Chit 12 –

create database bank;

use bank;

create table branch(

-> branch\_name varchar(100) primary key,

-> branch\_city varchar(100) not null,

-> assets varchar(100)

-> );

create table account(

-> acc\_no int primary key,

-> branch\_name varchar(100) not null,

-> balance int,

-> foreign key(branch\_name) references branch(branch\_name),

-> );

create table customer(

-> cust\_name varchar(100) primary key,

-> cust\_street varchar(100) not null,

-> cust\_city varchar(100) not null

-> );

create table depositor(

-> cust\_name varchar(100),

-> acc\_no int,

-> foreign key(cust\_name) references customer(cust\_name),

-> foreign key(acc\_no) references account(acc\_no)

-> );

create table loan(

-> loan\_no int primary key,

-> branch\_name varchar(100),

-> amount int not null,

-> foreign key(branch\_name) references branch(branch\_name)

-> );

create table borrower(

-> cust\_name varchar(100),

-> loan\_no int,

-> foreign key(cust\_name) references customer(cust\_name),

-> foreign key(loan\_no) references loan(loan\_no)

-> );

insert into branch

-> values ('Akurdi', 'Pune', 'Cash worth 200000'),

-> ('Nigdi', 'Pune', 'Cash worth 4000000'),

-> ('Andheri', 'Mumbai', 'Cash worth 10000000');

insert into customer

-> values ('Raj', 'Ravet', 'Pune'),

-> ('Shivam', 'Punawale', 'Pune'),

-> ('Aditya', 'Akurdi', 'Pune'),

-> ('Abhishek', 'Nigdi', 'Pune');

insert into account

-> values (111, 'Akurdi', 20000),

-> (112, 'Nigdi', 30000),

-> (113, 'Andheri', 50000),

-> (114, 'Andheri', 6000),

-> (115, 'Akurdi', 40000),

-> (116, 'Nigdi', 70000),

-> (117, 'Akurdi', 65000),

-> (118, 'Akurdi', 7400);

insert into depositor

-> values ('Abhishek', 111),

-> ('Aditya', 112),

-> ('Raj', 116);

insert into loan

-> values (1, 'Akurdi', 15000),

-> (2, 'Andheri', 200000),

-> (3, 'Akurdi', 20000),

-> (4, 'Akurdi', 4000),

-> (5, 'Nigdi', 50000),

-> (6, 'Akurdi', 1400),

-> (7, 'Nigdi', 1450);

insert into borrower

-> values ('Raj', 1),

-> ('Shivam', 3),

-> ('Aditya', 5);

QUERIES

• Find the names of all branches in loan relation.

select distinct branch\_name from loan;

• Find all loan numbers for loans made at Akurdi Branch with loan amount > 12000.

select loan\_no from loan

-> where branch\_name = 'Akurdi' and amount > 12000;

• Find all customers who have a loan from bank. Find their names,loan\_no and loan

amount.

select cust\_name, borrower.loan\_no, amount as loan\_amount

-> from borrower

-> join loan

-> on borrower.loan\_no = loan.loan\_no;

• List all customers in alphabetical order who have loan from Akurdi branch.

select customer.cust\_name, cust\_street, cust\_city from customer

-> join borrower

-> on customer.cust\_name = borrower.cust\_name

-> join loan

-> on borrower.loan\_no = loan.loan\_no

-> where branch\_name = 'Akurdi';

• Find all customers who have an account or loan or both at bank.

select customer.cust\_name, cust\_street, cust\_city

-> from customer

-> join depositor

-> on customer.cust\_name = depositor.cust\_name;

select customer.cust\_name, cust\_street, cust\_city

-> from customer

-> join borrower

-> on customer.cust\_name = borrower.cust\_name;

select customer.cust\_name, cust\_street, cust\_city

-> from customer

-> join depositor

-> on customer.cust\_name = depositor.cust\_name;

-> join borrower

-> on customer.cust\_name = borrower.cust\_name;

• Find average account balance at Akurdi branch.

select avg(balance) as avg\_balance

-> from account

-> where branch\_name = 'Akurdi';

• Find the average account balance at each branch

select branch\_name, avg(balance) as avg\_balance

-> from account

-> group by branch\_name;

• Find no. of depositors at each branch.

select branch\_name, count(\*) as no\_of\_depositors

-> from depositor

-> join account

-> on depositor.acc\_no = account.acc\_no

-> group by branch\_name;

• Find the branches where average account balance > 12000.

select branch\_name from account

-> where (select avg(balance) from account) > 12000

-> group by branch\_name;

• Find number of tuples in customer relation.

select count(\*) from customer;

• Delete all loans with loan amount between 1300 and 1500

delete from loan

-> where amount between 1300 and 1500;