Name of Experiment :-

Creating tables of “Banking Enterprise” Database.

Objectives:-

The objective is to creating tables of “Banking Enterprise” Database.

Theory:-

A database management system is a collection and a set of programs to access those data. The collection of , usually referred to as database.

Instances:

Database changes over time as information is inserted and deleted. The collection of information stored in the database at a particular moment in time as called an instance of the database, e.g., customer (customer_name, customer_street, customer_city) customer (‘Hayes’ , ‘ Main’ , ‘Harrison’)

Keys:

A key is a property of the entity set, rather than of the individual entities. It allows identifying a set of attributes that suffice to distinguish entities from each other. Keys are three types:

- ✓ Super key
- ✓ Candidate Key
- ✓ Primary key

Super key:

Super Key is a set of one or more attributes that, taken collectively, allows us to identify uniquely an entity in the entity set.

Primary key:

A primary key is one or more column (s) in a table used to uniquely identify each row in the table.

Primary key Constraint:

- ✓ It defines the column, as a mandatory column, i.e., the column can't be left blank,
- ✓ The NOT NULL attribute is active,
- ✓ The data held across the column MUST be UNIQUE.

Candidate key:

A candidate Key is a combination of one or more columns, the values of which uniquely identify each row of a table.

Foreign Key:

A foreign Key is a combination of columns with values based on the primary key from another table.

Foreign Key Constraint:

- ✓ Records can't be inserted into a detail table if corresponding records in the master table don't exist,
- ✓ Records of the master table can't be deleted if corresponding records in the detail table exist.

Schemas:

The overall design of the database is called the database Schema. A database schema is responsible for describing data in its various levels of abstraction.

For designing Banking Database the required Schemas becomes as follows:

Customer (**customer_name** , customer_street , customer_city)
primary key(customer_name)

Branch (**branch_name** , branch_city , assets)
primary key(branch_name)

Account (**account_no** , branch_name , balance)
primary key(account_no)

Depositor (**customer_name , account_no**)
primary key(customer_name, account_no));

Loan (**loan_no**, branch_name , amount)
primary key(loan_no)

Borrower (**customer_name** , **loan_no**)
primary key(customer_name,loan_no)

Procedure: Steps of Setting up a Database for a particular enterprise.

1. Define the high level requirements of the enterprise (this steps generates a document known as the system requirements specification).
2. Define a model containing all appropriate types of data and relationships.
3. Define the integrity constraints on the data.
4. Define the physical level.

5. For each known problem to be solved on a regular bases (e.g.tasks to be carried out by clerks or web users) define a user interface to carry out the task, and write the necessary application programs to implement the user interface.

SQL:

Now we create the tables in SQL for the above schemas

```
create table customer
(customer_name varchar(25) ,
customer_street varchar(25),
customer_city varchar(25),
primary key(customer_name));

sql> describe customer;
```

<u>output</u>

Field	Type	Null	Key	Default	Extra
customer_name	varchar(25)	NO	PRI		
customer_street	varchar(25)	YES		NULL	
customer_city	varchar(25)	YES		NULL	

```
create table branch
(branch_name varchar(25),
branch_city varchar(25),
assets numeric(15,4),
primary key(branch_name));

sql> describe branch;
```

<u>output</u>

Field	Type	Null	Key	Default	Extra
branch_name	varchar(25)	NO	PRI		
brnanch_city	varchar(25)	YES		NULL	
assets	decimal(15,4)	YES		NULL	

```
create table account
(account_no char(14),
branch_name varchar(25),
balance numeric(15,4),
primary key(account_no));

sql> describe account;
```

output

Field	Type	Null	Key	Default	Extra
account_no	char(14)	NO	PRI		
branch_name	varchar(25)	YES		NULL	
balance	decimal(15,4)	YES		NULL	

```
create table depositor
(customer_name varchar(25),
account_no char(14),
primary key(customer_name, account_no));
sql> describe depositor;
```

output

Field	Type	Null	Key	Default	Extra
customer_name	varchar(25)	NO	PRI		
account_no	char(14)	NO	PRI		

```
create table loan
(loan_no char(14),
branch_name varchar(25),
ammount numeric(14),
primary key(loan_no));
```

output

Field	Type	Null	Key	Default	Extra
loan_no	char(14)	NO	PRI		
branch_name	varchar(25)	YES		NULL	
ammount	decimal(14)	YES		NULL	

```
create table borrower
(customer_name varchar(25),
loan_no char(14),
primary key(customer_name,loan_no));
sql> describe borrower;
```

output

Field	Type	Null	Key	Default	Extra
customer_name	varchar(25)	NO	PRI		
loan_no	char(14)	NO	PRI		

Discussion:- By this database query we can creating tables of “Banking Enterprise ” Database.