**DBMS PROJECT**

**By**

**Dileep Gudena**

Banking System

**Case Study**

A bank has a branch which provides loans. Bank has employees and customers. Employees deal with the customers and these customers borrow loan from the branch and pay it back in instalments respectively. Each customer has an account where the money transactions are done using this account by the branch and customer as well. Manager is an employee who supervises other employees.

**Entity Relationship Modelling**

**Step 1:**  **Identify the Entities**

Branch

Loan

Employee

Cust\_Acc

Payment

**Step 2: Find Relations**

**Provides**

Loan

Branch

**Works in**

Branch

Employee

**Has**

Cust\_Acc

Branch

**Borrows**

Loan

Cust\_Acc

**Deals with**

Cust\_Acc

Employee

Employee

**Supervises**

**Step 3: Identify the Relevant Attributes**

**Branch:** Branch\_name, Branch\_city

**Loan:** Loan\_number, amount

**Employee:** Emp\_id, Emp\_name, contact, cust\_id

**Cust\_Acc:** Cust\_id, Cust\_name, address, Acc\_no, Balance, loan\_number

**Payment:** Pay\_no, pay\_date, amount, loan\_number

**Creating Tables**

SQL> create table branch (branch\_name varchar2(20) primary key, branch\_city varchar2(10));

Table created.

SQL> create table loan (loan\_number number(10) primary key, amount number(10));

Table created.

SQL> create table cust\_acc (cust\_id number(4), cust\_name varchar2(15), address varchar2(20), acc\_no number(10), balance number(6),loan\_number number(10), primary key(cust\_id, acc\_no, loan\_number), foreign key(loan\_number) references loan(loan\_number));

Table created.

SQL> create table employee (emp\_id number(5), emp\_name varchar2(20), contact number(10), cust\_id number(4), primary key(emp\_id, cust\_id), foreign key(cust\_id) references cust\_acc(cust\_id));

Table created.

SQL> create table payment (pay\_no number(10), pay\_date date, amount number(6), loan\_number number(10), primary key(pay\_no, loan\_number), foreign key(loan\_number) references loan(loan\_number));

Table created.

**Inserting Values into the Tables**

SQL> insert into branch values('&branch\_name','&branch\_city');

Enter value for branch\_name: union bank gitam

Enter value for branch\_city: vizag

old 1: insert into branch values('&branch\_name','&branch\_city')

new 1: insert into branch values('union bank gitam','vizag')

1 row created.

SQL> /

Enter value for branch\_name: union bank gpt

Enter value for branch\_city: vizag

old 1: insert into branch values('&branch\_name','&branch\_city')

new 1: insert into branch values('union bank gpt','vizag')

1 row created.

SQL> select \* from branch;



SQL> insert into loan values(&loan\_number, &amount);

Enter value for loan\_number: 1210310401

Enter value for amount: 22650

old 1: insert into loan values(&loan\_number,&amount)

new 1: insert into loan values(1210310401,22650)

1 row created.

SQL> /

Enter value for loan\_number: 1210310437

Enter value for amount: 26850

old 1: insert into loan values(&loan\_number,&amount)

new 1: insert into loan values(1210310437,26850)

1 row created.

SQL> select \* from loan;



SQL> insert into cust\_acc values (&cust\_id,'&cust\_name','&address',&acc\_no,&balance,&loan\_number);

Enter value for cust\_id: 1201

Enter value for cust\_name: Harish

Enter value for address: muralinagar

Enter value for acc\_no: 14105142

Enter value for balance: 56893

Enter value for loan\_number: 1210310437

old 1: insert into cust\_acc values(&cust\_id,'&cust\_name','&address',&acc\_no,&balance,&loan\_number)

new 1: insert into cust\_acc values(1201,'harish','muralinagar',14105142,56893,1210310437)

1 row created.

SQL> /

Enter value for cust\_id: 1305

Enter value for cust\_name: satish

Enter value for address: rushikonda

Enter value for acc\_no: 13204254

Enter value for balance: 45632

Enter value for loan\_number: 1210310401

old 1: insert into cust\_acc values(&cust\_id,'&cust\_name','&address',&acc\_no,&balance,&loan\_number)

new 1: insert into cust\_acc values(1305,'satish','rushikonda',13204254,45632,1210310401)

1 row created.

SQL> select \* from cust\_acc;



SQL> insert into employee values (&emp\_id,'&emp\_name', &contact,&cust\_id);

Enter value for emp\_id: 101

Enter value for emp\_name: Suresh

Enter value for contact: 9868582832

Enter value for cust\_id: 1305

old 1: insert into employee values (&emp\_id,'&emp\_name',&contact,&cust\_id)

new 1: insert into employee values (101,'suresh',9868582832,1305)

1 row created.

SQL> /

Enter value for emp\_id: 103

Enter value for emp\_name: Ramesh

Enter value for contact: 9565986523

Enter value for cust\_id: 1201

old 1: insert into employee values (&emp\_id,'&emp\_name',&contact,&cust\_id)

new 1: insert into employee values (103,'ramesh',9565986523,1201)

1 row created.

SQL> select \* from employee;



SQL> insert into payment values(&pay\_no,'&date',&amount,&loan\_number);

Enter value for pay\_no: 5555566565

Enter value for date: 12-oct-98

Enter value for amount: 2000

Enter value for loan\_number: 1210310401

old 1: insert into payment values(&pay\_no,'&date',&amount,&loan\_number)

new 1: insert into payment values(5555566565,'12-oct-98',2000,1210310401)

1 row created.

SQL> /

Enter value for pay\_no: 6666654545

Enter value for date: 15-aug-99

Enter value for amount: 15000

Enter value for loan\_number: 1210310437

old 1: insert into payment values(&pay\_no,'&date',&amount,&loan\_number)

new 1: insert into payment values(6666654545,'15-aug-99',15000,1210310437)

1 row created.

SQL> select \* from payment;



**Queries**

1. Find the employee name who is dealing with customer called satish

SQL>select emp\_name from employee e, cust\_acc c where e.cust\_id=c.cust\_id and cust\_name='satish';

EMP\_NAME

--------------------

Suresh

1. Find the employee name dealing with loan amount of Rs.26850

SQL>select emp\_name

from employee e, cust\_acc c, loan l

where e.cust\_id=c.cust\_id and c.loan\_number=l.loan\_number and l.amount=26850;

EMP\_NAME

--------------------

Ramesh

1. Find the payment amount details paid yet by harish

SQL>select p.amount

from payment p, cust\_acc c

where c.loan\_number=p.loan\_number and cust\_name='harish';

AMOUNT

----------

15000

1. Find date of payment and loan amount suresh is dealing with

SQL>select p.pay\_date, l.amount

from payment p, cust\_acc c, loan l, employee e

where p.loan\_number=c.loan\_number and p.loan\_number=l.loan\_number and

c.cust\_id=e.cust\_id and e.emp\_name='suresh';

PAY\_DATE AMOUNT

--------- ----------

12-OCT-98 22650

1. Harish has got a problem in his transaction. Now he wants to speak about the problem and approaches the bank. What is the information that the bank has provided him?

SQL>select emp\_name, contact

from employee e, cust\_acc c

where e.cust\_id=c.cust\_id and c.cust\_name='harish';

EMP\_NAME CONTACT

-------------------- ----------

ramesh 9565986523

1. The manager of the gitam branch bank wants to know the details of the loans provides by the bank to the concerned customers so that he would post some gifts for the members who paid their loans in time.

SQL> select cust\_name, l.amount

from loan l, cust\_acc c

where l.loan\_number=c.loan\_number;

CUST\_NAME AMOUNT

--------------- ----------

harish 26850

satish 22650

1. Find the last payment date and balance of the loan no. 1210310437

SQL> select pay\_date, c.balance

from cust\_acc c, payment p

where p.loan\_number=c.loan\_number and c.loan\_number=1210310437;

PAY\_DATE BALANCE

--------- ----------

15-AUG-99 56893

1. Find the payment amount to the loan number 1210310401

SQL> select amount from payment

where loan\_number=1210310401;

AMOUNT

----------

2000

1. Satish has deposited an amount of Rs 2000 which was not updated in the data and he posted a complaint. How would his concerned employee rectify it?

SQL>select balance from cust\_acc

where cust\_name='satish';

BALANCE

----------

45632

SQL>update cust\_acc

set balance=47632

where cust\_name='satish';

1 row updated.

SQL> select balance from cust\_acc

where cust\_name='satish';

BALANCE

----------

47632

1. Find balance of the customer with id=1305

SQL> select balance from cust\_acc

where cust\_id=1305;

BALANCE

----------

47632

**Normalization**

In the above scenario **Branch, Loan, Employee, Payment** tables can no further be decomposed as they are already in normalized forms. **Cust\_acc** is in un-normalized form, this table is to be further decomposed to reduce the redundancy problem.

**Decomposing cust\_acc table**

Cust\_acc is in 1st normal form as the table can be into small parts and still remain meaningful.

Cust\_details account



**2nd Normal form**

The table satisfies the 1st normal form so now to satisfy 2nd normal form we check for partial dependency and if there is any we eliminate it.

Here cust\_id and loan\_number can determine the cust\_details. Cust\_id and acc\_no can determine the account details. So we eliminate the partial dependency by decomposing the table into 2 different tables.

Cust\_details



Account\_details



The tables formed now are in normalized forms and can no further be decomposed into smaller tables.