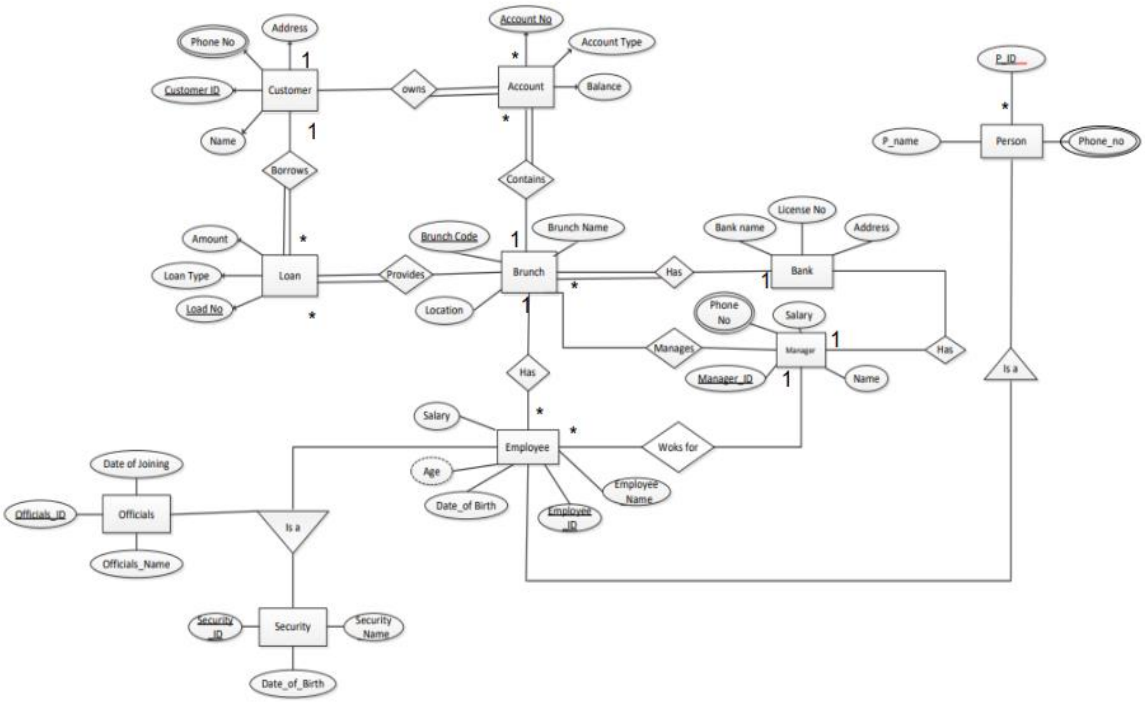
PROJECT REPORT
INTRODUCTION TO DATABASE

BANK MANAGEMENT SYSTEM

In this project we will discuss about a management system of a bank named CCPD. We have used total 10 entities such as **BANK, BRANCH, MANAGER, EMPLOYEE, LOAN, ACCOUNT, CUSTOMER, PERSON, SECURITY, OFFICIALS**. The attributes of **BANK** are: Bank name , licence_no (Primary key) and address. The bank has many managers and branches. One manager manages one branch. **MANAGER** and **BRANCH** also has some attributes. **BRANCH** has its name, id and location **. MANAGER** has m_id, m_name, salary, phone no and salary. One branch has many employee. Employees work for one manager who manages the branch. **EMPLOYEE** has name, id, date of birth , age and salary. One branch has many Account. **ACCOUNT** carries account no, type and balance. **CUSTOMER** has name, address, phone no, customer id. One customer can have one loan or many customers can have many loans. **LOAN** has loan type, amount, loan no. The brunch provides the loan.

ER diagram:


Phone No
Address
Customer ID
Customer
Name
owns
Account No
Account Type
Account
Balance
P_ID
P_name
Person
Phone_no
Borrows
Contains
Amount
Loan Type
Load No
Loan
Brunch Code
Brunch Name
Location
Brunch
Has
License No
Bank name
Address
Bank
Provides
Phone No
Salary
Manages
Manager
Has
Manager_ID
Name
Is a
Has
Salary
Employee
Woks for
Age
Date_of Birth
Employee Name
Employee _ID
Date of Joining
Officials_ID
Officials
Officials_Name
Is a
Security _ID
Security
Security Name
Date_of_Birth

* Normalization *

Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies. Normalization rules divides larger tables into smaller tables and links them using relationships. The purpose of Normalization in SQL is to eliminate redundant (repetitive) data and ensure data is stored logically.

1NF (First Normal Form):

- Each table cell should contain a single value.
- Each record needs to be unique.

2NF (Second Normal Form):

- Rule 1- Be in 1NF
- Rule 2- Single Column Primary Key that does not functionally dependent on any subset of candidate key relation

3NF (Third Normal Form):

- Rule 1- Be in 2NF
- Rule 2- Has no **transitive** functional dependencies

A Transitive dependency in a database is an indirect relationship between values in the same table that causes a functional dependency

1#Customer * Account

Owns(Customer_ID, C_Name, Phone_NO, C_Address, Account_No,Account Type, Balance)

Customer_ID	C_Name	Phone_NO	C_Address	Account_NO	Acc type	Balance
1001	Nasim	12324 12325	Dhaka	101	Current	1000
2002	Nahin	14361 14362	Dhaka	102	Saving	2000
3003	Rafia	13451 13452	Cumilla	103	Saving	3000
4004	Tangir	1024 1025	Barishal	104	Current	4000

1NF:

Customer_ID	C_Name	Phone_NO	C_Address	Account_NO	Acc type	Balance
1001	Nasim	12324	Dhaka	101	Current	1000
1001	Nasim	12325	Dhaka	101	Current	1000
2002	Nahin	14361	Dhaka	102	Saving	2000
2002	Nahin	14362	Dhaka	102	Saving	2000
3003	Rafia	13451	Cumilla	103	Saving	3000
3003	Rafia	13452	Cumilla	103	Saving	3000
4004	Tangir	1024	Barishal	104	Current	4000
4004	Tangir	1025	Barishal	104	Current	4000

2NF:

Customer_ID	C_Name	Phone_NO	C_Address
1001	Nasim	12324	Dhaka
1001	Nasim	12325	Dhaka
2002	Nahin	14361	Dhaka
2002	Nahin	14362	Dhaka
3003	Rafia	13451	Cumilla
3003	Rafia	13452	Cumilla
4004	Tangir	1024	Barishal
4004	Tangir	1025	Barishal

Account_NO	Acc type	Balance
101	Current	1000
102	Saving	2000
103	Saving	3000
104	Current	4000

Customer_ID	Account_NO
1001	101
2002	102
3003	103
4004	104

3NF : No transitive dependency

2#Customer * Loan

Borrows(Customer_ID, Name, Phone_NO, C_Address, Loan_NO, Loan Type, Amount)

Customer_ID	C_Name	Phone_NO	C_Address	Loan_NO	Loan type	Amount
1001	Nasim	12324	Dhaka	111	Credit	100
2002	Nahin	14361	Dhaka	112	Home	200
3003	Rafia	13451	Cumilla	113	Business	300
4004	Tangir	1024	Barishal	114	Personal	400

1NF:

Customer_ID	Name	Phone_NO	C_Address	Loan_NO	Loan type	Amount
1001	Nasim	12324	Dhaka	111	Credit	100
1001	Nasim	12325	Dhaka	111	Credit	100
2002	Nahin	14361	Dhaka	112	Home	200
2002	Nahin	14362	Dhaka	112	Home	200
3003	Rafia	13451	Cumilla	113	Business	300
3003	Rafia	13452	Cumilla	113	Business	300
4004	Tangir	1024	Barishal	114	Personal	400
4004	Tangir	1025	Barishal	114	Personal	400

2NF:

Customer_ID	C_Name	Phone_NO	C_Address
1001	Nasim	12324	Dhaka
1001	Nasim	12325	Dhaka
2002	Nahin	14361	Dhaka
2002	Nahin	14362	Dhaka
3003	Rafia	13451	Cumilla
3003	Rafia	13452	Cumilla
4004	Tangir	1024	Barishal
4004	Tangir	1025	Barishal

Loan_NO	Loan type	Amount
111	Credit	1000
112	Home	2000
113	Business	3000
114	Personal	4000

Customer_ID	Loan_NO
1001	111
2002	112
3003	113
4004	114

3NF : No transitive dependency

3#Loan * Branch

Provides(Loan_NO, Loan type, Amount, Branch_ID, Branch_Name, Location)

Loan_NO	Loan type	Amount	Branch_ID	Branch_Name	Location
111	Credit	100	1111	Dhaka Branch	Dhaka
112	Home	200	1111	Dhaka Branch	Dhaka
113	Business	300	2222	Tangail Branch	Tangail
114	Personal	400	2222	Tangail Branch	Tangail

1NF : No multivalued attributes.

2NF:

Loan_NO	Loan type	Amount
111	Credit	100
112	Home	200
113	Business	300
114	Personal	400

Branch_ID	Branch_Name	Location
1111	Dhaka Branch	Dhaka
2222	Tangail Branch	Tangail

Loan_NO	Branch_ID
111	1111
112	1111
113	2222
114	2222

3NF : No transitive dependency

4#Branch * Account

Contains(Branch_ID, Branch_Name, Location , Account_No, Account Type, Balance)

Branch_ID	Branch_Name	Location	Account_NO	Acc type	Balance
1111	Dhaka Branch	Dhaka	101	Current	1000
1111	Dhaka Branch	Dhaka	102	Saving	2000
2222	Tangail Branch	Tangail	103	Saving	3000
2222	Tangail Branch	Tangail	104	Current	4000

1NF : No multivalued attributes.

2NF:

Account_NO	Acc type	Balance
101	Current	1000
102	Saving	2000
103	Saving	3000

104	Current	4000

Branch_ID	Branch_Name	Location
1111	Dhaka Branch	Dhaka
2222	Tangail Branch	Tangail

Account_NO	Branch_ID
101	1111
102	1111
103	2222
104	2222

3NF : No transitive dependency

5#Branch * Employees

Has (Branch_ID, Branch_Name, Location, E_ID, E_Name, Salary, DOB, Age)

Branch_ID	Branch_Name	Location	E_ID	E_Name	Salary	DOB	Age
1111	Dhaka Branch	Dhaka	2001	Anng	100	1-sep-1980	40
1111	Dhaka Branch	Dhaka	2002	Katara	200	25-oct-1981	39
2222	Tangail Branch	Tangail	2003	Suki	300	12-may-1979	41
2222	Tangail Branch	Tangail	2004	Zuko	400	12-jan-1982	38

1NF : No multivalued attributes.

2NF:

E_ID	Branch_ID	E_Name	Salary	DOB	Age
2001	1111	Anng	100	1-sep- 1980	40
2002	1111	Katara	200	25-oct-1981	39
2003	2222	Suki	300	12-may-1979	41

2004	2222	Zuko	400	12-jan-1982	38

Branch_ID	Branch_Name	Location
1111	Dhaka Branch	Dhaka
2222	Tangail Branch	Tangail

3NF:

E_ID	Branch_ID	E_Name	Salary	Birth_ID
2001	1111	Anng	100	1
2002	1111	Katara	200	2
2003	2222	Suki	300	3
2004	2222	Zuko	400	4

Birth_ID	DOB	Age
1	1-sep- 1980	40
2	25-oct-1981	39
3	12-may-1979	41
4	12-jan-1982	38

Branch_ID	Branch_Name	Location
1111	Dhaka Branch	Dhaka
2222	Tangail Branch	Tangail

6#Branch * Bank

Has (Branch_ID, Branch_Name, Location, Licence_NO, Bank_Name, Address)

Branch_ID	Branch_Name	Location	Licence_NO	Bank_Name	Address
1111	Dhaka Branch	Dhaka	9090	CCPD	Ladakh
2222	Tangail Branch	Tangail	9090	CCPD	Ladakh
3333	Gazipur Branch	Gazipur	9090	CCPD	Ladakh
4444	Kumilla Branch	Kumilla	9090	CCPD	Ladakh

1NF : No multivalued attributes.

2NF:

Branch_ID	Branch_Name	Location	Licence_NO
1111	Dhaka Branch	Dhaka	9090
2222	Tangail Branch	Tangail	9090
3333	Gazipur Branch	Gazipur	9090
4444	Kumilla Branch	Kumilla	9090

Licence_NO	Bank_Name	Address
9090	CCPD	Ladakh

3NF: No transitive dependency

7#Branch * Manager

Manages (Branch_ID, Branch_Name, Location, M_ID, Name, Salary, Phone_NO)

Branch_ID	Branch_Name	Location	M_ID	Name	Salary	Phone_NO
1111	Dhaka Branch	Dhaka	601	Ash	9000	01789
						01798
2222	Tangail Branch	Tangail	602	Sam	8000	01756
						01765
3333	Gazipur Branch	Gazipur	603	Ram	7000	01723
						01732
4444	Kumilla Branch	Kumilla	604	Tom	8000	01712
						01721

1NF:

Branch_ID	Branch_Name	Location	M_ID	Name	Salary	Phone_NO
1111	Dhaka Branch	Dhaka	601	Ash	9000	01789
1111	Dhaka Branch	Dhaka	601	Ash	9000	01798
2222	Tangail Branch	Tangail	602	Sam	8000	01756
2222	Tangail Branch	Tangail	602	Sam	8000	01765
3333	Gazipur Branch	Gazipur	603	Ram	7000	01732
3333	Gazipur Branch	Gazipur	603	Ram	7000	01723
4444	Kumilla Branch	Kumilla	604	Tom	8000	01712
4444	Kumilla Branch	Kumilla	604	Tom	8000	01721

2NF:

Branch_ID	Branch_Name	Location	M_ID
1111	Dhaka Branch	Dhaka	601
2222	Tangail Branch	Tangail	602
3333	Gazipur Branch	Gazipur	603
4444	Kumilla Branch	Kumilla	604

M_ID	Name	Salary	Phone_NO
601	Ash	9000	01789
601	Ash	9000	01798
602	Sam	8000	01756
602	Sam	8000	01765
603	Ram	7000	01732
603	Ram	7000	01723
604	Tom	8000	01712
604	Tom	8000	01721

3NF: No transitive dependency

8#Manager * Employee

Works For (M_ID, Name, Salary, Phone_NO, E_ID, E_Name, Salary, DOB, Age)

M_ID	Name	M_Salary	Phone_NO	E_ID	E_Name	E_Salary	DOB	Age
601	Ash	9000	01789 01798	2001	Anng	100	1-sep-1980	40
601	Ash	9000	01789 01798	2002	Katara	200	25-oct-1981	39
602	Sam	8000	01756 01765	2003	Suki	300	12-may-1979	41
602	Sam	8000	01756 01765	2004	Zuko	400	12-jan-1982	38

1NF:

M_ID	Name	M_Salary	Phone_NO	E_ID	E_Name	E_Salary	DOB	Age
601	Ash	9000	01789	2001	Anng	100	1-sep-1980	40
601	Ash	9000	01789	2001	Anng	100	1-sep-1980	40
601	Ash	9000	01798	2002	Katara	200	25-oct-1981	39

601	Ash	9000	01798	2002	Katara	200	25-oct-1981	39
602	Sam	8000	01756	2003	Suki	300	12-may-1979	41
602	Sam	8000	01756	2003	Suki	300	12-may-1979	41
602	Sam	8000	01765	2004	Zuko	400	12-jan-1982	38
602	Sam	8000	01765	2004	Zuko	400	12-jan-1982	38

2NF:

E_ID	E_Name	E_Salary	DOB	Age	M_ID
2001	Anng	100	1-sep- 1980	40	601
2001	Anng	100	1-sep- 1980	40	601
2002	Katara	200	25-oct-1981	39	601
2002	Katara	200	25-oct-1981	39	601
2003	Suki	300	12-may-1979	41	602
2003	Suki	300	12-may-1979	41	602
2004	Zuko	400	12-jan-1982	38	602
2004	Zuko	400	12-jan-1982	38	602

M_ID	Name	M_Salary	Phone_NO
601	Ash	9000	01789
601	Ash	9000	01798
602	Sam	8000	01756
602	Sam	8000	01765

3NF :

M_ID	Name	M_Salary	Phone_NO
601	Ash	9000	01789
601	Ash	9000	01798
602	Sam	8000	01756
602	Sam	8000	01765

E_ID	E_Name	E_Salary	Birth_ID	M_ID
2001	Anng	1000	1	601
2002	Katara	2000	2	601
2003	Suki	3000	3	602
2004	Zuko	4000	4	602

Birth_ID	DOB	Age
1	1-sep-1980	40
2	25-oct-1981	39
3	12-may-1979	41
4	12-jan-1982	38

9#Bank * Manager

Has (Licence_NO, Bank_Name, Address, M_ID, Name, Salary, Phone_NO)

Licence_NO	Bank_Name	Address	M_ID	Name	Salary	Phone_NO
9090	CCPD	Ladakh	601	Ash	9000	01789
						01798
9090	CCPD	Ladakh	602	Sam	8000	01756
						01765
9090	CCPD	Ladakh	603	Ram	7000	01756
						01765
9090	CCPD	Ladakh	604	Tom	8000	01712
						01721

1NF:

Licence_NO	Bank_Name	Address	M_ID	Name	Salary	Phone_NO
9090	CCPD	Ladakh	601	Ash	9000	01789
9090	CCPD	Ladakh	601	Ash	9000	01798
9090	CCPD	Ladakh	602	Sam	8000	01756
9090	CCPD	Ladakh	602	Sam	8000	01765
9090	CCPD	Ladakh	603	Ram	7000	01756
9090	CCPD	Ladakh	603	Ram	7000	01765
9090	CCPD	Ladakh	604	Tom	8000	01712
9090	CCPD	Ladakh	604	Tom	8000	01721

2NF:

Licence_NO	M_ID	Name	Salary	Phone_NO
9090	601	Ash	9000	01789
9090	601	Ash	9000	01798
9090	602	Sam	8000	01756
9090	602	Sam	8000	01765

9090	603	Ram	7000	01756
9090	603	Ram	7000	01765
9090	604	Tom	8000	01712
9090	604	Tom	8000	01721

Licence_NO	Bank_Name	Address
9090	CCPD	Ladakh

3NF: No transitive dependency

* Final Tables *

Customer:

Customer_ID	C_Name	Phone_NO	C_Address

Account:

Account_NO	Acc type	Balance

Customer_Account:

Customer_ID	Account_NO

~~Customer_ID~~	~~Name~~	~~Phone_NO~~	~~Address~~

Loan:

Loan_NO	Loan type	Amount

Customer_Loan

Customer_ID	Loan_NO

~~Loan_NO~~	~~Loan type~~	~~Amount~~

Loan_Branch

Loan_NO	Branch_ID

~~Account_NO~~	~~Acc type~~	~~Balance~~

Branch:

Branch_ID	Branch_Name	Location

Account_Branch:

Account_NO	Branch_ID

Branch_Employee

E_ID	Branch_ID	E_Name	Salary	Birth_ID

Birth

Birth_ID	DOB	Age

~~Branch_ID~~	~~Branch_Name~~	~~Location~~

Bank:

Licence_NO	Bank_Name	Address

Branch_Bank:

Branch_ID	Branch_Name	Location	Licence_NO

Manager:

M_ID	M_Name	M_Salary	Phone_NO

Branch_Manager:

Branch_ID	Branch_Name	Location	M_ID

~~M_ID~~	~~Name~~	~~M_Salary~~	~~Phone_NO~~

Employee_Manager:

E_ID	E_Name	E_Salary	Birth_ID	M_ID

~~Birth_ID~~	~~DOB~~	~~Age~~

Bank_Manager:

Licence_NO	M_ID	Name	M_Salary	Phone_NO

~~Licence_NO~~	~~Bank_Name~~	~~Address~~

~~Branch_ID~~	~~Branch_Name~~	~~Location~~

From table Branch_Bank, Branch_Manager, Bank_Manager:

We can notice there is a transitive dependency between these three table so we have to normalize them again.

12.B_M_Bank:

Branch_ID	M_ID	Licence_NO

And from table Branch_Employee and Employee_Manager we get,

13.B_M_Emp:

Branch_ID	M_ID	E_ID

14.Employee:

E_ID	E_Name	E_Salary	Birth_ID

* QUERIES *

User Creation:

1. CREATE USER PROJECT IDENTIFIED BY PROJECT

Granting privileges:

1. GRANT CREATE TABLE, CREATE SEQUENCE, CREATE VIEW TO PROJECT
2. GRANT CONNECT, RESOURCE, UNLIMITED TABLESPACE TO PROJECT

Table creation:

1.Customer

CREATE TABLE CUSTOMER (CUSTOMER_ID NUMBER(20) CONSTRAINT PK_CUSTOMER_ID PRIMARY KEY, C_NAME VARCHAR2(50) ,PHONE_NO VARCHAR2(12), C_ADDRESS VARCHAR2(15))

2.Account:

CREATE TABLE ACCOUNT (ACCOUNT_NO NUMBER(20) CONSTRAINT PK_ACCOUNT_NO PRIMARY KEY, ACC_TYPE VARCHAR2(50), BALANCE NUMBER(12) CONSTRAINT CK_BAL CHECK(BALANCE >0))

3.Customer_Account:

CREATE TABLE CUSTOMER_ACCOUNT (CUSTOMER_ID NUMBER(20), ACCOUNT_NO NUMBER(20))

4.Loan:

CREATE TABLE LOAN (LOAN_NO NUMBER(10) CONSTRAINT PK_LOAN PRIMARY KEY, LOAN_TYPE VARCHAR2(14) ,BALANCE NUMBER)

5.Customer_Loan:

CREATE TABLE CUSTOMER_LOAN (CUSTOMER_ID NUMBER(20), LOAN_NO NUMBER(20))

6.Loan_Branch:

CREATE TABLE LOAN_BRANCH (LOAN_NO NUMBER(20) NOT NULL, BRANCH_ID NUMBER(20) CONSTRAINT FK_BRANCH REFERENCES BRANCH)

7.Branch:

CREATE TABLE BRANCH (BRANCH_ID NUMBER(4) CONSTRAINT PK_BRANCH PRIMARY KEY,BRANCH_NAME VARCHAR2(14) ,LOCATION VARCHAR2(13))

8.Account_Branch:

CREATE TABLE ACCOUNT_BRANCH (ACCOUNT_NO NUMBER, BRANCH_ID NUMBER)

9.Birth:

CREATE TABLE BIRTH (BIRTH_ID NUMBER(10) CONSTRAINT PK_BID PRIMARY KEY,DOB DATE,AGE VARCHAR2(2))

10.Bank:

CREATE TABLE BANK (LICENCE_NO NUMBER(4) CONSTRAINT PK_BANK PRIMARY KEY,BANK_NAME VARCHAR2(14) ,ADDRESS VARCHAR2(13))

11.Manager:

CREATE TABLE MANAGER (M_ID NUMBER(20) CONSTRAINT PK_MID PRIMARY KEY, M_NAME VARCHAR2(50) ,PHONE_NO VARCHAR2(12),SALARY NUMBER CONSTRAINT CK_SAL CHECK(SALARY >0))

12.B_M_Bank:

CREATE TABLE B_M_BANK (BRANCH_ID NUMBER, M_ID NUMBER, LICENCE_NO NUMBER)

13. B_M_EMP:

CREATE TABLE B_M_EMP (BRANCH_ID NUMBER, M_ID NUMBER, E_ID NUMBER)

14.Employee:

CREATE TABLE EMPLOYEE (E_ID NUMBER(4),E_NAME VARCHAR2(14) ,E_SALARY NUMBER , BIRTH_ID NUMBER)

15. Security:

CREATE TABLE SECURITY (SECURITY_ID NUMBER(20), S_NAME VARCHAR2(50) , DOB VARCHAR2(15))

16. Person:

CREATE TABLE PERSON (P_ID NUMBER(20) CONSTRAINT PK_PID PRIMARY KEY, P_NAME VARCHAR2(50) ,PHONE_NO VARCHAR2(12))

17. Officials:

CREATE TABLE OFFICIALS (O_ID NUMBER(20) CONSTRAINT PK_OID PRIMARY KEY, O_NAME VARCHAR2(50) , DOJ VARCHAR2(15))

* INSERTING VALUES *

1.CUSTOMER:

CUSTOMER_ID	C_NAME	PHONE_NO	C_ADDRESS
2002	Nahin	14361	Dhaka
3003	Rafia	13451	cumilla
1001	Nasim	12324	Dhaka
4004	TANGIR	1024	Barishal

2.ACCOUNT:

ACCOUNT_NO	ACC_TYPE	BALANCE
101	CURRENT	1000
102	SAVING	2000
103	SAVING	3000
104	SAVING	4000

3.CUSTOMER_ACCOUNT :

CUSTOMER_ID	ACCOUNT_NO
1001	101
2002	102
3003	103
4004	104

4.LOAN :

LOAN_NO	LOAN_TYPE	BALANCE
111	CREDIT	1000
112	HOME	2000
113	BUSINESS	3000
114	BUSINESS	4000

5. CUSTOMER_LOAN :

CUSTOMER_ID	LOAN_NO
1001	111
2002	112
3003	113
4004	114

6.ACCOUNT_BRANCH:

ACCOUNT_NO	BRANCH_ID
101	1111
102	1111
103	2222
104	2222

7.BIRTH

**INSERTED USING SEQUENCE

INSERT INTO BIRTH VALUES (BIRTH_BIRTH_ID.NEXTVAL,TO_DATE('12-01-1982','DD-MM-YYYY'),'38')

BIRTH_ID	DOB	AGE
6	12-JAN-82	38
7	12-MAY-79	41
8	25-OCT-81	39
9	01-SEP-80	40

8.BANK:

LICENCE_NO	BANK_NAME	ADDRESS
9090	CCPD	LADAKH

9.MANAGER:

M_ID	M_NAME	PHONE_NO	SALARY
601	ASH	01789	9000
602	SAM	01756	8000
603	RAM	01732	7000
604	TOM	01721	8000

10. B_M_BANK:

BRANCH_ID	M_ID	LICENCE_NO
1111	601	9090
2222	602	9090
3333	603	9090
4444	604	9090

11.B_M_Emp:

BRANCH_ID	M_ID	E_ID
1111	601	2001
1111	601	2002
2222	602	2003
2222	602	2004
3333	603	-
4444	604	-

12.EMPLOYEE

**INSERTED USING SEQUENCE

**INSERT INTO EMPLOYEE VALUES (EMPLOYEE_E_ID.NEXTVAL,'ANNG',1000,1)

E_ID	E_NAME	E_SALARY	BIRTH_ID
2001	ANNG	1000	1
2002	KATARA	2000	2
2003	SUKI	3000	3
2004	ZUKO	4000	4

13. SECURITY:

SECURITY_ID	S_NAME	DOB
1	ZACK	17-DEC-90
3	KARL	19-JUL-86
2	RON	01-FEB-89

14. PERSON:

P_ID	P_NAME	PHONE_NO
101	HANS	987
202	RICK	986

15. OFFICIALS:

O_ID	O_NAME	DOJ
7771	MORTY	1990-12-12
7772	NONTE	2000-1-18
7773	FONTE	2005-9-1
7774	BATUL	2003-5-27

* SEQUENCE *

FOR BIRTH ID:

CREATE SEQUENCE BIRTH_BIRTH_ID INCREMENT BY 1 START WITH 1 NOCACHE NOCYCLE

FOR EMPLOYEE ID:

CREATE SEQUENCE EMPLOYEE_E_ID INCREMENT BY 1 START WITH 2001 NOCACHE NOCYCLE

SEQUENCE_NAME	MIN_VALUE	INCREMENT_BY	LAST_NUMBER
BIRTH_BIRTH_ID	1	1	7
EMPLOYEE_E_ID	1	1	2005

* VIEW *

1.SIMPLE VIEW:

CREATE VIEW CINFO (ID, NAME, ADDRESS) AS SELECT CUSTOMER_ID, C_NAME, C_ADDRESS FROM CUSTOMER WHERE C_ADDRESS='DHAKA''

2.COMPLEX VIEW:

CREATE VIEW EMPINFO AS SELECT E.E_ID, E.E_NAME, B.M_ID, M.M_NAME FROM EMPLOYEE E, B_M_EMP B, MANAGER M WHERE E.E_ID=B.E_ID AND B.M_ID=M.M_ID

Simple Queries:

1.Using view:

a. Print customer name whose id is 1001 and lives in DHAKA.

Answer: SELECT NAME FROM CINFO WHERE ID IN (1001) AND ADDRESS IN ('DHAKA')

b. Show the employees working for ASH.

Answer: SELECT * FROM EMPINFO WHERE M_NAME='ASH'

2.Using tables:

a. Print customer name whose id is 1001 and lives in DHAKA.

Answers: SELECT C_NAME FROM CUSTOMER WHERE CUSTOMER_ID=1001 AND C_ADDRESS='DHAKA'

b. Show the employees working for ASH.

Answers: SELECT E.E_ID, E.E_NAME, E.E_SALARY,B.DOB, B.AGE, M.M_ID, M.M_NAME FROM MANAGER M , EMPLOYEE E, BIRTH B, B_M_EMP BM WHERE E.BIRTH_ID=B.BIRTH_ID AND E.E_ID=BM.E_ID AND BM.M_ID=M.M_ID AND M.M_NAME='ASH'

Queries:

1. View all the customer.

```
SELECT*
FROM CUSTOMER;
```

2.View customer name, id those who are from Dhaka

```
SELECT C_NAME, CUSTOMER_ID
FROM CUSTOMER
WHERE CUSTOMER_ADDRESS = DHAKA;
```

3. View account type and balance of account no 101

```
SELECT ACC_TYPE, BAL
FROM ACCOUNT
WHERE ACCOUNT_NO = 101;
```

4. View the amount of loan taken by loan id no 111

```
SELECT LOAN_AMOUNT
FROM LOAN
WHERE LOAN_NO = 111;
```

5. View dob of security name KARL

```
SELECT DOB
FROM SECURITY
WHERE S_NAME = 'KARL';
```

6. Show the phone no of manager name Tom

```
SELECT PHONE_NO
```

```
FROM MANAGER
WHERE M_NAME = 'TOM'
```

7.Show all the employee earning more than 2000

```
SELECT *
FROM EMP
WHERE SAL>2000;
```

8. View employee id earning more or equal to 4000

```
SELECT E_ID
FROM EMP
WHERE SAL>=4000;
```

9. Show the joining date of official whose joining month is “January”

```
SELECT OFFICIALS
FROM OFFICIALS
WHERE TO_CHAR(JOINING_DATE,'MM')='01';
```

10. Show the address of bank name CCPD

```
SELECT LICENSE_ID
FROM BANK
WHERE BANK_NAME = CCPD;
```