Ahsanullah University of Science & Technology

Department of Computer Science & Engineering

Project Name
Bank Management System

Database Lab
(CSE 3104)

PROJECT OBJECTIVE:

The main objective of 'BANK MANAGEMENT SYSTEM' is to prepare a

software or application which could maintain data of customer and provide a user friendly interface for retrieving customer related details just in few seconds with 100% accuracy. Software is completely computerized so we can update information instantly without any paperwork and as well it is not time consuming.

FEATURES OF THE PROJECT:

- Store customer information with branch specification
- Update balance after withdraw or deposit of money
- Search customer by firstname starting alphabet or date of birth or address or branch etc.
- Show balance sheet with customer details

TYPES OF USER:

CUSTOMER:

They can see their information and balance sheet.

ADMINISTRATOR:

Can modify queries to change information or to add future features of the system.

BANK MANAGEMENT:

Bank employees or manager can use this to edit or insert of customer and transation details.

FEATURE GROUPING ACCORDING TO USER:

Customer : Customer personal information,current balance,transaction

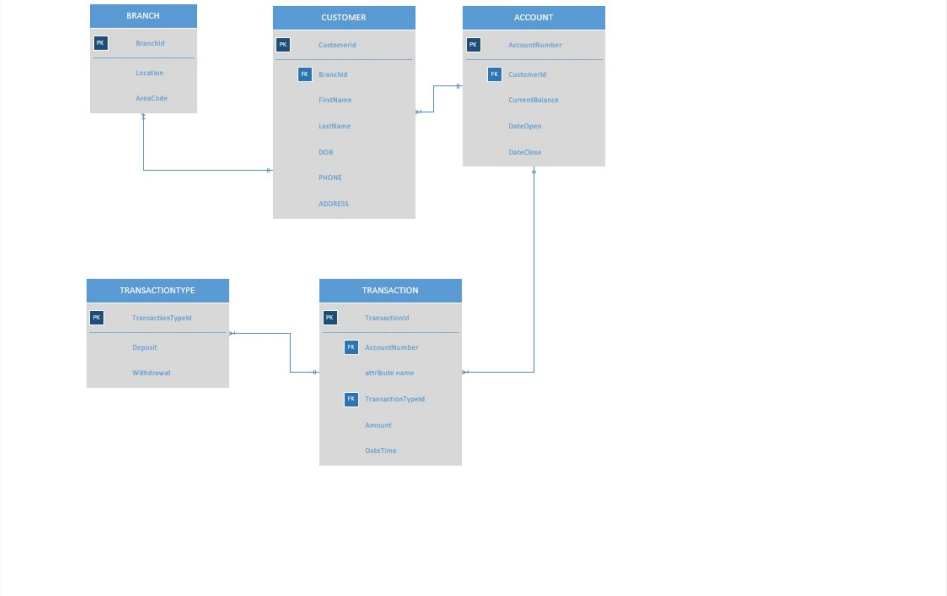
type,transaction date

Bank Management : Edit customer info,transactions of each customer,customers of specific branch

NAMES OF THE ENTITIES WITH PRIMARY KEY:

ENTITY	**:**	**PRIMARY KEY**
BRANCH	:	BranchId
CUSTOMER	:	CustomerId
ACCOUNT	:	AccountNumber
TRANSACTIONS	:	TransactionsId
TRANSACTIONTYPE	:	TransactiontypeId

ER DIAGRAM:



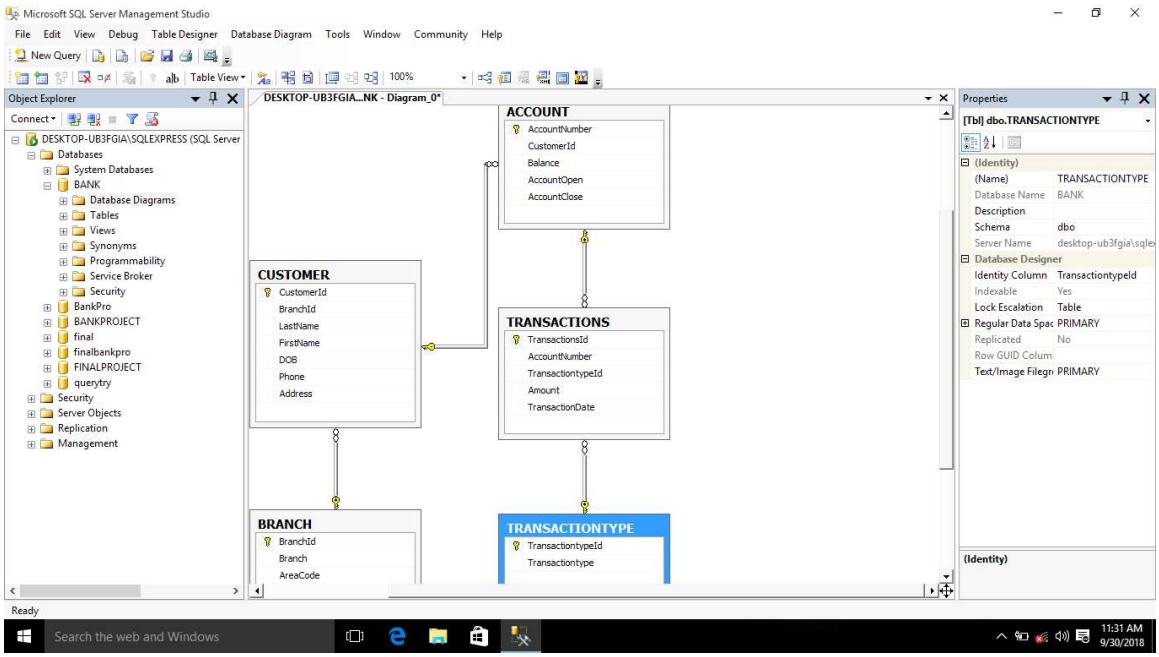
RELATIONAL MODEL:

Create table command

```
CREATE TABLE BRANCH(
BranchId int IDENTITY(1,1) PRIMARY KEY,
Branch varchar (8) not null,
AreaCode int not null,
)
CREATE TABLE CUSTOMER(
CustomerId int IDENTITY(1,1) PRIMARY KEY,
BranchId int not null FOREIGN KEY REFERENCES BRANCH(BranchId),
LastName varchar(20) not null,
FirstName varchar(20) not null,
DOB DATE NOT NULL,
Phone varchar(11) not null,
Address varchar(200) not null,
)
CREATE TABLE ACCOUNT(
AccountNumber int IDENTITY(1000,1) PRIMARY KEY,
CustomerId int not null FOREIGN KEY REFERENCES CUSTOMER(CustomerId),
Balance numeric(11,2),
AccountOpen DATE NOT NULL,
AccountClose Date,
)
CREATE TABLE TRANSACTIONTYPE(
TransactiontypeId int IDENTITY(1,1) PRIMARY KEY,
```

```
Transactiontype varchar(20),

)

CREATE TABLE TRANSACTIONS(

TransactionsId int IDENTITY(1,1) PRIMARY KEY,

AccountNumber int not null FOREIGN KEY REFERENCES
ACCOUNT(AccountNumber),

TransactiontypeId int not null FOREIGN KEY REFERENCES
TRANSACTIONTYPE(TransactiontypeId),

Amount numeric(9,2),

TransactionDate DATE,

)
```

DATABASE DIAGRAM:



SQL QUERIES GROUPED UNDER DIFFERENT TYPES OF USER:

Customer:

```
-->CUSTOMERS WHOSE AGE GREATER THAN 18

SELECT * FROM CUSTOMER WHERE DATEDIFF(YEAR,DOB,GETDATE())>18


-->CUSTOMER LIST ORDER BY FIRST NAME AND ADDRESS

SELECT * FROM CUSTOMER ORDER BY FirstName,Address


-->TOP 3 RECORDS OF CUSTOMER TABLE

SELECT TOP 3* FROM CUSTOMER


-->SEARCH CUSTOMER BY FIRST NAME OR ADDRESS AND PHONE

SELECT * FROM CUSTOMER WHERE FirstName LIKE 'M%' AND Phone LIKE'09%' OR
Address='GULSHAN'


-->CUSTOMER BALANCESHEET

SELECT
ACCOUNT.AccountNumber,CustomerId,Balance,TRANSACTIONS.TransactionDate,Am
ount,TRANSACTIONTYPE.Transactiontype

FROM ACCOUNT INNER JOIN TRANSACTIONS ON
ACCOUNT.AccountNumber=TRANSACTIONS.AccountNumber

INNER JOIN TRANSACTIONTYPE ON
TRANSACTIONS.TransactionsId=Transactiontype.TransactiontypeId WHERE
CustomerId=2


-->customer number at distinct area

SELECT COUNT(CustomerId) as 'NUMBER OF CUSTOMER',Address FROM CUSTOMER
GROUP BY Address


-->FIRST NAME STARTING WITH A OF CUSTOMER HAVING MAX BALANCE
```

```
SELECT FirstName,MAX(Balance)AS 'MAX BALANCE' FROM CUSTOMER,ACCOUNT

WHERE CUSTOMER.CustomerId=ACCOUNT.CustomerId GROUP BY FirstName HAVING
FirstName like'A%'


-->CUSTOMER WHO DOES NOT LIVE IN DHANMONDI AND DOB NOT IN 2001

SELECT * FROM CUSTOMER WHERE Address<>'DHANMONDI'

UNION SELECT * FROM CUSTOMER WHERE DOB NOT LIKE ('2001%')


-->NAME AND DAY OF DOB FROM CUSTOMER USING SCALAR FUNCTION

SELECT CUSTOMER.CustomerId,UPPER(FirstName)+'    '+LOWER(LastName) as
'NAME',DATEPART(DAY,DOB) AS'DOB',

LEFT(Branch,3)as 'BRANCH'  FROM CUSTOMER INNER JOIN BRANCH ON
CUSTOMER.BranchId=Branch.BranchId


-->NUMBER OF  DISTINCT PLACES

SELECT COUNT(DISTINCT Address) FROM CUSTOMER
```

Bank Management:

```
--> UPDATE ACCOUNT AFTER WITHDRAWAL

UPDATE ACCOUNT SET Balance=(Balance-4000.00) WHERE Balance>4000.00 and
AccountNumber='1008'


--> UPDATE ACCOUNT AFTER DEPOSIT

UPDATE ACCOUNT SET Balance=(Balance+4000.00) WHERE  AccountNumber='1008'


-->CUSTOMER AND BRANCH DETAILS

SELECT
CUSTOMER.CustomerId,LastName,FirstName,DOB,Phone,Address,Branch,AreaCode
```

```
FROM BRANCH INNER JOIN CUSTOMER ON BRANCH.BranchId=CUSTOMER.BranchId


-->select customer who have opened account on 2015 and have account in
mirpur branch

SELECT CUSTOMER.CustomerId,LastName,FirstName,DOB,Phone,Address FROM
BRANCH INNER JOIN CUSTOMER ON BRANCH.BranchId=CUSTOMER.BranchId

inner join ACCOUNT on CUSTOMER.CustomerId=ACCOUNT.CustomerId WHERE
Branch='MIRPUR' AND AccountOpen IN(SELECT AccountOpen FROM ACCOUNT WHERE
(YEAR(AccountOpen)=2015))


-->CUSTOMER SEARCH and show balance

SELECT
CUSTOMER.CustomerId,BranchId,LastName,FirstName,DOB,Phone,Address,ACCOUN
T.AccountNumber,Balance,AccountOpen

 FROM CUSTOMER inner join ACCOUNT on
CUSTOMER.CustomerId=ACCOUNT.CustomerId WHERE FirstName LIKE 'J%'


-->SHOW CUSTOMERS WHO HAVE ACCOUNT IN UTTARA

SELECT CUSTOMER.CustomerId,LastName,FirstName,DOB,Phone,Address FROM
BRANCH INNER JOIN CUSTOMER ON BRANCH.BranchId=CUSTOMER.BranchId WHERE
BRANCH='UTTARA'


-->CUSTOMER WHO HAVE ACCOUNT IN DELHI OR BIRTHDAY IN DECEMBER

SELECT CUSTOMER.CustomerId,LastName,FirstName,Phone,Address

FROM BRANCH INNER JOIN CUSTOMER ON BRANCH.BranchId=CUSTOMER.BranchId
WHERE Branch='DELHI' OR DOB LIKE '_____12___'


--> NUMBER OF CUSTOMER WHO HAVE WITHDRAW MONEY IN 2018

SELECT COUNT(TransactionsId) FROM TRANSACTIONS INNER JOIN
TRANSACTIONTYPE ON
TRANSACTIONS.TransactiontypeId=TRANSACTIONTYPE.TransactiontypeId

WHERE Transactiontype='WITHDRAW' AND TransactionDate LIKE ('2018%')
```

```
-->CUSTOMERS IN DISTINCT BRANCH

SELECT DISTINCT(BranchId),COUNT(CustomerId)as'NO. OF CUSTOMER IN BRANCH'
from CUSTOMER GROUP BY BranchId


-->customer who have balance more than 9000

SELECT * FROM CUSTOMER WHERE CustomerId>ANY(SELECT CustomerId FROM
ACCOUNT WHERE Balance>'9000.00')
```

PROJECT LIMITATIONS:

- Account balance after transaction of both table are not modified,so all transactions cant be get in balance sheet

CONCLUSION AND FUTURE WORK:

This system has been made focusing on easy data retrieving,modify data without waste of time and paper and also transaction record in short time.The data are stored in manual process.It can also be of great help for such banks who have many branches,therefore they can access information of customers from any where which will ease the customer to transacts from any of the branch of same bank.

- There will be added features for loan or transfer
- Customer will be able to open bank in many branches
- All transaction details of a customer can be seen in balancesheet