003,5000,601,905,NULL,NULL);
```

```
/*!40000 ALTER TABLE `loan` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `loan_type`
```

```
--
```

```
LOCK TABLES `loan_type` WRITE;
```

```
/*!40000 ALTER TABLE `loan_type` DISABLE KEYS */;
```

```
INSERT INTO `loan_type` VALUES (601,'Auto Loans',4.9,502),(602,'Education Loans',6.6,502),(603,'Home Loans',4.4,502);
```

```
/*!40000 ALTER TABLE `loan_type` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `loan_types_by_branch`
```

```
--
```

```
LOCK TABLES `loan_types_by_branch` WRITE;
```

```
/*!40000 ALTER TABLE `loan_types_by_branch` DISABLE KEYS */;
```

```
INSERT INTO `loan_types_by_branch` VALUES (201,601),(201,602),(201,603),(202,601),(202,602),(202,603),(208,601),(208,602),(208,603);
```

```
/*!40000 ALTER TABLE `loan_types_by_branch` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

--

-- Dumping data for table `login`

--

LOCK TABLES `login` WRITE;

/*!40000 ALTER TABLE `login` DISABLE KEYS */;

INSERT INTO `login` VALUES (801,'casey456','casey@456');

/*!40000 ALTER TABLE `login` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `transaction`

--

LOCK TABLES `transaction` WRITE;

/*!40000 ALTER TABLE `transaction` DISABLE KEYS */;

INSERT INTO `transaction` VALUES (3001,1000,'2018-12-04 15:16:54','Transaction done at ATM',2001,1596,NULL),(3002,450,'2018-12-11 15:11:54','Transaction done at Internet Banking',2001,9995,NULL),(3003,500,'2018-12-11 17:17:17','Transaction done at ATM',2001,9995,NULL),(3004,200,'2018-12-11 17:42:17','Debit of \$200',2002,9995,NULL),(3005,600,'2018-12-11 21:52:24','Transaction done at ATM',2001,9995,'Transaction greater than 400 are not allowed'),(3006,400,'2018-12-12 16:49:17','Transaction done at ATM',2001,5000,NULL),(3007,400,'2018-12-12 16:51:34','Transaction done at ATM',2001,6000,NULL);

/*!40000 ALTER TABLE `transaction` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `transaction_internet_banking`

--

```
LOCK TABLES `transaction_internet_banking` WRITE;
/*!40000 ALTER TABLE `transaction_internet_banking` DISABLE KEYS */;
INSERT INTO `transaction_internet_banking` VALUES (4001,702,3002);
/*!40000 ALTER TABLE `transaction_internet_banking` ENABLE KEYS */;
UNLOCK TABLES;
```

```
--
-- Dumping data for table `transaction_types`
--
```

```
LOCK TABLES `transaction_types` WRITE;
/*!40000 ALTER TABLE `transaction_types` DISABLE KEYS */;
INSERT INTO `transaction_types` VALUES (2001,'CREDIT'),(2002,'DEBIT');
/*!40000 ALTER TABLE `transaction_types` ENABLE KEYS */;
UNLOCK TABLES;
```

```
--
-- Dumping data for table `transactions_at_atm`
--
```

```
LOCK TABLES `transactions_at_atm` WRITE;
/*!40000 ALTER TABLE `transactions_at_atm` DISABLE KEYS */;
INSERT INTO `transactions_at_atm` VALUES (1002,3001),(1003,3003),(1004,3004);
/*!40000 ALTER TABLE `transactions_at_atm` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;
```



```
/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;  
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;  
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;  
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
```