VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Database Management Systems (23CS3PCDBM)

Submitted by

DISHA H JAIN (1BM23CS095)

in partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
December-2023 to Feb-2024
B. M. S. College of Engineering,
Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled “Database Management Systems (22CS3PCDBM)” carried out by **DISHA H JAIN (1BM23CS095),** who is a bonafide student of **B. M. S. College of Engineering.** It is in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2024. The Lab report has been approved as it satisfies the academic requirements in respect of a Database Management Systems (23CS3PCDBM) work prescribed for the said degree.

Dr. Kayarvizhi N
Associate Professor
Department of CSE, BMSCE

Dr. Kavitha Soodi
Professor HOD
Department of CSE, BMSCE

`

Index

Insurance Database

Question

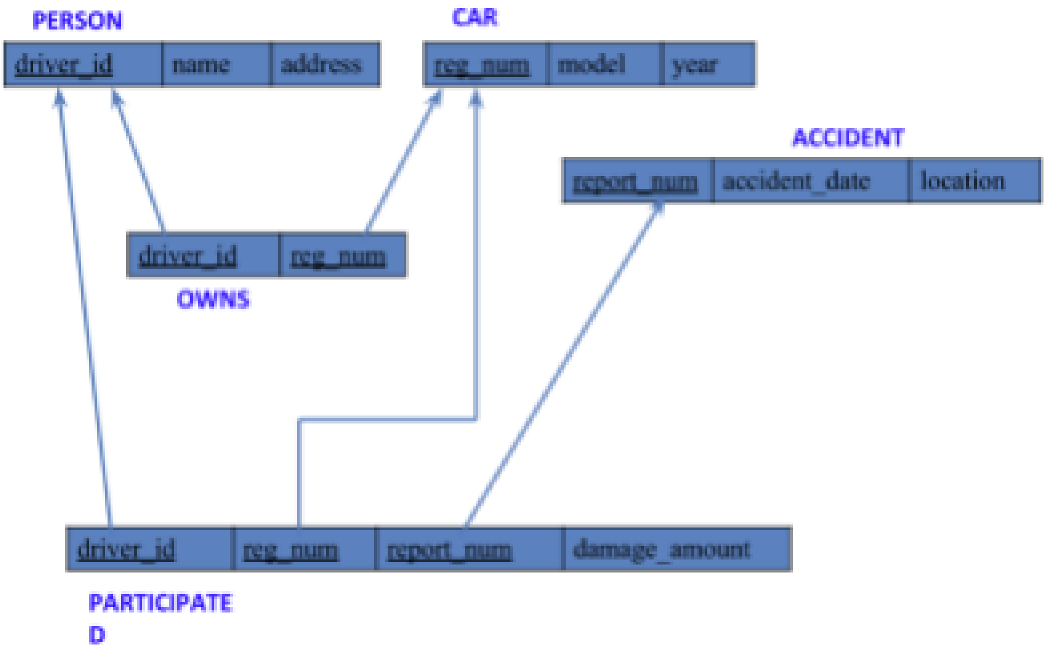
(Week 1)

- PERSON (driver_id: String, name: String, address: String)
- CAR (reg_num: String, model: String, year: int)
- ACCIDENT (report_num: int, accident_date: date, location: String)
- OWNS (driver_id: String, reg_num: String)
- PARTICIPATED (driver_id: String,reg_num: String, report_num: int, damage_amount: int)

- Create the above tables by properly specifying the primary keys and the foreign keys. - Enter at least five tuples for each relation

- Display Accident date and location
- Update the damage amount to 25000 for the car with a specific reg_num (example 'K A031181') for which the accident report number was 12.
- Add a new accident to the database.
- To Do
- Display Accident date and location
- Display driver_id who did accident with damage amount greater than or equal to Rs.25000

Schema Diagram



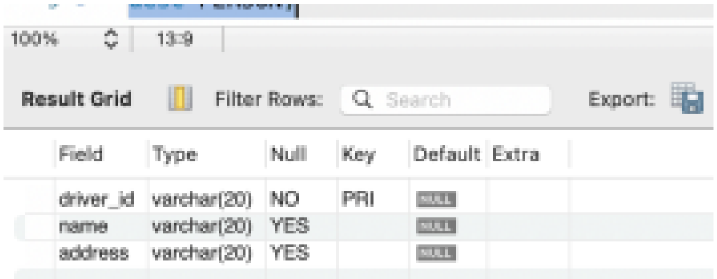
Create database

```
create database dharu;
use dharu;
```

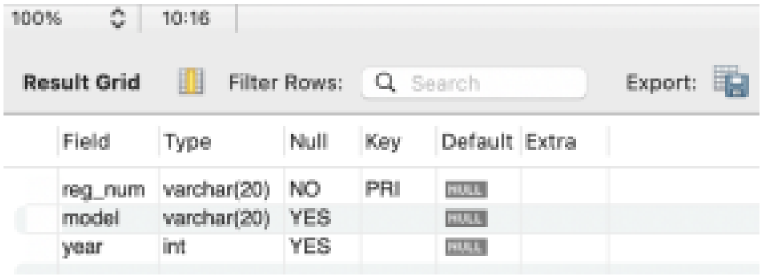
Create table

```
create table PERSON(
driver_id varchar(20),
name varchar(20),
address varchar(20),
primary key(driver_id)
);
desc PERSON;
create table CAR(
reg_num varchar(20),
model varchar(20),
year int,
primary key(reg_num)
);
desc CAR;
create table ACCIDENT(
report_num int,
accident_date date,location varchar(20),
primary key(report_num)
);
create table OWNS(
driver_id varchar(20),
reg_num varchar(20),
primary key(driver_id,reg_num),
foreign key(driver_id) references PERSON(driver_id),
foreign key (reg_num) references CAR(reg_num)
);
create table PARTICIPATED(
driver_id varchar(20),
reg_num varchar(20),
report_num int,
damage_amt int,
primary key(driver_id,reg_num,report_num),
foreign key(driver_id) references PERSON (driver_id),
foreign key (reg_num) references CAR(reg_num),
foreign key (report_num) references ACCIDENT (report_num)
);
```

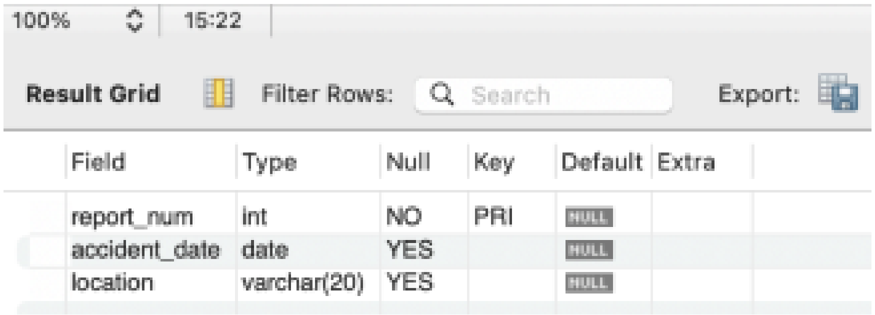
Structure of the table



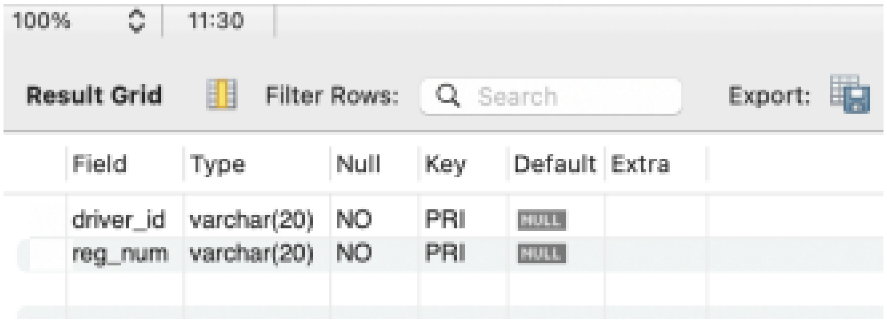
Field	Type	Null	Key	Default	Extra
driver_id	varchar(20)	NO	PRI	NULL	
name	varchar(20)	YES		NULL	
address	varchar(20)	YES		NULL	



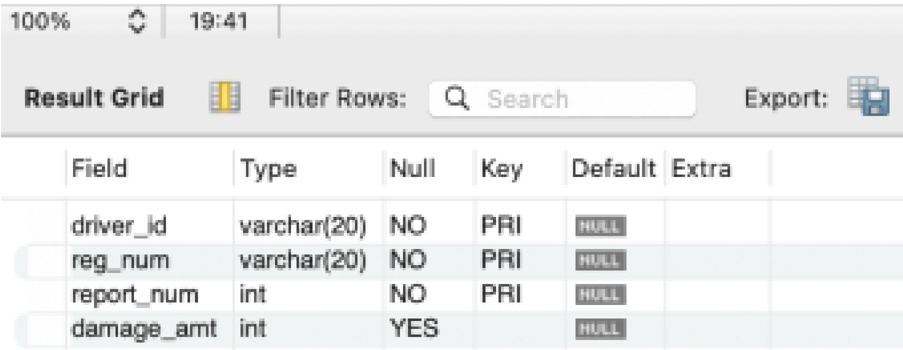
Field	Type	Null	Key	Default	Extra
reg_num	varchar(20)	NO	PRI	NULL	
model	varchar(20)	YES		NULL	
year	int	YES		NULL	



Field	Type	Null	Key	Default	Extra
report_num	int	NO	PRI	NULL	
accident_date	date	YES		NULL	
location	varchar(20)	YES		NULL	



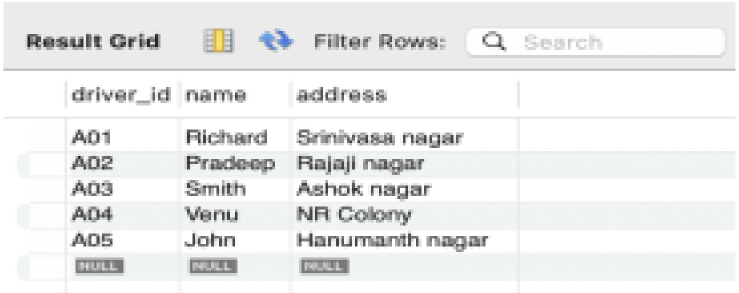
Field	Type	Null	Key	Default	Extra
driver_id	varchar(20)	NO	PRI	NULL	
reg_num	varchar(20)	NO	PRI	NULL	



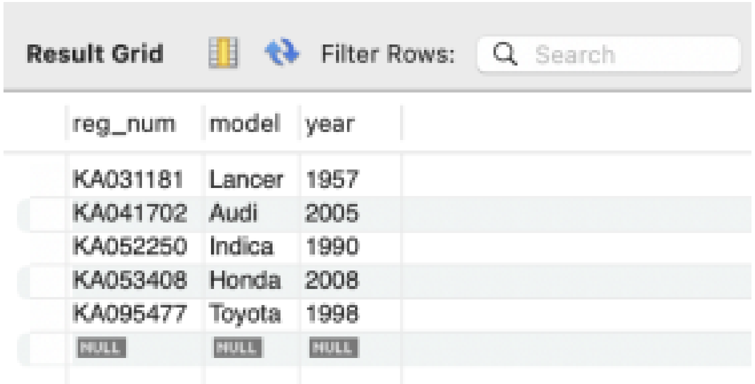
Field	Type	Null	Key	Default	Extra
driver_id	varchar(20)	NO	PRI	NULL	
reg_num	varchar(20)	NO	PRI	NULL	
report_num	int	NO	PRI	NULL	
damage_amt	int	YES		NULL	

Inserting Values into the table

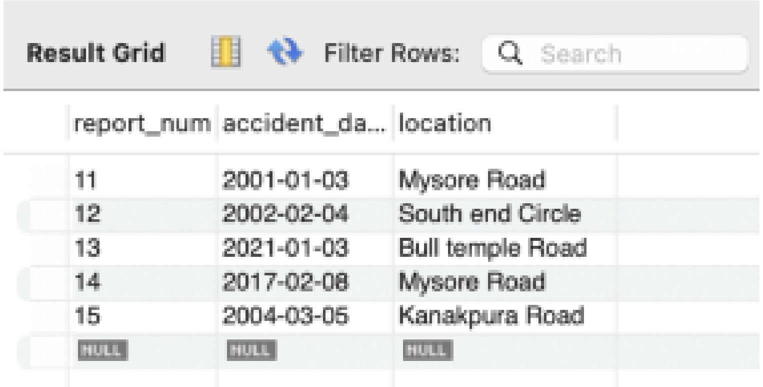
```
insert into PERSON values("A01","Richard","Srinivasa nagar");
insert into PERSON values("A02","Pradeep","Rajaji nagar");
insert into PERSON values("A03","Smith","Ashok nagar");
insert into PERSON values("A04","Venu","NR Colony");
insert into PERSON values("A05","John","Hanumanth nagar");
insert into CAR values("KA052250","Indica",1990);
insert into CAR values("KA031181","Lancer",1957);
insert into CAR values("KA095477","Toyota",1998);
insert into CAR values("KA053408","Honda",2008);
insert into CAR values("KA041702","Audi",2005);
insert into ACCIDENT values(11,"01-01-03","Mysore Road");
insert into ACCIDENT values(12,"02-02-04","South end Circle");
insert into ACCIDENT values(13,"21-01-03","Bull temple Road");
insert into ACCIDENT values(14,"17-02-08","Mysore Road");
insert into ACCIDENT values(15,"04-03-05","Kanakpura Road");
insert into OWNS values("A01","KA052250");
insert into OWNS values("A02","KA053408");
insert into OWNS values("A03","KA031181");
insert into OWNS values("A04","KA095477");
insert into OWNS values("A05","KA041702");
insert into PARTICIPATED values("A01","KA052250",11,10000);
insert into PARTICIPATED values("A02","KA053408",12,50000);
insert into PARTICIPATED values("A03","KA095477",13,25000);
insert into PARTICIPATED values("A04","KA031181",14,3000);
insert into PARTICIPATED values("A05","KA041702",15,5000);
select * from PERSON;
select * from CAR;
select * from ACCIDENT;
select * from OWNS;
select * from PARTICIPATED;
```



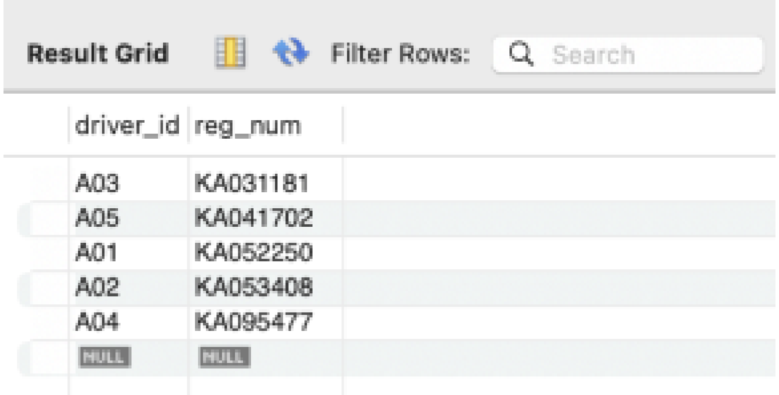
driver_id	name	address
A01	Richard	Srinivasa nagar
A02	Pradeep	Rajaji nagar
A03	Smith	Ashok nagar
A04	Venu	NR Colony
A05	John	Hanumanth nagar
NULL	NULL	NULL



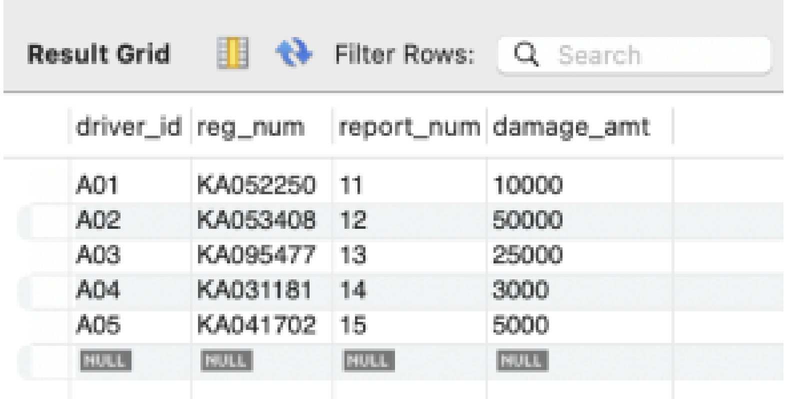
reg_num	model	year
KA031181	Lancer	1957
KA041702	Audi	2005
KA052250	Indica	1990
KA053408	Honda	2008
KA095477	Toyota	1998
NULL	NULL	NULL



report_num	accident_da...	location
11	2001-01-03	Mysore Road
12	2002-02-04	South end Circle
13	2021-01-03	Bull temple Road
14	2017-02-08	Mysore Road
15	2004-03-05	Kanakpura Road
NULL	NULL	NULL



driver_id	reg_num
A03	KA031181
A05	KA041702
A01	KA052250
A02	KA053408
A04	KA095477
NULL	NULL

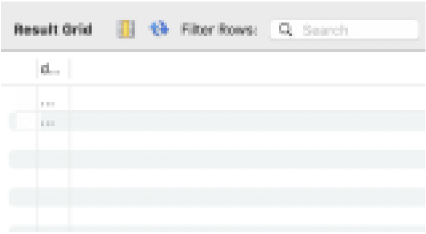


driver_id	reg_num	report_num	damage_amt
A01	KA052250	11	10000
A02	KA053408	12	50000
A03	KA095477	13	25000
A04	KA031181	14	3000
A05	KA041702	15	5000
NULL	NULL	NULL	NULL

Queries

Display driver id who did accident with damage amount greater than or equal to Rs.25000

```
select driver_id

from PARTICIPATED

where damage_amt>=25000;
```



Add a new accident to the database.

insert into ACCIDENT values(16,"15-03-08","Domlur");

select * from ACCIDENT;

Display Accident date and location

select accident_date date,location
from ACCIDENT;

More Queries on Insurance Database

Question (Week 2)

1.LIST THE ENTIRE PARTICIPATED RELATION IN THE DESCENDING ORDER OF DAMAGE AMOUNT.

2.FIND THE AVERAGE DAMAGE AMOUNT.

3.DELETE THE TUPLE FROM PARTICIPATED RELATION WHOSE DAMAGE AMOUNT IS BELOW THE AVERAGE DAMAGE AMOUNT.

4.LIST THE NAME OF DRIVERS WHOSE DAMAGE IS GREATER THAN THE AVERAGE DAMAGE AMOUNT.

5.FIND MAXIMUM DAMAGE AMOUNT .

Bank Database

Question (Week 3)

- Branch (branch-name: String, branch-city: String, assets: real)

- BankAccount(accno: int, branch-name: String, balance: real)

- BankCustomer (customer-name: String, customer-street: String, customer-city: String) - Depositer(customer-name: String, accno: int)

- LOAN (loan-number: int, branch-name: String, amount: real)

Schema Diagram

Create database

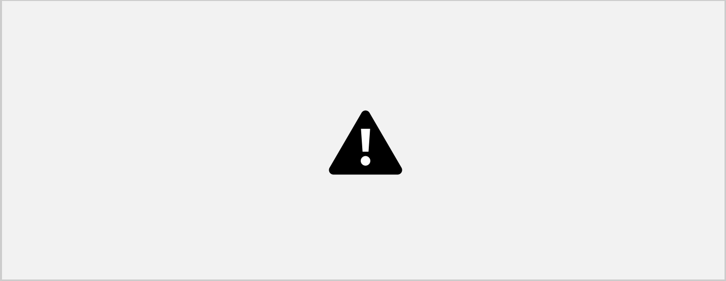
```
create database bank_204;
use bank_204;
```

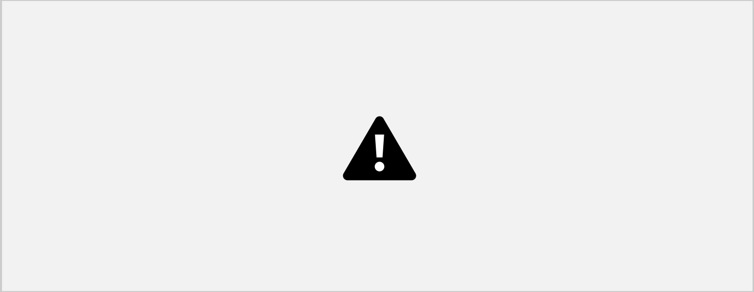
Create table

```
create table BRANCH(
branchname varchar(20),
branchcity varchar(20),
assets float,
primary key (branchname)
);
desc BRANCH;


create table BANKACCOUNT(
accno int,
branchname varchar(20),
balance float,
primary key(accno),
foreign key (branchname) references BRANCH (branchname)
);
desc BANKACCOUNT;
```

```
create table BANKCUSTOMER(
customername varchar(20),
customerstreet varchar(20),
city varchar(20),
primary key(customername)
);
desc BANKCUSTOMER;

create table DEPOSITER(
customername varchar(20),
accno int,
primary key(customername,accno),
foreign key (customername) references BANKCUSTOMER (customername),
foreign key (accno) references BANKACCOUNT (accno)
);
desc DEPOSITER;

create table LOAN(
loannumber int,
branchname varchar(20),
amount float,
primary key(loannumber),
foreign key (branchname) references BRANCH (branchname)
);
desc LOAN;
```

Structure of the table





Inserting Values to the table

```
insert into BRANCH values("SBI_Chamrajpet","Bangalore",50000);
insert into BRANCH values("SBI_ResidencyRoad","Bangalore",10000);
insert into BRANCH values("SBI_ShivajiRoad","Bombay",20000);
insert into BRANCH values("SBI_ParlimentRoad","Delhi",10000);
insert into BRANCH values("SBI_Jantarmantar","Delhi",20000);
select * from BRANCH;

insert into BANKACCOUNT values(1,"SBI_Chamrajpet",2000);
insert into BANKACCOUNT values(2,"SBI_ResidencyRoad",5000);
insert into BANKACCOUNT values(3,"SBI_ShivajiRoad",6000);
insert into BANKACCOUNT values(4,"SBI_ParlimentRoad",9000);
insert into BANKACCOUNT values(5,"SBI_Jantarmantar",8000);
insert into BANKACCOUNT values(6,"SBI_ShivajiRoad",4000);
insert into BANKACCOUNT values(8,"SBI_ResidencyRoad",4000);
insert into BANKACCOUNT values(9,"SBI_ParlimentRoad",3000);
insert into BANKACCOUNT values(10,"SBI_ResidencyRoad",5000);
insert into BANKACCOUNT values(11,"SBI_Jantarmantar",2000);
select * from BANKACCOUNT;
```

```
insert into BANKCUSTOMER values("Avinash","Bull_Temple_Road","Bangalore");
insert into BANKCUSTOMER values("Dinesh","Bannergatta_Road","Bangalore");
insert into BANKCUSTOMER values("Mohan","NationalCollege_Road","Bangalore");
insert into BANKCUSTOMER values("Nikhil","Akbar_Road","Delhi"); insert into
BANKCUSTOMER values("Ravi","Prithviraj_Road","Delhi"); select * from
BANKCUSTOMER;

insert into DEPOSITER values("Avinash",1);
insert into DEPOSITER values("Dinesh",2);
insert into DEPOSITER values("Nikhil",4);
insert into DEPOSITER values("Ravi",5);
insert into DEPOSITER values("Avinash",8);
insert into DEPOSITER values("Nikhil",9);
insert into DEPOSITER values("Dinesh",10);
insert into DEPOSITER values("Nikhil",11);
select * from DEPOSITER;

insert into LOAN values(1,"SBI_Chamrajpet",1000);
insert into LOAN values(2,"SBI_ResidencyRoad",2000);
insert into LOAN values(3,"SBI_ShivajiRoad",3000);
insert into LOAN values(4,"SBI_ParlimentRoad",4000);
insert into LOAN values(5,"SBI_Jantarmantar",5000);
select * from LOAN;
```

Queries

- **Display the branch name and assets from all branches and rename the assets column to 'assets in lakhs'.**

```
select branchname, assets/100000 as "Assetsinlakhs"
from BRANCH;
```

- **Find all the customers who have at least two accounts at the same branch (ex.SBI_ResidencyRoad).**

```
select d.customername
from DEPOSITER d, BANKACCOUNT b
where b.branchname='SBI_ResidencyRoad'
and d.accno=b.accno
group by d.customername
having count(b.accno)>=2;
```

More Queries on Bank Database

Question (Week 4)

Queries

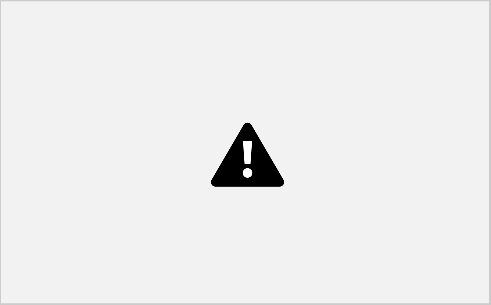
1. Retrieve all branches and their respective total
assets select branchname, sum(assets) as total_assets
from branch
group by branchname;

2. List all customers who live in a particular city
select customername, city
from bankcustomer
where city="bangalore";

3. List all customers with their account numbers
select distinct customername, accno
from depositer;

4. List all customers with their loan amounts

```
select d.customername, sum(l.amount)
from depositer d, bankaccount ba, loan l
where d.accno=ba.accno and ba.branchname=l.branchname
group by d.customername;
```



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

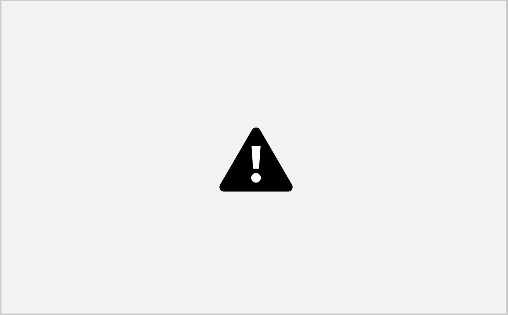
```
select d.customername, ba.city
from depositer d, bankcustomer ba
where d.customername=ba.customername and ba.city="delhi"
group by d.customername;
```

6. Find all customers who have accounts with a balance greater than a specified amount (100000)

```
select d.customername, ba.balance
from depositer d, bankaccount ba
where d.accno=ba.accno
having ba.balance>100000;
```

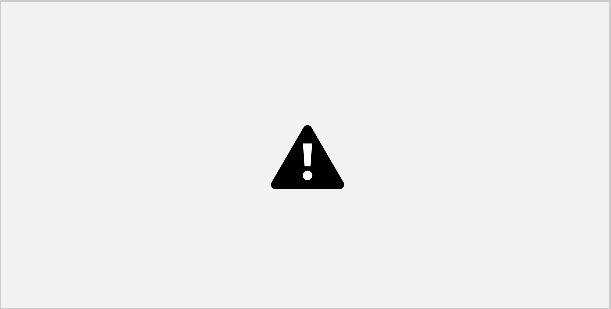
7. List all customers who have both a loan and an account at the same branch

```
select distinct d.customername
from depositer d, bankaccount ba, loan l
where d.accno=ba.accno and l.branchname=ba.branchname;
```



8. Get the number of accounts held at each branch

```
 select branchname, count(accno)
 from bankaccount
 group by branchname;
```



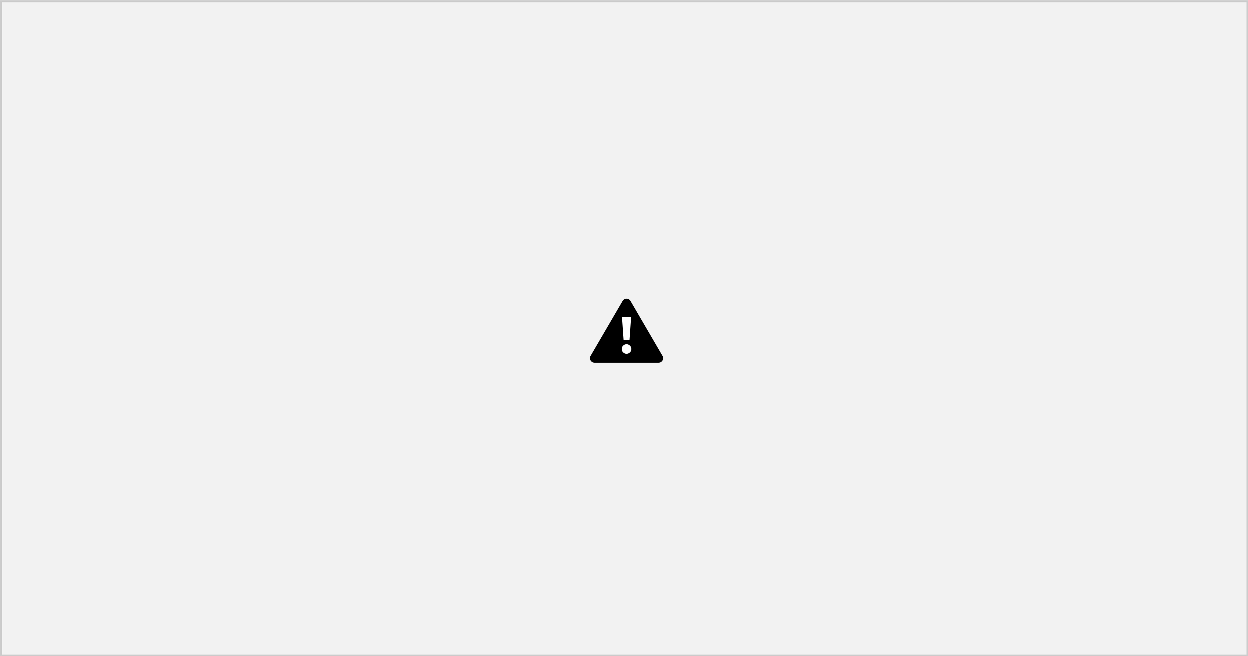
9. Find all branches that have no loans issued

```
select b.branchname
from branch b
left join loan l on b.branchname=l.branchname
where l.loannumber is null;
```

Employee Database

Question (Week 5)

Schema Diagram



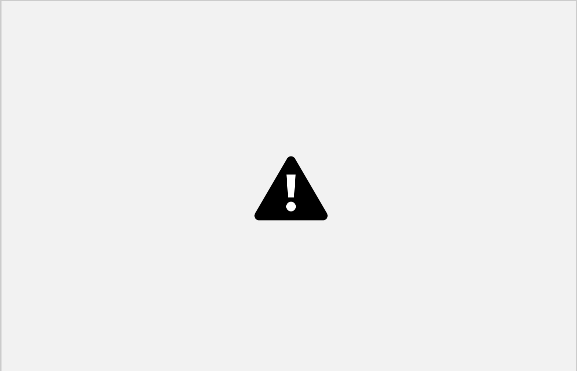
Create database

```
create database employee_database_204;
use employee_database_204;
```

Create table

```
create table project(
pno int ,
ploc varchar(30),
pname varchar(30),
primary key(pno)
);
create table dept(
deptno int,
dname varchar(30),
dloc varchar(30),
primary key(deptno)
);
create table employees(
empno int ,
ename varchar(30),
mgr_no int,
 hireddate date,
 sal int,
 deptno int,
 primary key (empno),
 foreign key(deptno) references dept(deptno)
 );
create table assigned_to(
empno int,
pno int,
jobrole varchar(30),
foreign key(pno) references project(pno), foreign
key(empno) references employees(empno) );
create table incentives(
empno int,
incentivedate date,
incentiveamt int,
primary key(incentivedate),
foreign key (empno) references employees(empno)
);
desc project;
desc employees;
desc dept;
desc assigned_to;
desc incentives;
```

Structure of the table



Inserting Values to the table

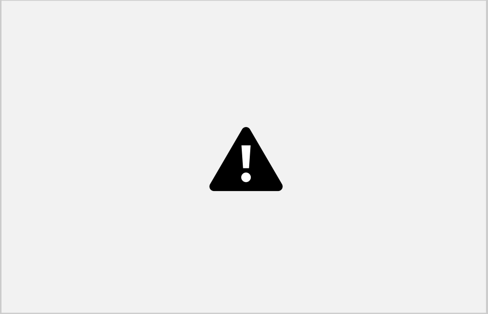
```
insert into project values(1,'bengaluru','syntax');
insert into project values(2,'gujarat','rolex');
insert into project values(3,'mysuru','hybrid');
insert into project values(4,'hyderabad','hydro');
insert into project values(5,'westbengal','solar');

insert into dept values(10,'sales','bengaluru');
insert into dept values(20,'finance','westbengal');
insert into dept values(30,'marketing','bihar');
insert into dept values(40,'research_and_development','delhi');
insert into dept values(50,'management','hyderabad');

insert into employees values(100,'pranathi',321,'2003-01-01',100000,10);
insert into employees values(101,'pranav',322,'2004-02-07',350000,20);
insert into employees values(102,'pranay',323,'2004-10-13',60000,20);
insert into employees values(103,'prerana',324,'2003-04-11',110000,50);
insert into employees values(104,'priya',325,'2003-08-02',10000,40);
insert into employees values(105,'pratham',326,'2005-11-18',100000,10);
insert into employees values(106,'prem',327,'2003-08-01',100000,30);
insert into employees values(107,'pranitha',328,'2002-07-12',100000,50);
insert into employees values(108,'pramod',329,'2003-01-16',100000,30);
insert into employees values(109,'prakash',3330,'2005-12-31',100000,50);

insert into assigned_to values(100,1,"projectmanager");
insert into assigned_to values(108,4,"projectsupervisor");
insert into assigned_to values(104,3,"financer");
insert into assigned_to values(103,2,"advisor");
insert into assigned_to values(109,5,"projecthead");

insert into incentives values(105,'2005-11-18',6000);
insert into incentives values(100,'2003-01-01',5000);
insert into incentives values(107,'2002-07-12',3500);
insert into incentives values(103,'2003-04-11',5900);
insert into incentives values(108,'2003-01-16',4200);

select * from project;
```

select * from dept;

select * from employees;

select * from incentives;

select * from assigned_to;



Queries

1. Retrieve 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');
```

2.Get Employee ID’s of those employees who didn’t receive incentives

```
select e.empno

from employees e

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

3. Write a SQL query to 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 location.

```
select e.ename emp_name,e.empno emp_number,d.dname dept, a.jobrole job_role,d.dloc
dept_location,p.ploc project_location

from project p,dept d,employees e,assigned_to a
```

```
where e.empno=a.empno and p.pno=a.pno and e.deptno=d.deptno and p.ploc=d.dloc;
```

More Queries on Employee Database

Question (Week 6)

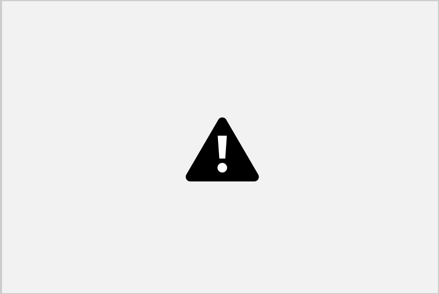
Queries

1.List all employees along with their project details (if assigned)

```
select e.ename, p.pname,p.ploc,p.pno
 from employees e, project p, assigned_to a
 where e.empno=a.empno and p.pno=a.pno;
```

2.Find all employees who received incentives, along with the total incentive amount

```
select e.ename, sum(incentiveamt) as totalamount
 from employees e, incentives i
 where e.empno=i.empno
 group by ename;
```

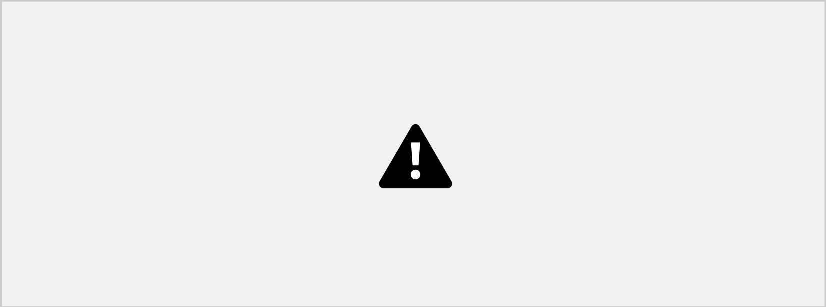


3.Retrieve the project names and locations of projects with employees assigned as 'Manager'

```
select p.pname,p.ploc
 from project p, assigned_to a
 where p.pno=a.pno and a.jobrole="projectmanager";
```

4.List departments along with the number of employees in each department

```
select d.dname, count(empno) as totalemployee
 from dept d, employees e
 where d.deptno=e.deptno
 group by e.deptno;
```



5.Find employees who have not been assigned to any project

```
select e.ename
 from employees e
 where not exists(select 1
 from assigned_to a
 where e.empno=a.empno);
```

6.List all employees along with their department names and location

```
select e.ename, d.dname, d.dloc
 from employees e, dept d
 where d.deptno=e.deptno;
```

7.Retrieve the details of employees who work under a specific manager (e.g., manager with empno = 104)

```
select e.ename
 from employees e, assigned_to a
 where e.empno=a.empno and e.empno=104;
```

8.List all projects that have employees assigned and the number of employees on each project.

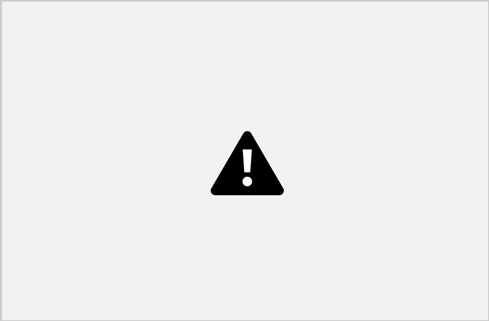
```
select p.pname, count(a.empno) as totalemployee
 from project p, assigned_to a
 where p.pno=a.pno
 group by p.pname;
```

9.List the total number of incentives given to each employee and the sum of incentives for each.

```
select e.ename, count(i.empno) as numberofincentive , sum(i.incentiveamt)
from incentive i, employees e
where e.empno=i.empno
group by e.empno;
```

10.Retrieve all employees who have the role of 'Developer' on any project.

```
select e.ename
from employees e, assigned_to a
where e.empno=a.empno and jobrole="projecthead";
```



11.Display the department-wise average salary of employees.

```
select d.dname, avg(e.sal)
from dept d, employees e
where d.deptno=e.deptno
group by d.dname;
```