DBMS LAB REPORT

NAME:VAIBHAVI PATIL
USN:1BM19CS217
CSE SECTION ‘A’

PROGRAM 1: INSURANCE DATABASE

Consider the Insurance database given below.
The data types are specified.
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)

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

```
create database insudb;
use insudb;

create table PERSON(
driver_id varchar(10) NOT NULL,name varchar(10) NOT NULL,address varchar(20) NOT
NULL,primary key (driver_id)
);
create table  CAR(
Regno varchar(10) NOT NULL,model varchar(10) NOT NULL,year int NOT NULL,primary key
(Regno)
);
create table ACCIDENT(
report_number int NOT NULL,accdate date,location varchar(20),primary key (report_number)
);
create table OWNS(
driver_id varchar(10),Regno varchar(10),primary key (driver_id,Regno),foreign key (driver_id)
references PERSON (driver_id),foreign key (Regno) references CAR (regno)
);
create table PARTICIPATED(
driver_id varchar(10),Regno varchar(10),report_number int,damage_amount int,primary key
(driver_id,Regno,report_number),foreign key (driver_id) references PERSON (driver_id),foreign
key (Regno) references car (Regno),
foreign key (report_number) references ACCIDENT (report_number)
);

insert into PERSON values("A01","Richard","Srinivas nagar"),
("A02","Pradeep","Rajaji nagar"),
("A03","Smith","Ashok nagar"),
("A04","Venu","NR Colony"),
("A05","Jhon","Hanumanth nagar");
```

```
insert into CAR values("KA052250","Indica","1990"),
("KA031181","Lancer","1957"),
("KA095477","Toyota","1998"),
("KA053408","Honda","2008"),
("KA041702","Audi","2005");


insert into OWNS VALUES("A01","KA052250"),
("A02","KA053408"),
("A03","KA031181"),
("A04","KA095477"),
("A05","KA041702");

insert into ACCIDENT values("11","2003-01-01","Mysore Road"),
("12","2004-02-02","Southend circle"),
("13","2003-01-21","Bull temple road"),
("14","2008-02-17","Mysore Road"),
("15","2005-03-04","Kanakpura road");

insert into PARTICIPATED values ("A01","KA052250","11","10000"),
 ("A02","KA053408","12","50000"),
 ("A03","KA095477","13","25000"),
 ("A04","KA031181","14","3000"),
 ("A05","KA041702","15","5000");
```

select * from person;

PERSON

Result Grid | Filter Rows:

driver_id	name	address
A01	Richard	Srinivas nagar
A02	Pradeep	Rajaji nagar
A03	Smith	Ashok nagar
A04	Venu	NR Colony
A05	Jhon	Hanumanth nagar

person 3

select * from car;

CAR

Result Grid | Filter Rows:

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

Select * from owns;

OWNS

driver_id	Regno
A03	KA031181
A05	KA041702
A01	KA052250
A02	KA053408
A04	KA095477

OWNS 4

Select * from accident;

ACCIDENT

Result Grid | Filter Rows: | Edit

report_number	accdate	location
11	2003-01-01	Mysore Road
12	2004-02-02	Southend circle
13	2003-01-21	Bull temple road
14	2008-02-17	Mysore Road
15	2005-03-04	Kanakpura road

accident 1

Select * from participated;

PARTICIPATED

driver_id	Regno	report_number	damage_amount
A01	KA052250	11	10000
A02	KA053408	12	50000
A03	KA095477	13	25000
A04	KA031181	14	3000
A05	KA041702	15	5000

participated 5

iii)Demonstrate how you

a.Update the damage amount to 25000 for the car with a specific reg-num(example 'K A053408') for which the accident report number was 12.

Ans : update participated set damage_amount = 25000 where report_number = 12 and Regno = "KA053408";

driver_id	Regno	report_number	damage_amount
A01	KA052250	11	10000
A02	KA053408	12	25000
A03	KA095477	13	25000
A04	KA031181	14	3000
A05	KA041702	15	5000
NULL	NULL	NULL	NULL

b.Add a new accident to the database.

Ans : insert into ACCIDENT values("16","2001-11-09","Majestic circle");

Result Grid | Filter Rows: | Edit

report_number	accdate	location
11	2003-01-01	Mysore Road
12	2004-02-02	Southend circle
13	2003-01-21	Bull temple road
14	2008-02-17	Mysore Road
15	2005-03-04	Kanakpura road
16	2001-11-09	Majestic circle
NULL	NULL	NULL

iv)Find the total number of people who owned cars that were involved in accidents in 2008.

Ans : select count(*) from person p,accident ac,participated pa where (p.driver_id=pa.driver_id) and (ac.report_number=pa.report_number) and (accdate like "2008%");

Result Grid | Filter Rows:

no_of_accidents_in_2008
1

Result 10

v)Find the number of accidents in which cars belonging to a specific model (example Indica)were involved.

select count(*) as accident_beloging_to_model_Indica from car c,accident ac,participated pa where(c.Regno=pa.Regno) and (ac.report_number=pa.report_number) and c.model="Indica";

accident_beloging_to_model_Indica
1

PROGRAM 2: BANKING ENTERPRISE DATABASE

Consider the following database for a banking enterprise.

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)

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

```
create database bankingdatabase1;
use bankingdatabase1;
create table BRANCH(
branch_name varchar(20) NOT NULL,branch_city varchar(20) NOT NULL,assets real,primary
key(branch_name)
);
create table BANKACCOUNT(
accno int NOT NULL,branch_name varchar(20) NOT NULL,balance real,primary key(accno),
foreign key (branch_name) references BRANCH(branch_name)on delete cascade
);

create table BANKCUSTOMER(
customer_name varchar(20) ,customer_street varchar(25) NOT NULL,customer_city
varchar(20) NOT NULL,primary key(customer_name)
);
create table DEPOSITOR(
customer_name varchar(20),
accno int,
primary key(customer_name,accno),
foreign key(customer_name) references bankcustomer(customer_name),
foreign key (accno) references BANKACCOUNT(accno)
);
create table LOAN(
loan_number int,branch_name varchar(20),amount real,primary key(loan_number),
foreign key (branch_name) references branch(branch_name)
);


insert into branch values("SBI_Chamrajpet","Banglore","50000"),
("SBI_ResidencyRoad","Banglore","10000"),
("SBI_ShivajiRoad","Bombay","20000"),
```

```
("SBI_ParlimentRoad","Delhi","10000"),
("SBI_Jantarmantar","Delhi","20000");
select * from branch;

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

insert into bankcustomer values("Avinash","Bull_Temple_Road","Banglore"),
("Dinesh","Baneergatta_Road","Banglore"),
("Mohan","NationalCollege_Road","Banglore"),
("Nikil","Akbar_Road","Delhi"),
("Ravi","Prithviraj_Road","Delhi");
select * from bankcustomer;

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

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

Select * from branch;
BRANCH

Result Grid | Filter Rows: | Edit:

branch_name	branch_city	assets
SBI_Chamrajpet	Banglore	50000
SBI_Jantarmantar	Delhi	20000
SBI_ParlimentRoad	Delhi	10000
SBI_ResidencyRoad	Banglore	10000
SBI_ShivajiRoad	Bombay	20000
NULL	NULL	NULL

Select * from bankaccount;
BANKACCOUNT

Result Grid | Filter Rows:

accno	branch_name	balance
1	SBI_Chamrajpet	2000
2	SBI_ResidencyRoad	5000
3	SBI_ShivajiRoad	6000
4	SBI_ParlimentRoad	9000
5	SBI_Jantarmantar	8000
6	SBI_ShivajiRoad	4000
8	SBI_ResidencyRoad	4000
9	SBI_ParlimentRoad	3000
10	SBI_ResidencyRoad	5000
11	SBI_Jantarmantar	2000
NULL	NULL	NULL

Select * from bankcustomer;
BANKCUSTOMER

Result Grid | Filter Rows: | Edit:

customer_name	customer_street	customer_city
Avinash	Bull_Temple_Road	Banglore
Dinesh	Baneergatta_Road	Banglore
Mohan	NationalCollege_Road	Banglore
Nikil	Akbar_Road	Delhi
Ravi	Prithviraj_Road	Delhi
NULL	NULL	NULL

bankcustomer 3

Select * from depositor;
DEPOSITOR

Result Grid | Filter Rows:

customer_name	accno
Avinash	1
Dinesh	2
Nikil	4
Ravi	5
Avinash	8
Nikil	9
Dinesh	10
Nikil	11
NULL	NULL

Select * from loan;
LOAN

loan_number	branch_name	amount
1	SBI_Chamrajpet	1000
2	SBI_ResidencyRoad	2000
3	SBI_ShivajiRoad	3000
4	SBI_ParlimentRoad	4000
5	SBI_Jantarmantar	5000
NULL	NULL	NULL

loan 1

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

select d.customer_name from depositor d ,bankaccount a where a.accno=d.accno and a.branch_name="SBI_ResidencyRoad" group by d.customer_name having count(*)>=2;

Result Grid | Filter R

customer_name
Dinesh

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

select d.customer_name from bankaccount a, depositor d,branch b where d.accno=a.accno and b.branch_name=a.branch_name and b.branch_city="Delhi" group by d.customer_name having count(distinct b.branch_name)=(select count(branch_name) from branch where branch_city="Delhi");

customer_name
Nikil

v. Demonstrate how you delete all account tuples at every branch located in a specific city (Ex. Bombay).

delete from bankaccount where branch_name in(select branch_name from branch where branch_city="Bombay");

Result Grid | Filter Rows:

accno	branch_name	balance
1	SBI_Chamrajpet	2000
2	SBI_ResidencyRoad	5000
4	SBI_ParlimentRoad	9000
5	SBI_Jantarmantar	8000
8	SBI_ResidencyRoad	4000
9	SBI_ParlimentRoad	3000
10	SBI_ResidencyRoad	5000
11	SBI_Jantarmantar	2000
NULL	NULL	NULL

bankaccount 8

PROGRAM 3: SUPPLIER DATABASE

Consider the following schema:

SUPPLIERS(<u>sid: integer</u>, sname: string, address: string)

PARTS(<u>pid: integer</u>, pname: string, color: string)

CATALOG(<u>sid: integer, pid</u>: integer, cost: real)

The Catalog relation lists the prices charged for parts by Suppliers.

```
create database  supplierdb1;
use supplierdb1;

create table suppliers(
sid int,
sname varchar(20),
address varchar(25),
primary key(sid)
);

create table parts(
pid int,
pname varchar(20),
color varchar(20),
primary key(pid)
);

create table catalog(
sid int,
pid int,
cost real,
primary key(sid,pid),
foreign key (sid)references suppliers (sid),
foreign key (pid)references parts (pid));
```

```
insert into suppliers values ("10001","Acme Widget","Banglore");
insert into suppliers values ("10002","Johns","Kolkata");
insert into suppliers values ("10003","Vimal","Mumbai");
insert into suppliers values ("10004","Reliance","Delhi");

insert into parts values ("20001","Book","Red");
insert into parts values ("20002","Pen","Red");
insert into parts values ("20003","Pencil","Green");
insert into parts values ("20004","Mobile","Green");
insert into parts values ("20005","Charger","Black");

insert into catalog values ("10001","20001","10");
insert into catalog values ("10001","20002","10");
insert into catalog values ("10001","20003","30");
insert into catalog values ("10001","20004","10");
insert into catalog values ("10001","20005","10");
insert into catalog values ("10002","20001","10");
insert into catalog values ("10002","20002","20");
insert into catalog values ("10003","20003","30");
insert into catalog values ("10004","20003","40");
insert into catalog values ("10003","20002","10");
```

Select * from suppliers;

SUPPLIERS

Result Grid | Filter Rows:

sid	sname	address
10001	Acme Widget	Banglore
10002	Johns	Kolkata
10003	Vimal	Mumbai
10004	Reliance	Delhi
NULL	NULL	NULL

suppliers 13

Output

Select * from parts;

PARTS

Result Grid | Filter Rows:

pid	pname	color
20001	Book	Red
20002	Pen	Red
20003	Pencil	Green
20004	Mobile	Green
20005	Charger	Black
NULL	NULL	NULL

parts 14

Output

Select * from catalog;

CATALOG

Result Grid | Filter Rows:

sid	pid	cost
10001	20001	10
10001	20002	10
10001	20003	30
10001	20004	10
10001	20005	10
10002	20001	10
10002	20002	20
10003	20003	30
10004	20003	40
NULL	NULL	NULL

catalog 15

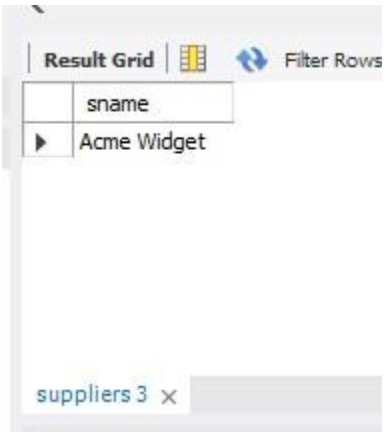
i) **Find the pnames of parts for which there is some supplier.**

select distinct P.pname from Parts P,Catalog C where P.pid=C.pid and exists(select 'X' from Catalog where pid=P.pid);

pname
Book
Pen
Pencil
Mobile
Charger

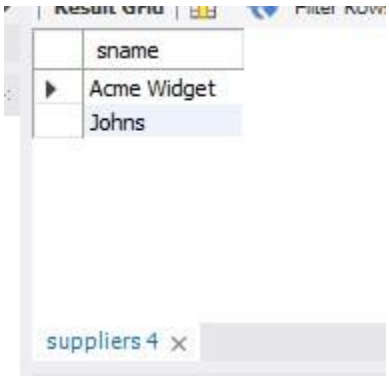
ii) Find the snames of suppliers who supply every part.

select S.sname, S.sid from suppliers S where S.sid IN (select C.sid from catalog C group by C.sid having count(distinct C.pid) = (select count(pid) from parts));



iii)Find the snames of suppliers who supply every red part.

select S.sname from suppliers S where S.sid IN (select C.sid from catalog c where 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))));



iv)Find the pnames of parts supplied by Acme Widget Suppliers and by no one else.

select P.pname from parts P,catalog C,suppliers 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=C1.pid and C1.sid=S1.sid and S1.sname<>"Acme Widget");

pname
Mobile
Charger

v) Find the sids of suppliers who charge more for some part than the average cost of that part (averaged over all the suppliers who supply that part)

select distinct C.sid from Catalog C where C.cost>(select avg(C1.cost) from catalog C1 where C1.pid=C.pid);

Result Grid Filter Rows:

sid
10002
10004

Catalog 6 ×

vi) For each part, find the sname of the supplier who charges the most for that part.

Select P.pname, P.pid ,S.sname from parts p,Suppliers S,catalog C where C.pid=P.pid and C.sid=S.sid and C.cost = (select max(C1.cost) from catalog C1 where C1.pid=P.pid);

Result Grid | Filter Rows:

pname	pid	sname
Book	20001	Acme Widget
Book	20001	Johns
Pen	20002	Johns
Pencil	20003	Reliance
Mobile	20004	Acme Widget
Charger	20005	Acme Widget

Result 8

PROGRAM 4: STUDENT FACULTY DATABASE

Consider the following database for
student enrollment for course :

STUDENT(<u>snum</u>: integer, sname:string, major: string, lvl: string, age: integer)

CLASS(<u>cname</u>: string, meetsat: time, room: string, fid: integer)

ENROLLED(<u>snum</u>: integer, <u>cname</u>:string)

FACULTY(<u>fid</u>: integer, fname:string, deptid: integer)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level(lvl) is a two character
code with 4 different values (example: Junior: JR etc)

Write the following queries in SQL.
No duplicates should be printed in any of the answers.

Select * from student;

snum	sname	major	lvl	age
1	jhon	CS	Sr	19
2	Smith	CS	Jr	20
3	Jacob	CV	Sr	20
4	Tom	CS	Jr	20
5	Rahul	CS	Jr	20
6	Rita	CS	Sr	21
NULL	NULL	NULL	NULL	NULL

student 13

select * from faculty;

fid	fname	deptid
11	Harish	1000
12	MV	1000
13	Mira	1001
14	Shiva	1002
15	Nupur	1000
NULL	NULL	NULL

faculty 14 ×

select * from class;

cname	meetsat	room	fid
Class1	2012-11-15 10:15:16	R1	14
Class10	2012-11-15 10:15:16	R128	14
Class2	2012-11-15 10:15:20	R2	12
Class3	2012-11-15 10:15:25	R3	11
Class4	2012-11-15 20:15:20	R4	14
Class5	2012-11-15 20:15:20	R3	15
Class6	2012-11-15 13:20:20	R2	14
Class7	2012-11-15 10:10:10	R3	14
NULL	NULL	NULL	NULL

select * from enrolled;

snum	cname
1	class1
2	class1
1	class10
3	class3
4	class3
5	class4
NULL	NULL

i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by the name "Shiva".

```
select s.sname from STUDENT s, CLASS c, ENROLLED e where s.snum=e.snum and c.cname =
e.cname and c.fid =(select fid from FACULTY where fname = "Shiva") and s.lvl ="Jr";
```

	sname
▶	Smith
	Rahul

ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.

```
SELECT C.cname FROM Class C
WHERE C.room = "R128"
or C.cname IN (SELECT E.cname FROM Enrolled E GROUP BY E.cname HAVING
count(E.snum) >= 5);
```

	cname
▶	Class10
*	NULL

iii. Find the names of all students who are enrolled in two classes that meet at the same Time.

```
select distinct s.sname from student s where s.snum in
(select e1.snum from enrolled e1, enrolled e2, class c1, class c2
where e1.snum = e2.snum and e1.cname != e2.cname and e1.cname = c1.cname
and e2.cname = c2.cname and c1.meetsat = c2.meetsat);
```

	sname
▶	jhon

iv. Find the names of faculty members who teach in every room in which some class is taught.

```
select f.fname,c.fid from faculty f,class c where f.fid=c.fid group by c.fid having
count(c.fid)=(select count(distinct room)from class);
```

Result Grid

fname	fid
Shiva	14

v. Find the names of faculty members for whom the combined enrollment of the courses that they teach less than five.

```
select distinct f.fname from faculty f where 5>(select COUNT(e.snum) from Class c, enrolled e
where c.cname = e.cname and c.fid = f.fid);
```

fname
Harish
MV
Mira
Shiva
Nupur

vi. Find the names of students who are not enrolled in any class.

```
select distinct s.sname from student s
where s.snum not in(select e.snum from enrolled e);
```

sname
Rita

vii. For each age value that appears in Students, find the level value that appears most

```
select s.age ,s.lvl from student s group by s.age having s.lvl in (select s1.lvl from student s1
where s1.age = s.age group by s1.age having count(*)>= all(select s2.lvl from student s2
where s2.age = s1.age group by s2.age));
```

age	lvl
19	Sr
20	Jr
21	Sr

PROGRAM 5: AIRLINE FLIGHT DATABASE

Consider the following database that keeps track of airline flight information:

FLIGHTS(flno: integer, from: string, to: string, distance: integer, departs: time, arrives: time, price: integer)

AIRCRAFT(aid: integer, aname: string, cruisingrange: integer)

CERTIFIED(eid: integer, aid: integer)

EMPLOYEES(eid: integer, ename: string, salary: integer)

Note that the Employees relation describes pilots and other kinds of employees as well; Every pilot is certified for some aircraft, and only pilots are certified to fly.

Write each of the following queries in SQL.

```
create database airlinedb;
use airlinedb;

create table FLIGHTS(
flno int,
fromplace varchar(20),
toplace  varchar(20),
distance integer,
departs datetime,
arrives datetime,
price int,
primary key(flno)
);
create table AIRCRAFT(
aid int,
aname varchar(20),
cruisingrange int,
primary key (aid)
);
create table EMPLOYEES(
eid int,
ename varchar(20),
salary int,
primary key (eid)
);
```

```
create table certified(
eid int,
aid int,
foreign key (eid) references employees(eid),
foreign key (aid) references aircraft(aid)
);


insert into aircraft values
("101","747","3000"),("102","Boeing","900"),("103","647","800"),("104","Dreamliner","10000"),("1
05","Boeing","3500"),("106","707","1500"),("107","Dream","120000");
select * from aircraft;

insert into flights values("101","Banglore","Delhi","2500","05/05/13 07.15.31","05/05/13
07.15.31","5000"),("102","Banglore","Lucknow","3000","05/05/13 07.15.31","05/05/13
11.15.31","6000"),("103","Lucknow","Delhi","500","05/05/13 12.15.31","05/05/13
17.15.31","3000"),
("107","Banglore","Frankfrut","8000","05/05/13 07.15.31","05/05/13 22.15.31","60000"),
("104","Banglore","Frankfrut","8500","05/05/13 07.15.31","05/05/13 23.15.31","75000"),
("105","Kolkata","Delhi","3400","05/05/13 07.15.31","05/05/13 09.15.31","7000");
insert into flights values("106","Delhi","kolkata","4000","05/05/13 09.15.31","06/05/13
06.00.00","2000");
select * from flights;

insert into employees values
("701","A","50000"),("702","B","100000"),("703","C","150000"),("704","D","90000"),("705","E","40
000"),("706","F","60000"),("707","G","90000");
select * from employees;

insert into certified
values("701","101"),("701","102"),("701","106"),("701","105"),("702","104"),("703","104"),("704","
104"),("702","107"),("703","107"),("704","107"),("702","101"),("703","105"),("704","105"),("705","1
03");
```

select * from aircraft;

Result Grid | Filter Rows:

aid	aname	cruisingrange
101	747	3000
102	Boeing	900
103	647	800
104	Dreamliner	10000
105	Boeing	3500
106	707	1500
107	Dream	120000
NULL	NULL	NULL

aircraft 6

select * from flights;

Result Grid | Filter Rows: | Edit: | Export/Import:

flno	fromplace	toplace	distance	departs	arrives	price
101	Banglore	Delhi	2500	2005-05-13 07:15:31	2005-05-13 07:15:31	5000
102	Banglore	Lucknow	3000	2005-05-13 07:15:31	2005-05-13 11:15:31	6000
103	Lucknow	Delhi	500	2005-05-13 12:15:31	2005-05-13 17:15:31	3000
104	Banglore	Frankfrut	8500	2005-05-13 07:15:31	2005-05-13 23:15:31	75000
105	Kolkata	Delhi	3400	2005-05-13 07:15:31	2005-05-13 09:15:31	7000
106	Delhi	kolkata	4000	2005-05-13 09:15:31	2006-05-13 06:00:00	2000
107	Banglore	Frankfrut	8000	2005-05-13 07:15:31	2005-05-13 22:15:31	60000
NULL	NULL	NULL	NULL	NULL	NULL	NULL

flights 7

select * from employees;

eid	ename	salary
701	A	50000
702	B	100000
703	C	150000
704	D	90000
705	E	40000
706	F	60000
707	G	90000
NULL	NULL	NULL

select * from certified;

Result Grid Filter Rows:

eid	aid
701	101
701	102
701	106
701	105
702	104
703	104
704	104
702	107
703	107
704	107
702	101
703	105
704	105
705	103

certified 9

Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80,000

select distinct A.aname from aircraft A where A.aid in (select C.aid from certified C,Employees E where C.eid=E.eid and not exists (select * from employees E1 where E1.eid=E.eid and E1.salary<80000));

Result Grid | Filter Rov

aname
747
Dreamliner
Boeing
Dream

For each pilot who is certified for more than three aircrafts, find the eid and the maximum cruisingrange of the aircraft for which she or he is certified.

select C.eid ,max(A.cruisingrange) from certified C,aircraft A where C.aid=A.aid group by C.eid having count(*)>3;

Result Grid | Filter Rows:

eid	max(A.cruisingrange)
701	3500

Find the names of pilots whose salary is less than the price of the cheapest route from Bengaluru to Frankfurt.

select distinct E.ename from employees E where E.salary <(select min(F.price) from flights F where F.fromplace="Banglore" and F.toplace="Frankfrut");

Result Grid | Filte

ename
A
E

For all aircraft with cruisingrange over 1000 Kms, find the name of the aircraft and the average salary of all pilots certified for this aircraft.

select A.aname,A.cruisingrange,avg(E.salary) from aircraft A,employees E,Certified C where C.eid=E.eid and C.aid=A.aid group by A.aname having A.cruisingrange>1000;

aname	cruisingrange	avg(E.salary)
747	3000	75000.0000
Dreamliner	10000	113333.3333
707	1500	50000.0000
Dream	120000	113333.3333

Find the names of pilots certified for some Boeing aircraft.

select distinct E.ename from employees E,certified C,aircraft A where E.eid = C.eid and C.aid =A .aid and A.aname LIKE "Boeing";

Result Grid | Filter R

ename
A
C
D

Result 1

Find the aids of all aircraft that can be used on routes from Bengaluru to New Delhi

select A.aid from aircraft A where A.cruisingrange > (select min(F.distance) from flights F where F.fromplace = "Banglore" and F.toplace = "Delhi");

Result Grid | Filter Rov

aid
101
104
105
107
NULL

aircraft 2

A customer wants to travel from Bangalore to Kolkata New with no more than two changes of flight. List the choice of departure times from Madison if the customer wants to arrive in Kolkata by 6 p.m.

select f.fromplace,f.toplace,f.arrives from flights f where(f.fromplace="Banglore" and f.toplace=(select fromplace from flights where toplace="kolkata"))or f.toplace="kolkata";

fromplace	toplace	arrives
Banglore	Delhi	2005-05-13 07:15:31
Delhi	kolkata	2006-05-13 06:00:00