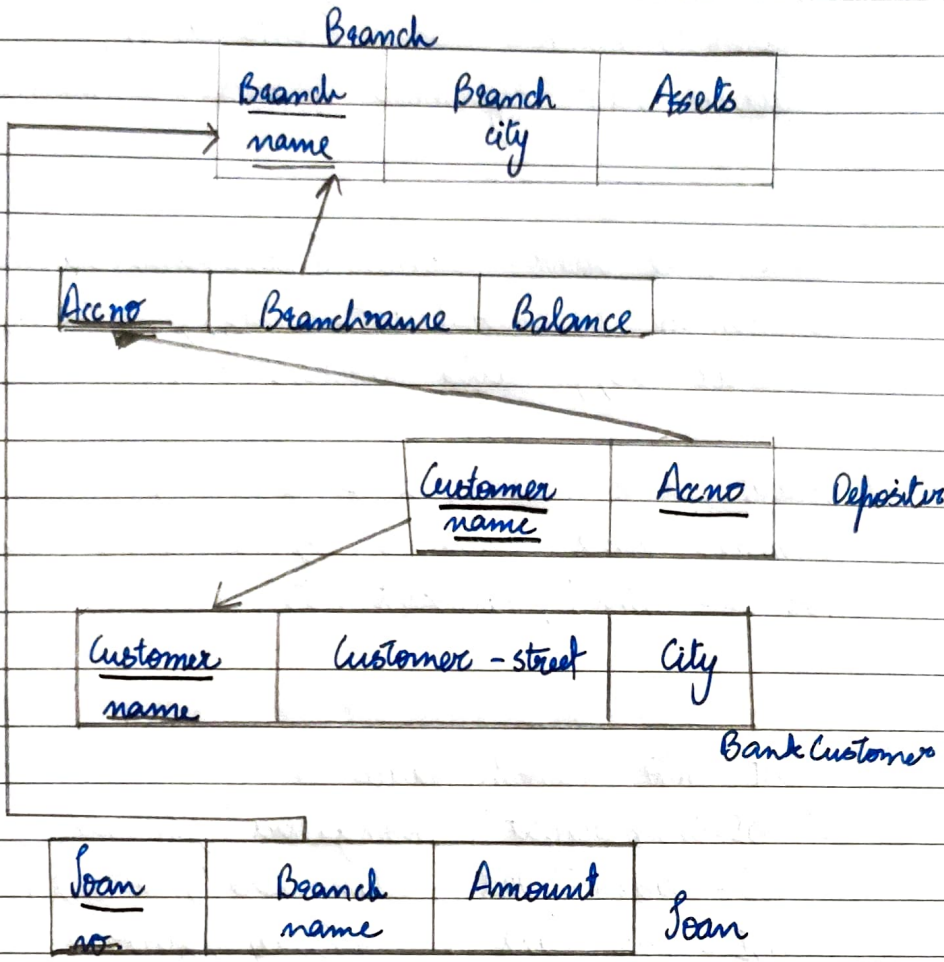
PAGE NO :
DATE :

WEEK - 3




```
create database bnkss;
use bnkss;

create table branch(
branch_name varchar(50) primary Key,
branch_city varchar(50),
assets bigint);

create table branch_account(
accno int primary key,
branch_name varchar(50) references branch(branch_name),
balance int);
```

PAGE NO :
DATE :

```
create table depositer ( cust_name varchar(50)
references bank_customer (cust_name),
accno int references branch_account(accno));

create table bank_customer (cust_name varchar (50)
references depositer (cust_name), customer_street
vdrchar (50), city varchar (40));

create table loan (loan_no int,
branch_name varchar (50) references branch
(branch_name), account int);

insert into branch value
("SBI_Chamarajpet", "bangalore", 50000),
("SBI_ResidencyRoad", "bangalore", 10000),
("SBI_ShivajiRoad", "bombay", 20000),
("SBI_ParliamentRoad", "delhi", 10000),
("SBI_jantarmantar", "delhi", 20000).

insert into branch_account values
(1, "SBI_Chamarajpet", 2000),
(2, "SBI_ResidencyRoad", 5000),
(3, "SBI_ShivajiRoad", 6000),
(4, "SBI_jantarmantar", 9000),
(5, "SBI_Parliamentroad", 8000),
(6, "SBI_ShivajiRoad", 4000),
(7, "SBI_ResidencyRoad", 4000),
(8, "SBI_ParliamentRoad", 3000),
(9, "SBI_ResidencyRoad", 5000),
(10, "SBI_jantarmantar", 2000),
(11, "SBI_ShivajiRoad", 3000),
```

PAGE NO :
DATE :

```
insert into depositer values
("Avinash", 1),
("Dinesh", 2),
("Nikil", 4),
("Ravi", 5),
("Avinash", 8),
("Nikil", 9),
("Dinesh", 10),
("Nikhil", 11),

insert into bank_customer values
("Avinash", "Bull_Temple_Road", "Banglore");
```

Q) Display the branch name and assets from all branches in lakhs of rupees and rename the assets column to 'assets in lakhs'.

→ select branch_name, assets as 'assets in lakh' from branch;

Output :-

branch name	assets in lakh
SBI_jantarmantar	20000
SBI_Parliamentroad	10000
SBI_ResidencyRoad	10000
SBI_ShivajiRoad	20000
SBI_Chamarajpet	50000

PAGE NO :
DATE :

Q.) Find all customers who have at least two accounts at the same branch (Ex: SBI - ResidencyRoad)

→ select accno, branch_name
from branch_account
group by branch_name
having count(branch_name) > 1 ;

Output :-

accno	branch_name
2	SBI_ResidencyRoad
3	SBI_ResidencyRoad
4	SBI_ResidencyRoad
5	SBI_ResidencyRoad

Q.) Create a view which gives each branch the sum of amount of all loans at the branch

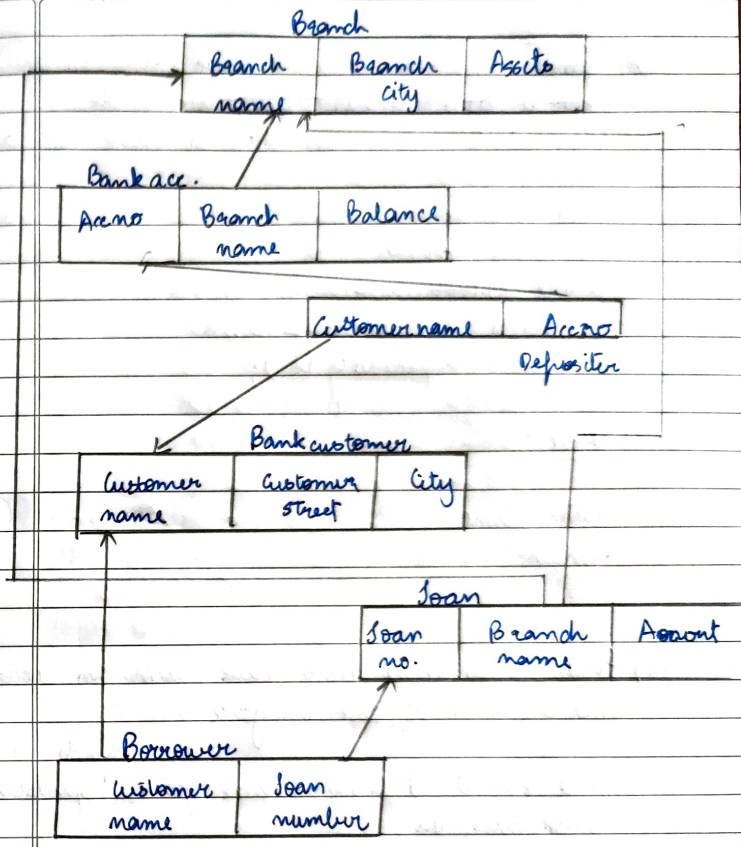
→ create view loan_sum
as select branch_name, amount
from loan
group by branch_name ;
select from loan_sum ;

O/P:-

branch_name	amount
SBI - chamarajpet	1000
SBI - ResidencyRoad	2000
SBI - Shivaji Road	5000
SBI_Parliament	4000
SBI_jantarmantar	5000

PAGE NO :
DATE :

WEEK-4



Q.) Find all the customer who have an account at all the branches located in a specific city (Ex. Delhi)

select customer_name from bankcustomer
where customer city = 'Delhi' ;

Output:-

customer_name
Nikhil
Ravi

PAGE NO :
DATE :

Q2.) Find all customers who have ~~both an account and a loan at the Banglore branch~~ loan at the bank but do not have an account.

```
Select B.customer_name
from borrower B
where B.customer_name NOT IN(
    select D.customer_name
    from depositor D );
```

Output:-

customername
mohan

Q3.) Find all customer who have both an account and a loan at the Banglore branch

```
select distinct d.customername from Depositor d, branch br,
bankaccount ba
where br.Branch_city = 'Banglore' and
br.Branch_name = ba.Branch_name and
ba.accno = d.accno and customername IN(
 select customername from Borrower));
```

Op:-

SBI_Mantrimarg

PAGE NO :
DATE :

Q4.) find names of all branches that have greater assets than all branches located in Banglore.

```
Select branch_name from branch
where assets > some(select assets from branch
                    -city = 'Bangalore');
```

O/P:-

branch_name
SBI_chamarajpet
SBI_JantarmantarRoad
SBI_ShivajiRoad

Q5.) Demonstrate how you delete all account tuples at every branch located in a specific city

```
delete ba.* from Bank_Account ba, branch b
where branch_name = b.branch_name;
select * from BankAcount;
```

Op:-

NULL

Q6.) Update balance of all accounts by 5%.

```
update bankaccount
set balance = balance * 1.05;
select * from bankaccount;
```

P.T.O.

PAGE NO :
DATE :

Q2:) Find all customers who have ~~both an account and a loan at the Banglore branch~~ loan at the bank but do not have an account.

```
select B.customer_name
from borrower B
where B.customer_name NOT IN(
    select D.customer_name
    from depositor D );
```

Output:-

customername
moham

Q3:) Find all customer who have both an account and a loan at the Banglore branch

```
select distinct d.customername from Depositor d, branch br,
bankaccount ba
where br.Branch_city = 'Banglore' and
br.Branch_name = ba.Branch_name and
ba.accno = d.accno and customername IN(
 select customername from Borrower));
```

Op:-

SBI_Mantrimarg

Q4:) Find names of all branches that have greater assets than all branches located in Bangalore.

```
Select branch_name from branch
where assets > some(select assets from branch
                    -city = 'Bangalore');
```

O/P :-

branch_name
SBI_chamarajpet
SBI_JantarmantarRoad
SBI_ShivajiRoad

Q5.) Demonstrate how you delete all account tuples at every branch located in a specific city

```
delete ba.* from Bank_Account ba, branch b
where branch_name = b.branch_name;
select * from BankAccount;
```

Op:-

NULL

Q6.) Update balance of all accounts by 5%.

```
update bankaccount
set balance = balance * 1.05;
select * from bankaccount;
```

P.T.O.

PAGE NO :
DATE :

Output :-

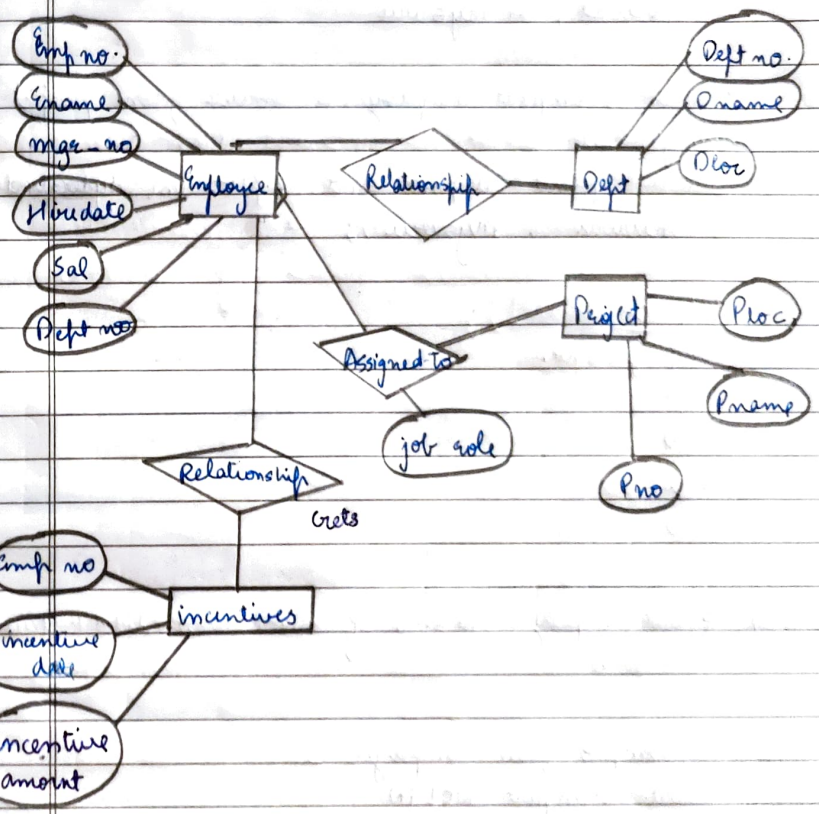
1	SBI - dharmjot	21000
2	SBI - Residency	5250
3	SBI - Shivaj	6300
4	SBI - Parliament	9450
5	SBI - Jantarmantar	8400
6	SBI - Shivaji	4200
7	SBI - Residency	4200
8	SBI - dharmraj	3150
9	SBI - Residency	5250
10	SBI - Shivaji	2100
11	SBI - Jantarmantar	4200

Dev
18/1/24

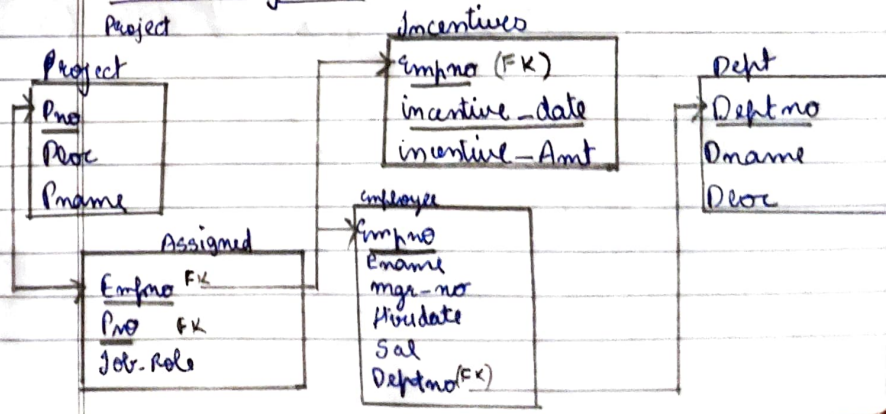
PAGE NO :
DATE : 25/1/24

LAB - 5

Employee Database:-



Schema Diagram :-



PAGE NO :
DATE :

Q3) Retrive the employee numbers of all employees who work on project located in Bengaluru, Hyderabad, or mysuru

→ select a.empno Employee_number from project p, assigned_to a
where p.pno = a.pno and p.ploc in("Hyderabad", "Bengaluru", "mysuru");

Output :-

Employee_no.
100
200
400
500

Q4) Get employee ID's of those who didn't recive incentives

→ select e.empno from employee e
where e.empno NOT IN
(select i.empno incentives i);

O/P :-

empno
700
300

PAGE NO :
DATE :

Q5) Find the employees name, number, dept, job_role, department location and project location who are working for a project location same as his/her department locations.

→ select e.ename Emp_name, e.empno Emp_Numb, d.dname Dept, a.job_role job_Role, d.dloc Department_location, p.ploc Project_location
from project p, dept d, employee e, assigned_to a
where e.empno = a.empno and p.pno = a.pno and e.deptno = d.deptno and p.ploc = d.dloc

O/P :-

emp_name	Emp_no	Dept	Job_Role	Dept_location	Proj_location
Pranay	100	Sales	Project Mang.	Bengaluru	Bengaluru

11/2/24

LAB-6 → 31/1/24

PAGE NO :
DATE :

Q1) list the name of the managers with the maximum employees

→ select e1.name
from employee e1, employee e2
where e1.empno = e2.mgr_no group by e1.ename
having count(e1.mgr_no) = (select count(e1.mgr_no)
from employee e1, employee e2 where e1.empno = e2.
group by e1.name order by count(e1.name)
desc limit 1);

o/p:-

Pranay

Q2) Display those managers name whose salary is more than average salary of his employee.

→ select m.ename from employee m
where m.empno in
(select mgr_no from employee)
and m.salary > (select avg(m.salary) from employee m
where m.mgr_no = m.empno);

o/p:-

ename
Pranay
Sakshi

Q.) Find the name of 2nd top level managers of each dept.

→ select ename from employee where empno in (
select distinct mgr_no from employee where
empno in (select distinct mgr_no from employee
where empno in (select distinct mgr_no
from employee)));

mgr_no. o/p:-

Pranay
Mahima

Q.) Employee details of who got second maximum incentive in January 2019.

→ select * from employee where empno = (select
i.empno from incentives i where i.incentive_amt =
(select max(in.incentive_amount) from incentives in
where in.incentive_amount < (select max(inc.incentive_
amount) from incentive inc where inc.incentive_date
between '2019-01-01' and '2019-12-31') and
incentive_date between '2019-01-01' and '2019-12-31'
));

o/p:-

empno	ename	mgrno	hire_date	sale	dept_no
500	Vishith	400	2004-03-05	3000	40

PAGE NO :
DATE :

Q:) Employee who is working in the same dept as the manager.

→ select e2.name
from employee e1, employee e2
where e1.empno = e2.mgr_no and e1.deptno = e2.deptno;

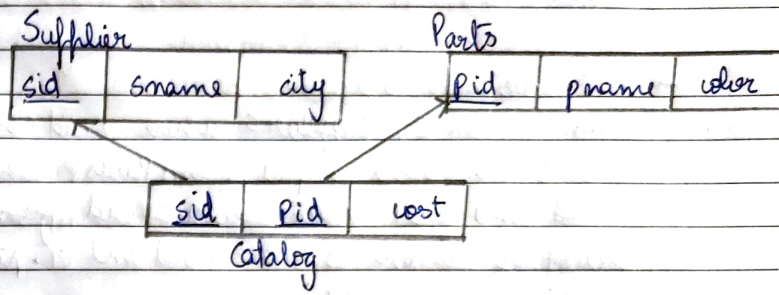
op:-

Nithith

1/2/24

PAGE NO :
DATE : /

~~Week~~ Lab-7:- 1/2/24



Q:) Find the pnames of parts for which there is some supplier.

→ select distinct p.name
from Parts p, Catalog c
where p.pid = c.pid;

op:-
Book
Pen
Pencil
Mobile
Charger

Q:) Find snames of suppliers who suffly every part

→ selet distinct s.name
from Catalog C, Supplier S where C.sid = S.sid and not exists (select P.pid from parts P where not exists (select C1.sid from Catalog C1 where C1.sid = C.sid and C1.pid = P.pid));

op:-
Acme Widget

PAGE NO :
DATE :

Q.) Find the suppliers name who supply every red part.

→ select distinct S.sname
from Catalog C, Suppliers S where C.sid = S.sid and NOT EXISTS (select P.pid from Parts P where P.color = "Red" and NOT EXISTS (select C1.sid from Catalog C1 where C1.sid = C.sid and C1.pid = P.pid and P.color = "Red"));

o/p:-
Acme Widgets
Johns

Q.) Find the pnames supplied by Acme Widgets

→ select P.pname
from Parts P, Catalog C, Supplier S
where P.pid = C.pid and C.sid = S.sid and S.sname = "Acme Widget" and NOT EXISTS (select * from Catalog C1, Suppliers S1 where P.pid = C.pid and C1.sid = S1.sid and S1.sname = "Acme Widget");

o/p:-
Mobile
Charger

29/2/24

PAGE NO :
DATE :

LAB - 8

Q 1.) Create a database "student" with the following attributes Roll no, Age, ContactNo, Email Id
2.) Insert appropriate value
3.) Write query to update Email-Id of a student with rollno 10
4.) Replace the student name from "ABC" to "FEM" of roll no 11
5.) Export the created table into local file system
6.) Drop the table
7.) Import a given csv dataset from local file system into mongodb collection.

Q1.) → db.createCollection("Student");

Q2.) → db.Student.insert({RollNo:1, Age:21, Cont:9876, email:"antara.de@gmail.com"});
db.Student.insert({RollNo:2, Age:22, Cont:9976, email:"anuska.de@gmail.com"});

Q3.) db.Student.update({RollNo:10}, {$set:{email:"abhinav@gmail"}});

Q4.) db.Student.insert({RollNo:11, Age:22, Name:"ABC", Count:2276, email:"reo.de@gmail.com"});
db.Student.update({RollNo:11, Name:"ABC"}, {$set:{Name:"FEM"}});

PAGE NO :
DATE :

Q5) Execute at the command prompt:

mongoexport mongodb+srv://antarare:****@cluster0.mfxfeys.mongodb.net/myDB --collection=student --out c:\Users\BMSCECSE\Downloads\output.json

Q6) db.Student.drop();

Q7) At the command prompt:-

mongoimport mongodb+srv://antarare:Test1234@cluster0.mfxfeys.mongodb.net/myDB --collection=New_Student --type json --file c:\Users\BMSCECSE\Downloads\output.json

At the mongosh

Execute:-

db.New_Student.find()

PAGE NO :
DATE :

LAB-9

Perform the following DB operations using MongoDB

1) Create a collection by name Customer with the following attributes Cust_id, Acc_Bal, Acc_Type
2) Insert at least 5 values into the table.
3) Write a query to display those records whose total account balance is greater than 1200 of account type 'z' for each customer_id.
4) Determine the Minimum & Maximum acc bal for each cust_id.
5) Export the created collection into local file system.
6) Drop the Table
7) Import a given CSV dataset from local file system into mongodb collection

1) db.createCollection("customer");

2) db.customers.insertMany([
{Cust_Id:"1", Acc_Bal:1500, Acc_Type:"Z"},
{Cust_Id:"2", Acc_Bal:1000, Acc_Type:"Z"},
{Cust_Id:"3", Acc_Bal:1300, Acc_Type:"Z"},
{Cust_Id:"4", Acc_Bal:1005, Acc_Type:"Z"},
])

2) db.customers.aggregate([
{$match:{Acc_Type:"Z"}},
{$group:{_id:"$cust_id", totalBalance:{$sum:"$Acc_Bal"}}},
{$match:{totalBalance:{$gt:1200}}},
])

PAGE NO :
DATE :

4.) db.Customers.aggregate([
{ $group: { _id: "$cust_id", minBalance: { $min: "$Acc_Bal" }, maxBalance: { $max: "$Acc_Bal" } } }
])

5.) mongoexport mongodb+srv://antarara:*****@cluster0.mfnfeys.mongodb.net/myDB --collection=Customer --out C:\Users\BMSCECSE\Downloads\output.json

6.) db.Customer.drop();

7.) At the cmd :-
mongoimport mongodb+srv://antarara:Test1234@cluster0.mfnfeys.mongodb.net/myDB --collection=New_Customer --type json --file C:\Users\BMSCECSE\Downloads\output.json

At the mongosh end :-
db.New_Student.find()

Sun 29/1/24