

LAB RECORD- DBMS (FIRST 5 PROGRAMS)

Anitej Prasad

1BM19CS194

4-D

LAB 1 QUERIES:

```
create database insurance;
```

```
use insurance;
```

```
create table person(  
    driver_id varchar(10),  
    name varchar(20),  
    address varchar(30),  
    primary key(driver_id)  
);
```

```
desc person;
```

```
create table car(  
    reg_num varchar(10),
```

```
        model varchar(10),  
        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(10),  
    reg_num varchar(10),  
    primary key(driver_id,reg_num),  
    foreign key(driver_id) references person(driver_id),  
    foreign key(reg_num) references car(reg_num)  
);
```

```
desc owns;
```

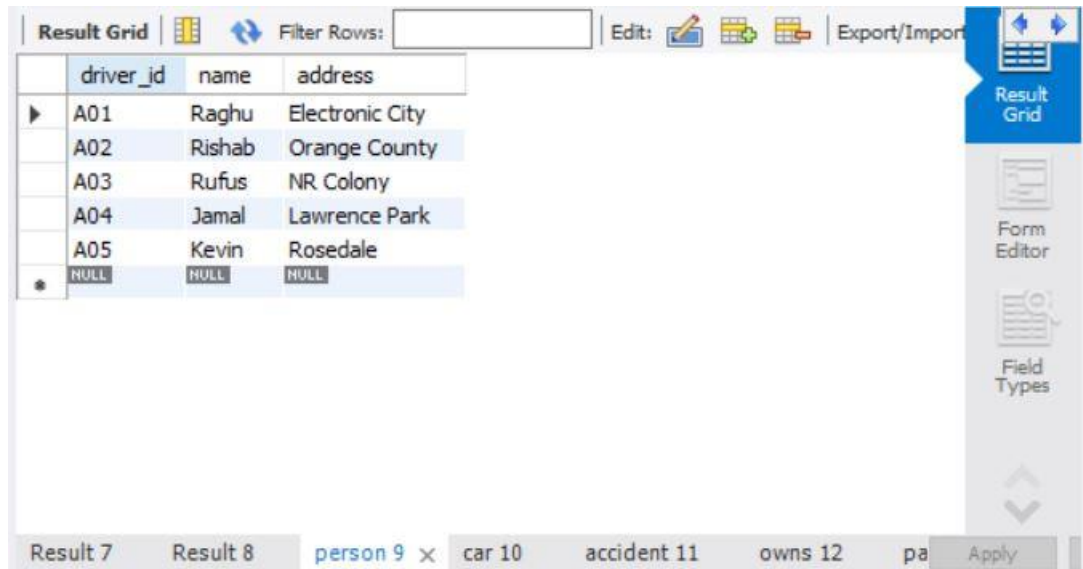
```
create table participated(  
    driver_id varchar(10),  
    reg_num varchar(10),  
    report_num int,  
    damage_amount int,  
    primary key(driver_id,reg_num,report_num),  
    foreign key(driver_id) references person(driver_id),  
    foreign key(reg_num) references car(reg_num),  
    foreign key(report_num) references accident(report_num)  
);
```

```
desc participated;
```

```
insert into person values('A01','Raghu','Electronic City');  
insert into person values('A02','Rishab','Orange County');  
insert into person values('A03','Rufus','NR Colony');  
insert into person values('A04','Jamal','Lawrence Park');  
insert into person values('A05','Kevin','Rosedale');
```

```
commit;
```

select * from person;



	driver_id	name	address
▶	A01	Raghu	Electronic City
	A02	Rishab	Orange County
	A03	Rufus	NR Colony
	A04	Jamal	Lawrence Park
	A05	Kevin	Rosedale
*	NULL	NULL	NULL

insert into car values('KA031111','Accord',2005);

insert into car values('KA041122','MX-5',2019);

insert into car values('KA051133','Indica',2010);

insert into car values('KA061144','Prius',2015);

insert into car values('KA071155','Camry',2020);

commit;

select * from car;

Result Grid			
reg_num	model	year	
KA031111	Accord	2005	
KA041122	MX-5	2019	
KA051133	Indica	2010	
KA061144	Prius	2015	
KA071155	Camry	2020	
NULL	NULL	NULL	

```

insert into accident values(111,'2020-01-01','NR Road');
insert into accident values(122,'2020-02-02','Dalhousie Road');
insert into accident values(133,'2020-03-03','Henry Road');
insert into accident values(144,'2020-04-04','Beehive Road');
insert into accident values(155,'2020-05-05','Orange Street');
commit;

```

```

select * from accident;

```

Result Grid			
report_num	accident_date	location	
11	2008-01-01	NR Road	
12	2008-02-02	Dalhousie Road	
13	2020-03-03	Henry Road	
14	2020-04-04	Beehive Road	
15	2020-05-05	Orange Street	
NULL	NULL	NULL	

```
insert into owns values ('A01','KA031111');
insert into owns values ('A02','KA041122');
insert into owns values ('A03','KA051133');
insert into owns values ('A04','KA061144');
insert into owns values ('A05','KA071155');
commit;
```

```
select * from owns;
```

	driver_id	reg_num
▶	A01	KA031111
	A02	KA041122
	A03	KA051133
	A04	KA061144
	A05	KA071155
★	NULL	NULL

```
insert into participated values ('A01','KA031111',111, 10000);
insert into participated values ('A02','KA041122',122, 20000);
insert into participated values ('A03','KA051133',133, 30000);
insert into participated values ('A04','KA061144',144, 40000);
insert into participated values ('A05','KA071155',155, 50000);
commit;
```

select * from participated;

	driver_id	reg_num	report_num	damage_amount
▶	A01	KA031111	11	10000
	A02	KA041122	12	20000
	A03	KA051133	13	30000
	A04	KA061144	14	40000
	A05	KA071155	15	50000
★	NULL	NULL	NULL	NULL

Additional Queries:

update participated

set damage_amount = 2500

where reg_num='KA031111';

select * from participated;

driver_id	reg_num	report_num	damage_amount
A01	KA031111	11	10000
A02	KA041122	12	25000
A03	KA051133	13	30000
A04	KA061144	14	40000
A05	KA071155	15	50000
NULL	NULL	NULL	NULL

```

insert into accident values(101,'2008-03-08',Domlur);

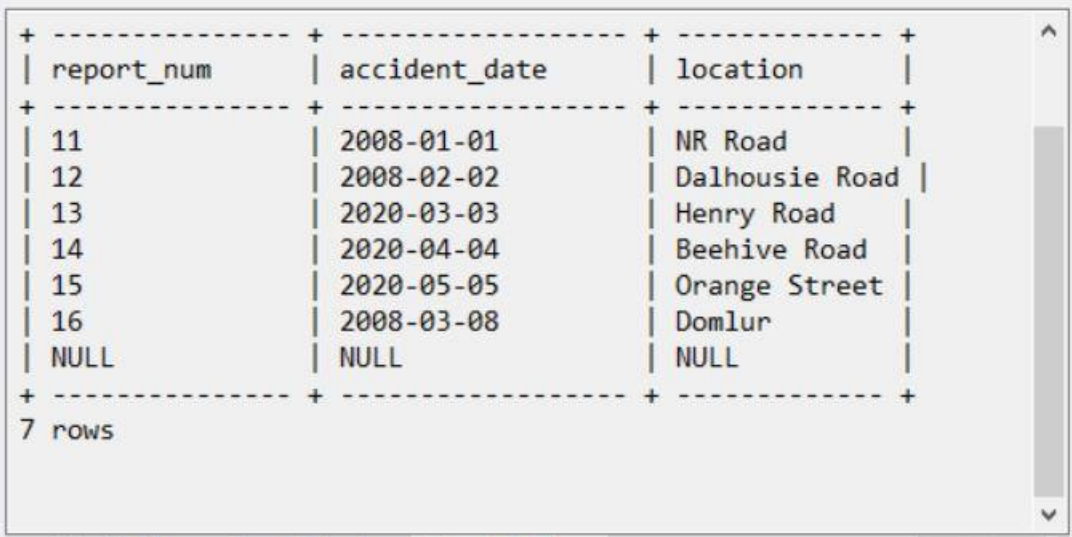
insert into participated values('A01','KA031111',101, 1001);

commit;

select * from accident;

select * from participated;

```



report_num	accident_date	location
11	2008-01-01	NR Road
12	2008-02-02	Dalhousie Road
13	2020-03-03	Henry Road
14	2020-04-04	Beehive Road
15	2020-05-05	Orange Street
16	2008-03-08	Domlur
NULL	NULL	NULL

7 rows

participated 15 Query Output Query Output x Apply

```

insert into car values('KA01010', 'Indica', 2002);

insert into owns values('A02', 'KA01010');

insert into accident values(200, '2008-12-01', 'Pinto Road');

insert into participated values('A02', 'KA01010', 200, 500);

commit;

```

```

select * from car;

select * from owns;

select * from accident;

select * from participated;

```


Result Grid	Filter Rows:	Export:	Wrap Cell Content:	Result Grid
count(distinct driver_id)				Form Editor
0				Field Types

Query Output Query Output Query Output Result 16 x Read Only

select count(*) from accident where year(accident_date)=2008;

select count(*) from participated where reg_num in (select reg_num from car where model="Indica");

Result Grid	Filter Rows:	Export:	Wrap Cell Content:	Result Grid
count(report_num)				Form Editor
1				Field Types

Query Output Query Output Query Output Result 17 x Read Only

LAB 2- QUERIES

```
create database bank;
```

```
use bank;
```

```
create table branch (
```

```
    branch_name varchar(25),
```

```
    branch_city varchar(15),
```

```
    assets int,
```

```
    primary key (branch_name)
```

```
);
```

```
create table bank_account (
```

```
    accno int,
```

```
    branch_name varchar(25),
```

```
    balance int,
```

```
    primary key (accno),
```

```
    foreign key (branch_name) references branch(branch_name)
```

```
);
```

```
create table bank_customer (
```

```
    customer_name varchar(10),
```

```
    customer_street varchar(25),
```

```
    customer_city varchar(15),
```

```
    primary key (customer_name)
```

);

```
create table depositer (  
    customer_name varchar(10),  
    accno int,  
    primary key(customer_name, accno),  
    foreign key (customer_name) references  
bank_customer(customer_name),  
    foreign key (accno) references bank_account(accno)  
);
```

```
create table loan (  
    loan_number int,  
    branch_name varchar(25),  
    amount int,  
    primary key (loan_number),  
    foreign key (branch_name) references branch(branch_name)  
);
```

```
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_ParliamentRoad', 'Delhi', 10000);  
insert into branch values('SBI_Jantarmanatar', 'Delhi', 20000);
```

commit;

	branch_name	branch_city	assets
▶	SBI_Chamrajpet	Bangalore	50000
	SBI_Jantarmanatar	Delhi	20000
	SBI_ParliamentRoad	Delhi	10000
	SBI_ResidencyRoad	Bangalore	10000
	SBI_ShivajiRoad	Bombay	20000
*	NULL	NULL	NULL

```
insert into bank_account values(1, 'SBI_Chamrajpet', 2000);
insert into bank_account values(2, 'SBI_ResidencyRoad', 5000);
insert into bank_account values(3, 'SBI_ShivajiRoad', 6000);
insert into bank_account values(4, 'SBI_ParliamentRoad', 9000);
insert into bank_account values(5, 'SBI_Jantarmanatar', 8000);
insert into bank_account values(6, 'SBI_ShivajiRoad', 4000);
insert into bank_account values(8, 'SBI_ResidencyRoad', 4000);
insert into bank_account values(9, 'SBI_ParliamentRoad', 3000);
insert into bank_account values(10, 'SBI_ResidencyRoad', 5000);
insert into bank_account values(11, 'SBI_Jantarmanatar', 2000);
```

Result Grid			
Filter Rows:			
	accno	branch_name	balance
▶	1	SBI_Chamrajpet	2000
	2	SBI_ResidencyRoad	5000
	3	SBI_ShivajiRoad	6000
	4	SBI_ParliamentRoad	9000
	5	SBI_Jantarantar	8000
	6	SBI_ShivajiRoad	4000
	8	SBI_ResidencyRoad	4000
	9	SBI_ParliamentRoad	3000
	10	SBI_ResidencyRoad	5000
	11	SBI_Jantarantar	2000
•	NULL	NULL	NULL

Result Grid
Form Editor
Field Types

Query Output Query Output Query Output branch 18 bank_account : Apply

commit;

insert into bank_customer values ('Abhay', 'Bull_Temple_Road', 'Bangalore');




insert into bank_customer values ('Dhruv', 'Bannerghatta_Road', 'Bangalore');

insert into bank_customer values ('Mohit', 'National_College_Road', 'Bangalore');

insert into bank_customer values ('Nick', 'Akbar_Road', 'Delhi');

insert into bank_customer values ('Rahul', 'Prithviraj_Road', 'Delhi');

commit;

Result Grid			
Filter Rows: <input type="text"/>			
Edit:    Export/Import			
	customer_name	customer_street	customer_city
▶	Abhay	Bull Temple Road	Bangalore
	Dhruv	Bannergatta Road	Bangalore
	Mohit	National College Road	Bangalore
	Nick	Akbar Road	Delhi
	Rahul	Prithviraj Road	Delhi
*	NULL	NULL	NULL





Result Grid
Form Editor
Field Types
Apply

ry Output Query Output branch 18 bank_account 19 bank_customer 2

```

insert into depositer values('Abhay', 1);
insert into depositer values('Dhruv', 2);
insert into depositer values('Nick', 4);
insert into depositer values('Rahul', 5);
insert into depositer values('Abhay', 8);
insert into depositer values('Nick', 9);
insert into depositer values('Dhruv', 10);
insert into depositer values('Nick', 11);
commit;

```

Result Grid		
Filter Rows: <input type="text"/>		
Edit:    Export/Import 		
	customer_name	accno
▶	Abhay	1
	Dhruv	2
	Nick	4
	Rahul	5
	Abhay	8
	Nick	9
	Dhruv	10
	Nick	11
*	NULL	NULL

Result Grid
Form Editor
Field Types

k_account 19 bank_customer 20 **depositer 21** × loan 22 bank_account 23 Apply

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_ParliamentRoad', 4000);

insert into loan values(5, 'SBI_Jantarantar', 5000);

commit;

Result Grid			
Filter Rows: <input type="text"/>			
Edit:    Export/Import 			
	loan_number	branch_name	amount
▶	1	SBI_Chamrajpet	1000
	2	SBI_ResidencyRoad	2000
	3	SBI_ShivajiRoad	3000
	4	SBI_ParliamentRoad	4000
	5	SBI_Jantarantar	5000
*	NULL	NULL	NULL

Result Grid
Form Editor
Field Types

k_account 19 bank_customer 20 depositer 21 **loan 22** × bank_account 23 Apply

```

select * from branch;

select * from bank_account;

select * from bank_customer;

select * from depositer;

select * from loan;

```

Additional Queries:

```

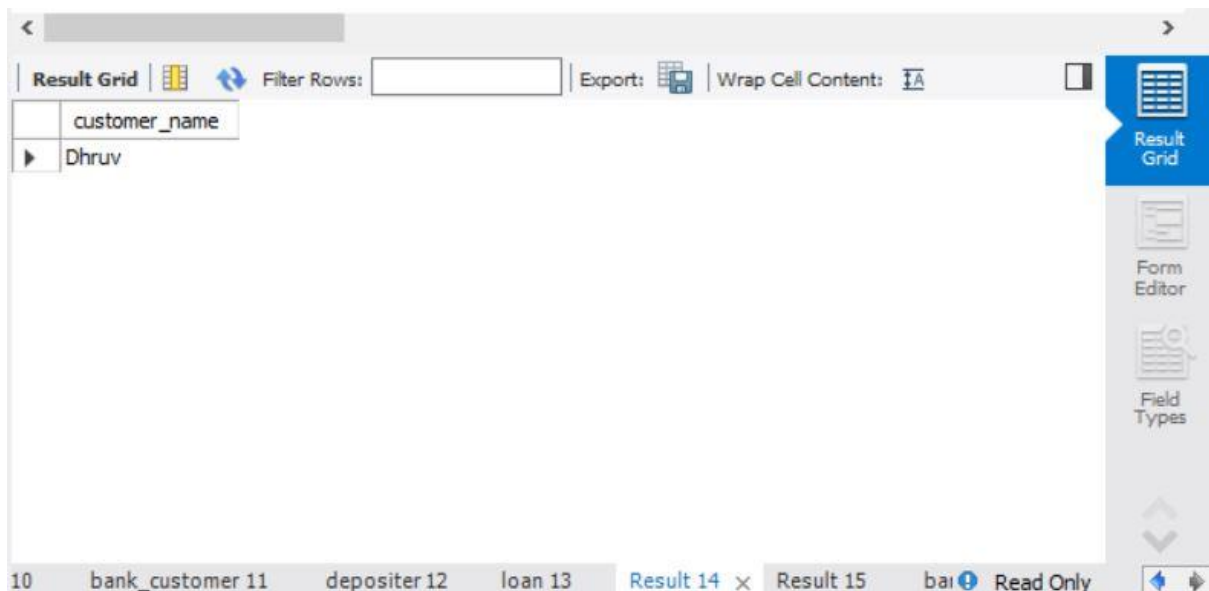
select distinct c.customer_name from bank_customer c, bank_account b
where exists(select d.customer_name, count(d.customer_name) from
depositer d, bank_account ba where ba.accno = d.accno and

```

```

c.customer_name = d.customer_name and ba.branch_name =
'SBI_ResidencyRoad' group by d.customer_name having
count(d.customer_name) >= 2);

```



```

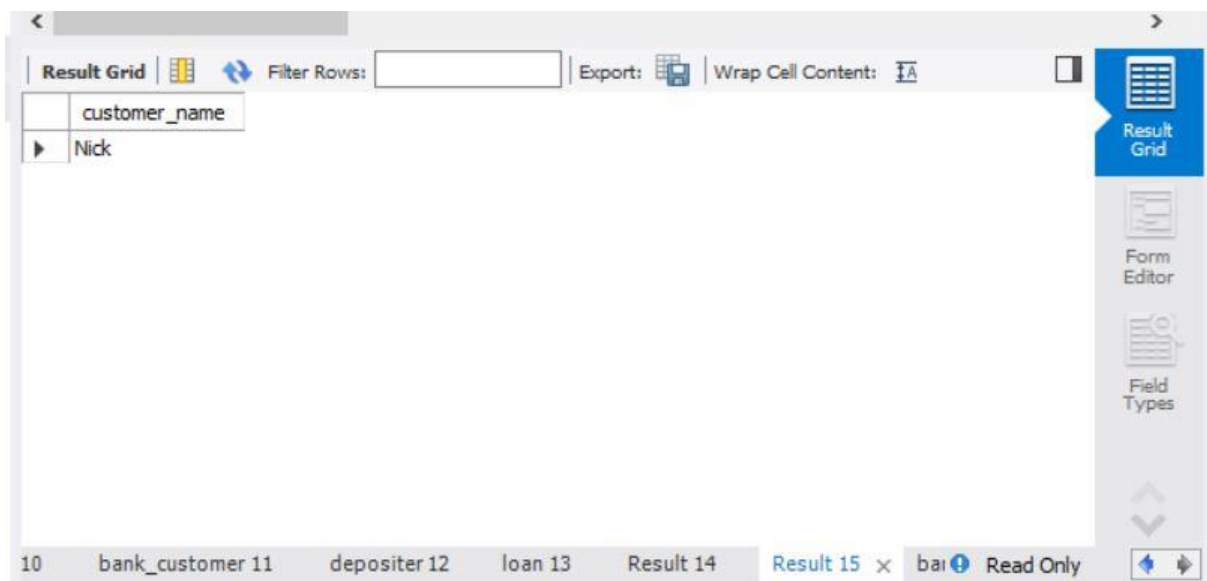
select d.customer_name from depositer d, branch b, bank_account a
where b.branch_name=a.branch_name

AND a.accno=d.accno

```



```
and branch_city='Delhi'  
group by d.customer_name  
HAVING COUNT(distinct b.branch_name)=(  
    SELECT COUNT(branch_name)  
    FROM branch  
    WHERE branch_city='Delhi');
```



```
delete from bank_account where branch_name in (select branch_name  
from branch where branch_city = 'Bombay');  
select * from bank_account;
```

Result Grid				Filter Rows:				Edit:				Export/Import:				Wrap			
	accno	branch_name	balance																
▶	1	SBI_Chamrajpet	2000																
	2	SBI_ResidencyRoad	5000																
	4	SBI_ParliamentRoad	9000																
	5	SBI_Jantarmanatar	8000																
	8	SBI_ResidencyRoad	4000																
	9	SBI_ParliamentRoad	3000																
	10	SBI_ResidencyRoad	5000																
	11	SBI_Jantarmanatar	2000																
*	NULL	NULL	NULL																

Result Grid

Form Editor

Field Types

10

bank_customer 11

depositer 12

loan 13

Result 14

Result 15

bai

Apply

Revert

LAB 3 QUERIES:

```
create database supplier;
```

```
use supplier;
```

```
create table SUPPLIERS(sid integer,sname varchar(20),address  
varchar(40),primary key(sid));
```

```
INSERT INTO `supplier`.`suppliers` (`sid`, `sname`, `address`) VALUES ('10001',  
'Acme Widget', 'Bangalore');
```

```
INSERT INTO `supplier`.`suppliers` (`sid`, `sname`, `address`) VALUES ('10002',  
'Johns', 'Kolkata');
```

```
INSERT INTO `supplier`.`suppliers` (`sid`, `sname`, `address`) VALUES ('10003',  
'Vimal', 'Mumbai');
```

```
INSERT INTO `supplier`.`suppliers` (`sid`, `sname`, `address`) VALUES ('10004',  
'Reliance', 'Delhi');
```

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

```
commit;
```

```
select* from SUPPLIERS;
```

```
create table PARTS(pid integer,pname varchar(20),color varchar(30),primary  
key(pid));
```

```
INSERT INTO `supplier`.`parts` (`pid`, `pname`, `color`) VALUES ('20001', 'Book',  
'Red');
```

```
INSERT INTO `supplier`.`parts` (`pid`, `pname`, `color`) VALUES ('20002', 'Pen',  
'Red');
```

```
INSERT INTO `supplier`.`parts` (`pid`, `pname`, `color`) VALUES ('20003', 'Pencil', 'Green');
```

```
INSERT INTO `supplier`.`parts` (`pid`, `pname`, `color`) VALUES ('20004', 'Mobile', 'Green');
```

```
INSERT INTO `supplier`.`parts` (`pid`, `pname`, `color`) VALUES ('20005', 'Charger', 'Black');
```

```
commit;
```

```
select* from PART;
```

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

```
create table CATALOG(sid integer,pid integer,foreign key(sid) references SUPPLIERS(sid),foreign key(pid) references PARTS(pid),
```

```
cost integer,primary key(sid,pid));
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10001', '20001', '10');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10001', '20002', '10');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10001', '20003', '30');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10001', '20004', '10');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10001', '20005', '10');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10002', '20001', '10');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10002', '20002', '20');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10003', '20003', '30');
```

```
INSERT INTO `supplier`.`catalog` (`sid`, `pid`, `cost`) VALUES ('10004', '20003', '40');
```

```
commit;
```

```
select* from CATALOG;
```

Additional Queries:

```
SELECT DISTINCT P.pname
```

```
FROM Parts P, Catalog C
```



```
WHERE P.pid = C.pid;
```

	pname
▶	Book
	Pen
	Pencil
	Mobile
	Charger



```
select S.sname from SUPPLIERS S where not exists
```

```
(select P.pid from PARTS P where not exists
```

```
(select C.sid from CATALOG C where C.sid = S.sid and C.pid = P.pid));
```

Result Grid				Filter Rows: [
	sname			
▶	Acme Widget			

```
select S.sname from SUPPLIERS S where not exists
(select P.pid from PARTS P where P.color = 'Red' and
(not exists (select C.sid from CATALOG C where C.sid = S.sid and C.pid =
P.pid)));
```

Result Grid				Filter Rows: [
	sname			
▶	Acme Widget			
	Johns			

```
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');
```

Result Grid		Filter Rows:
	pname	
▶	Mobile	
	Charger	

```

SELECT DISTINCT C.sid FROM Catalog C
WHERE C.cost > ( SELECT AVG (C1.cost)
FROM Catalog C1
WHERE C1.pid = C.pid );

```

	sid
▶	10002
	10004

```

SELECT 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:	
	pid	sname		
▶	20001	Acme Widget		
	20004	Acme Widget		
	20005	Acme Widget		
	20001	Johns		
	20002	Johns		
	20003	Reliance		

LAB 4 QUERIES:

```
CREATE DATABASE student_faculty;
```

```
USE student_faculty;
```

```
CREATE TABLE student(  
    snum INT,  
    sname VARCHAR(10),  
    major VARCHAR(2),  
    lvl VARCHAR(2),  
    age INT, primary key(snum));
```

```
CREATE TABLE faculty(  
    fid INT, fname VARCHAR(20),  
    deptid INT,  
    PRIMARY KEY(fid));
```

```
CREATE TABLE class(  
    cname VARCHAR(20),  
    metts_at TIMESTAMP,  
    room VARCHAR(10),  
    fid INT,  
    PRIMARY KEY(cname),  
    FOREIGN KEY(fid) REFERENCES faculty(fid));
```

```
CREATE TABLE enrolled(  
    snum INT,
```

```

cname VARCHAR(20),
PRIMARY KEY(snum,cname),
FOREIGN KEY(snum) REFERENCES student(snum),
FOREIGN KEY(cname) REFERENCES class(cname));

```

```

INSERT INTO STUDENT VALUES(1, 'jhon', 'CS', 'Sr', 19);
INSERT INTO STUDENT VALUES(2, 'Smith', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(3, 'Jacob', 'CV', 'Sr', 20);
INSERT INTO STUDENT VALUES(4, 'Tom ', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(5, 'Rahul', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(6, 'Rita', 'CS', 'Sr', 21);

```

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

```

INSERT INTO FACULTY VALUES(11, 'Harish', 1000);
INSERT INTO FACULTY VALUES(12, 'MV', 1000);
INSERT INTO FACULTY VALUES(13, 'Mira', 1001);
INSERT INTO FACULTY VALUES(14, 'Shiva', 1002);
INSERT INTO FACULTY VALUES(15, 'Nupur', 1000);

```

Result Grid			
Filter Rows:			
	fid	fname	deptid
▶	11	Harish	1000
	12	MV	1000
	13	Mira	1001
	14	Shiva	1002
	15	Nupur	1000
*	NULL	NULL	NULL

STUDENT 29 FACULTY 30 CLASS 31 ENROLLED 32 Result 33 class 34 Studen Apply

insert into class values('class1', '12/11/15 10:15:16', 'R1', 14);

insert into class values('class10', '12/11/15 10:15:16', 'R128', 14);

insert into class values('class2', '12/11/15 10:15:20', 'R2', 12);

insert into class values('class3', '12/11/15 10:15:25', 'R3', 11);

insert into class values('class4', '12/11/15 20:15:20', 'R4', 14);

insert into class values('class5', '12/11/15 20:15:20', 'R3', 15);

insert into class values('class6', '12/11/15 13:20:20', 'R2', 14);

insert into class values('class7', '12/11/15 10:10:10', 'R3', 14);

Result Grid				
Filter Rows:				
	cname	metts_at	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

STUDENT 29 FACULTY 30 CLASS 31 ENROLLED 32 Result 33 class 34 Studen Apply

insert into enrolled values(1, 'class1');

insert into enrolled values(2, 'class1');

insert into enrolled values(3, 'class3');

insert into enrolled values(4, 'class3');

insert into enrolled values(5, 'class4');

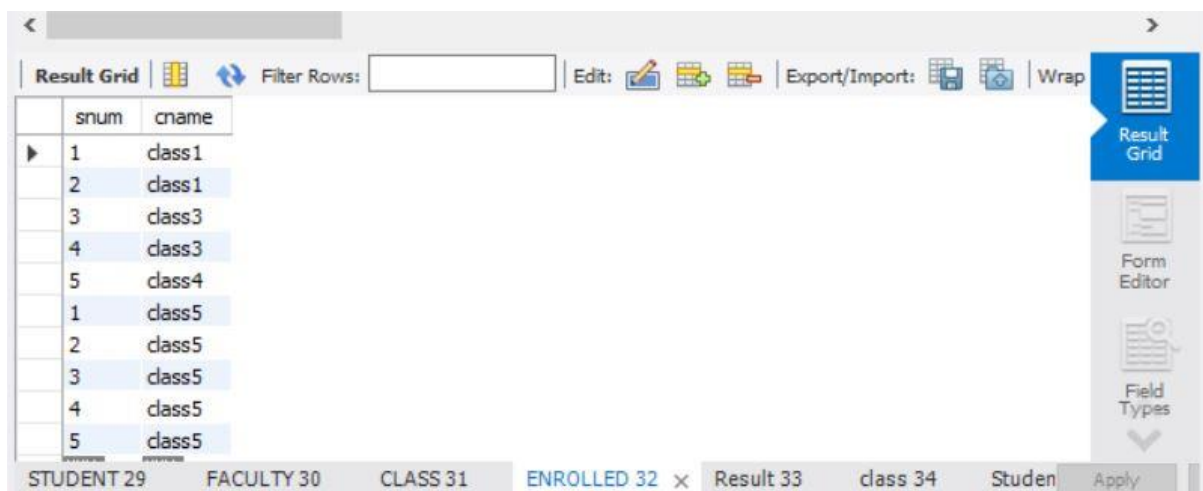
insert into enrolled values(1, 'class5');

insert into enrolled values(2, 'class5');

insert into enrolled values(3, 'class5');

insert into enrolled values(4, 'class5');

insert into enrolled values(5, 'class5');



	snum	cname
▶	1	class1
	2	class1
	3	class3
	4	class3
	5	class4
	1	class5
	2	class5
	3	class5
	4	class5
	5	class5

Additional Queries:

SELECT DISTINCT S.Sname

FROM Student S, Class C, Enrolled E, Faculty F

WHERE S.snum = E.snum AND E.cname = C.cname AND C.fid = F.fid AND

F.fname = 'Harish' AND S.lvl = 'Jr';

Sname
Tom

SELECT DISTINCT cname

FROM class

WHERE room='R128'

OR

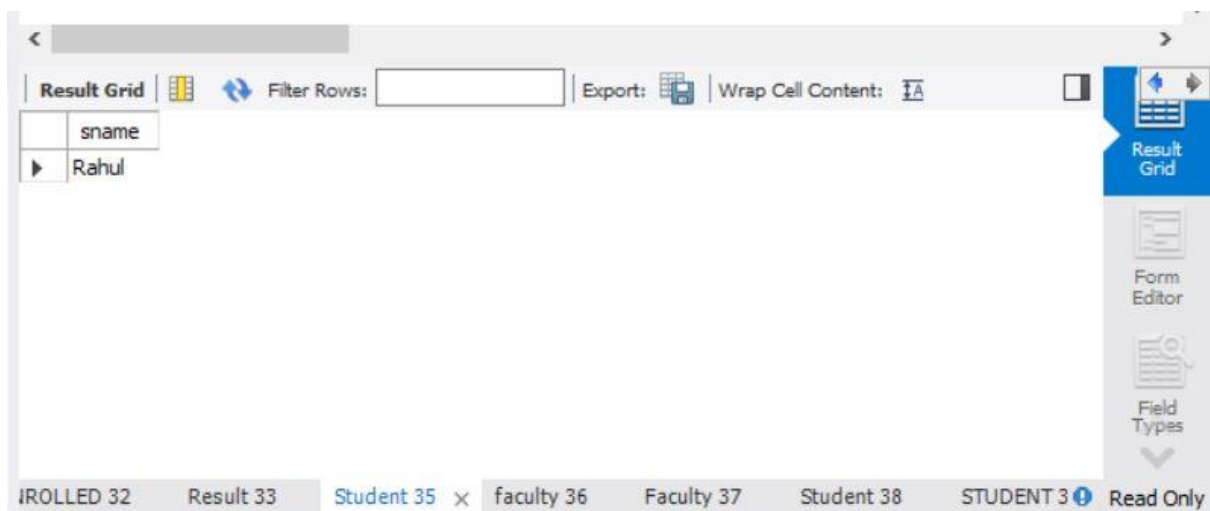
cname IN (SELECT e.cname FROM enrolled e GROUP BY e.cname
HAVING COUNT(*)>=5);

cname
class10
class5
NULL

```

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.metts_at =
C2.metts_at);

```



```

SELECT f.fname,f.fid
      FROM faculty f
      WHERE f.fid in ( SELECT fid FROM class
                      GROUP BY fid HAVING COUNT(*)=(SELECT COUNT(DISTINCT
room) FROM class) );

```

Result Grid		Filter Rows:	Edit:	Export/Import:	Wrap
	fname	fid			
▶	Shiva	14			
*	NULL	NULL			

Result Grid

Form Editor

Field Types

ROLLED 32 Result 33 Student 35 faculty 36 x Faculty 37 Student 38 STUDENT 3 Apply

```

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);

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	fname			
▶	Harish			
	MV			
	Mira			
	Shiva			

Result Grid

Form Editor

Field Types

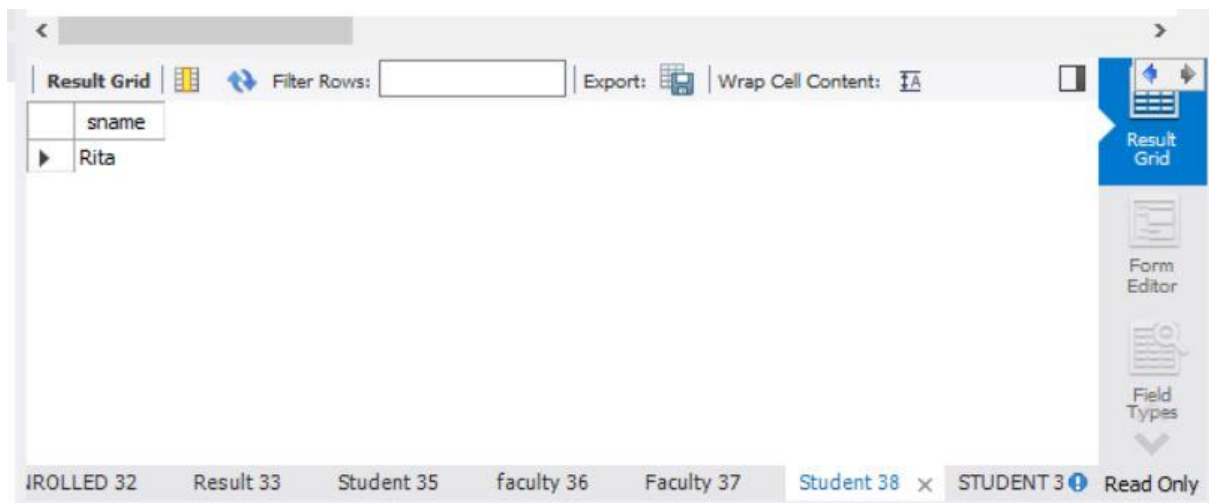
ROLLED 32 Result 33 Student 35 faculty 36 Faculty 37 x Student 38 STUDENT 3 Read Only

```

SELECT DISTINCT S.sname
FROM Student S

```

WHERE S.snum NOT IN (SELECT E.snum
FROM Enrolled E);



```
SELECT S.age, S.lvl
FROM STUDENT S
GROUP BY S.age, S.lvl
HAVING S.lvl IN(SELECT S1.lvl
                FROM STUDENT S1
                WHERE S1.age=S.age
                GROUP BY S1.age, S1.lvl
                HAVING COUNT(*) >= ALL (SELECT COUNT(*)
                                        FROM STUDENT S2
                                        WHERE S1.age=S2.age
                                        GROUP BY S2.lvl, S2.age))
ORDER BY S.age;
```


Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	age	lvl
▶	19	Sr
	20	Jr
	21	Sr

Result Grid

Form Editor

Field Types

ROLLED 32Result 33Student 35faculty 36Faculty 37Student 38STUDENT 3Read Only

LAB 5 QUERIES:

```
create database flightdb;
```

```
use flightdb;
```

```
create table flights(  
    flno int,  
    fromplace varchar(15),  
    toplace varchar(15),  
    distance int,  
    departs datetime,  
    arrives datetime,  
    price int,  
    primary key (flno)  
);  
desc flights;
```

```
create table aircraft(  
    aid int,  
    aname varchar(15),  
    cruisingrange int,  
    primary key (aid)  
);
```

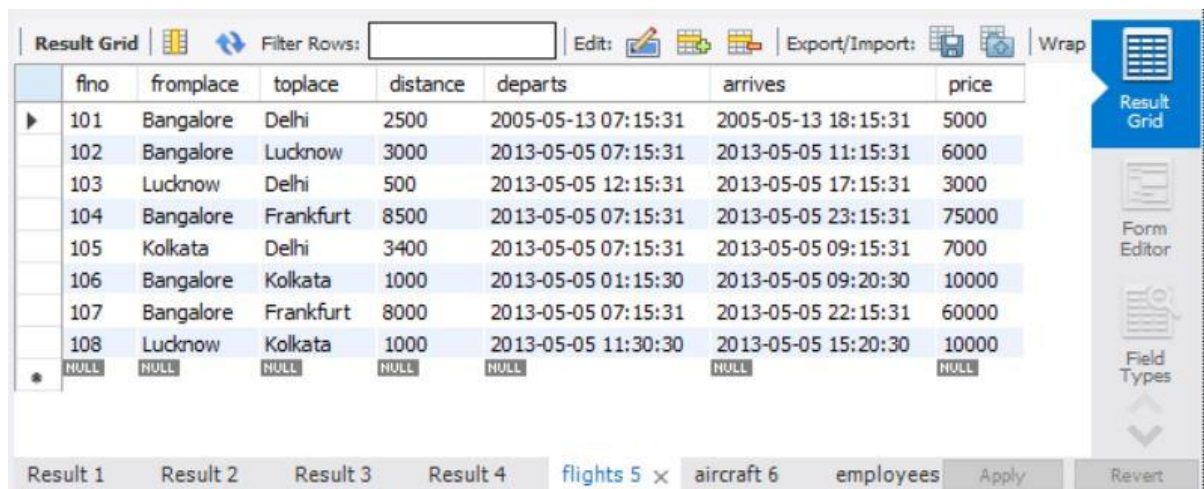
```
desc aircraft;
```

```
create table employees (  
    eid int,  
    ename varchar(15),
```

```
    salary int,
    primary key (eid)
);
desc employees;
create table certified (
    eid int,
    aid int,
    foreign key (eid) references employees(eid),
    foreign key (aid) references aircraft(aid)
);
desc certified;
insert into flights values(101, 'Bangalore', 'Delhi', 2500, '2005-05-13 07:15:31',
'2005-05-13 18:15:31', 5000);
insert into flights values(102, 'Bangalore', 'Lucknow', 3000, '2013-05-05
07:15:31', '2013-05-05 11:15:31', 6000);
insert into flights values(103, 'Lucknow', 'Delhi', 500, '2013-05-05 12:15:31',
'2013-05-05 17:15:31', 3000);
insert into flights values(107, 'Bangalore', 'Frankfurt', 8000, '2013-05-05
07:15:31', '2013-05-05 22:15:31', 60000);
insert into flights values(104, 'Bangalore', 'Frankfurt', 8500, '2013-05-05
07:15:31', '2013-05-05 23:15:31', 75000);
insert into flights values(105, 'Kolkata', 'Delhi', 3400, '2013-05-05 07:15:31',
'2013-05-05 09:15:31', 7000);
insert into flights values(106, 'Bangalore', 'Kolkata', 1000, '2013-05-05
01:15:30', '2013-05-05 09:20:30', 10000);
insert into flights values(108, 'Lucknow', 'Kolkata', 1000, '2013-05-05 11:30:30',
'2013-05-05 15:20:30', 10000);

commit;
```

select * from flights;



	fno	fromplace	toplace	distance	departs	arrives	price
▶	101	Bangalore	Delhi	2500	2005-05-13 07:15:31	2005-05-13 18:15:31	5000
	102	Bangalore	Lucknow	3000	2013-05-05 07:15:31	2013-05-05 11:15:31	6000
	103	Lucknow	Delhi	500	2013-05-05 12:15:31	2013-05-05 17:15:31	3000
	104	Bangalore	Frankfurt	8500	2013-05-05 07:15:31	2013-05-05 23:15:31	75000
	105	Kolkata	Delhi	3400	2013-05-05 07:15:31	2013-05-05 09:15:31	7000
	106	Bangalore	Kolkata	1000	2013-05-05 01:15:30	2013-05-05 09:20:30	10000
	107	Bangalore	Frankfurt	8000	2013-05-05 07:15:31	2013-05-05 22:15:31	60000
	108	Lucknow	Kolkata	1000	2013-05-05 11:30:30	2013-05-05 15:20:30	10000
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

insert into aircraft values(101, '747', 3000);

insert into aircraft values(102, 'Boeing', 900);

insert into aircraft values(103, '647', 800);

insert into aircraft values(104, 'Dreamliner', 10000);

insert into aircraft values(105, 'Boeing', 3500);

insert into aircraft values(106, '707', 1500);

insert into aircraft values(107, 'Dream', 120000);

insert into aircraft values(108, '707', 760);

insert into aircraft values(109, '747', 1000);

commit;

select * from aircraft;

aid	aname	cruisingrange
101	747	3000
102	Boeing	900
103	647	800
104	Dreamliner	10000
105	Boeing	3500
106	707	1500
107	Dream	120000
108	707	760
109	747	1000
NULL	NULL	NULL

```

insert into employees values(701, 'A', 50000);
insert into employees values(702, 'B', 100000);
insert into employees values(703, 'C', 150000);
insert into employees values(704, 'D', 90000);
insert into employees values(705, 'E', 40000);
insert into employees values(706, 'F', 60000);
insert into employees values(707, 'G', 90000);
commit;

```

```

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

```
insert into certified values(701, 101);  
insert into certified values(701, 102);  
insert into certified values(701, 106);  
insert into certified values(701, 105);
```

```
insert into certified values(702, 104);  
insert into certified values(703, 104);  
insert into certified values(704, 104);
```

```
insert into certified values(702, 107);  
insert into certified values(703, 107);  
insert into certified values(704, 107);
```

```
insert into certified values(702, 101);  
insert into certified values(702, 108);  
insert into certified values(701, 109);  
commit;  
select * from certified;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	eid	aid
▶	701	101
	701	102
	701	106
	701	105
	702	104
	703	104
	704	104
	702	107
	703	107
	704	107
	702	101
	702	108
	701	109

Result Grid

Form Editor

Field Types

Query Stats

Result 1

Result 2

Result 3

Result 4

flights 5

aircraft 6

employees 7

certified 8

Read Only

Additional Queries:

```

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
    )
);

```

The screenshot shows a database query tool interface. At the top, there's a toolbar with 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. The main area displays a table with two columns: 'aname' and an unlabeled column. The data rows are:

aname	
747	
Dreamliner	
Dream	
707	

At the bottom, a tab bar shows 'Result 3', 'Result 4', 'flights 5', 'aircraft 6', 'employees 7', 'certified 8', 'aircraft 9', and 'Read Only'. The 'aircraft 9' tab is active.

n

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

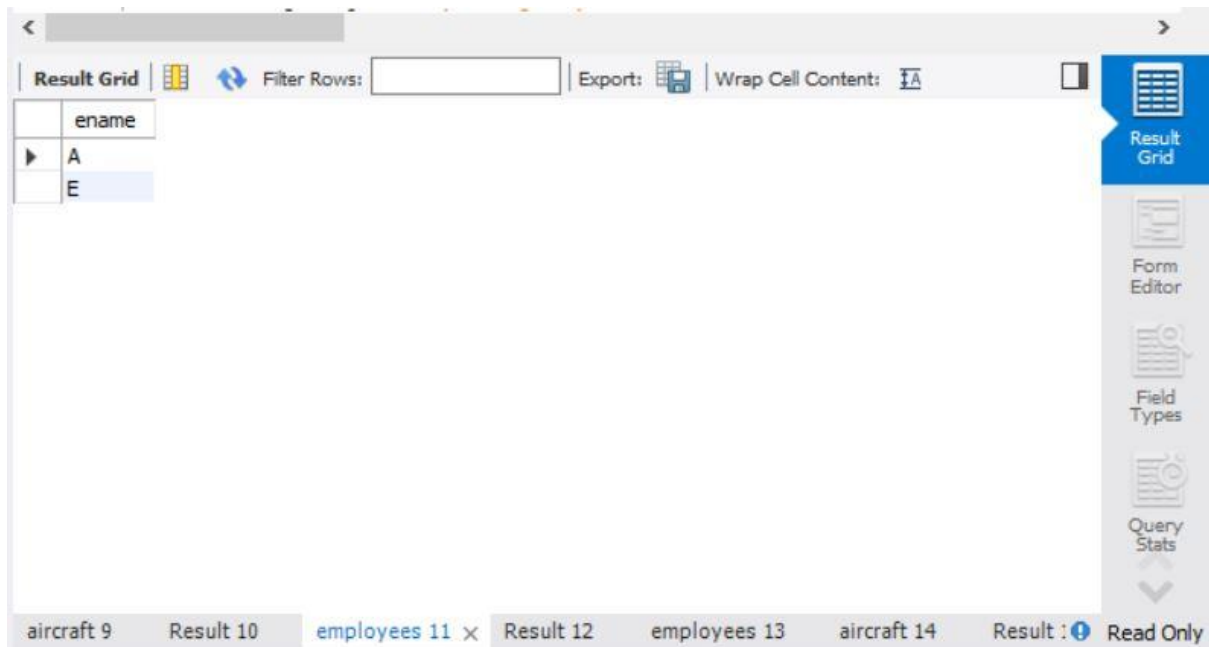
The screenshot shows the same database query tool interface. The main area displays a table with two columns: 'max(a.cruisingrange)' and 'eid'. The data rows are:

max(a.cruisingrange)	eid
3500	701
120000	702

At the bottom, a tab bar shows 'aircraft 9', 'Result 10', 'employees 11', 'Result 12', 'employees 13', 'aircraft 14', and 'Result : Read Only'. The 'Result 10' tab is active.

select ename from employees where salary <(

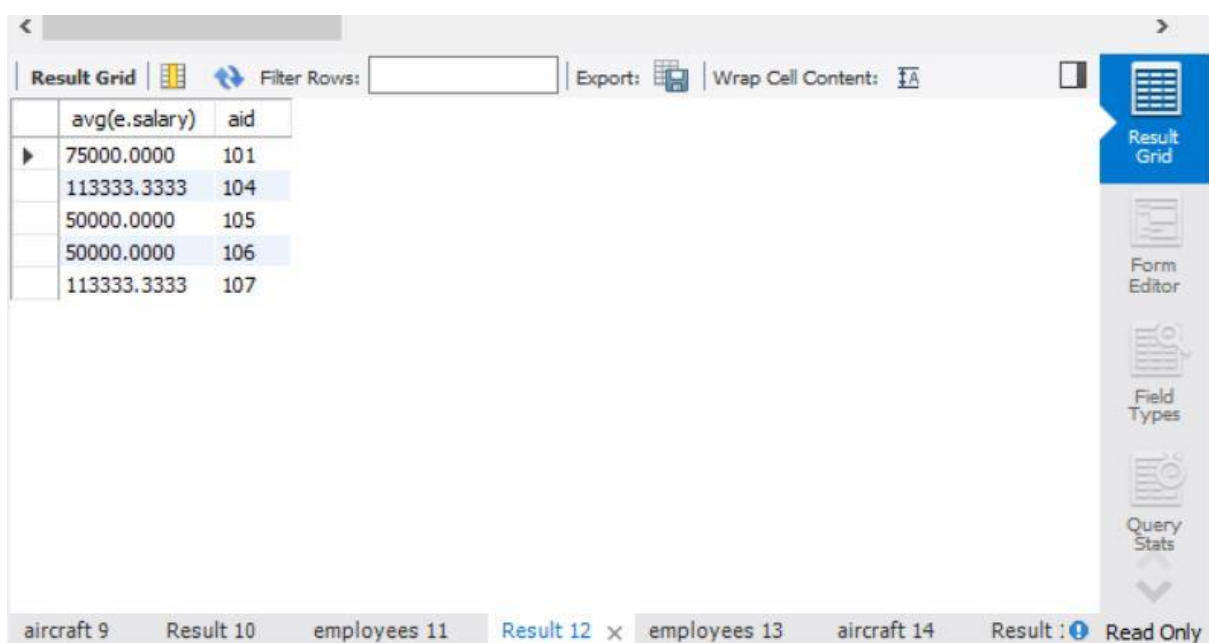
select min(price) from flights where fromplace='Bangalore' and
toplace='Frankfurt');



The screenshot shows a database query tool interface. At the top, there is a toolbar with icons for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. Below the toolbar, the 'Result Grid' is displayed with two rows of data. The first row has a column header 'ename' and a value 'A'. The second row has a value 'E'. On the right side, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Stats'. At the bottom, there is a tab bar with tabs labeled 'aircraft 9', 'Result 10', 'employees 11', 'Result 12', 'employees 13', 'aircraft 14', and 'Result 1'. The 'employees 11' tab is currently selected.

ename
A
E

select avg(e.salary), c.aid from certified c, employees e where c.aid in(
select aid from aircraft where cruisingrange>1000) and e.eid = c.eid group by
c.aid;



The screenshot shows a database query tool interface. At the top, there is a toolbar with icons for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. Below the toolbar, the 'Result Grid' is displayed with six rows of data. The first row has column headers 'avg(e.salary)' and 'aid'. The subsequent rows have values for these columns. On the right side, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Stats'. At the bottom, there is a tab bar with tabs labeled 'aircraft 9', 'Result 10', 'employees 11', 'Result 12', 'employees 13', 'aircraft 14', and 'Result 1'. The 'Result 12' tab is currently selected.

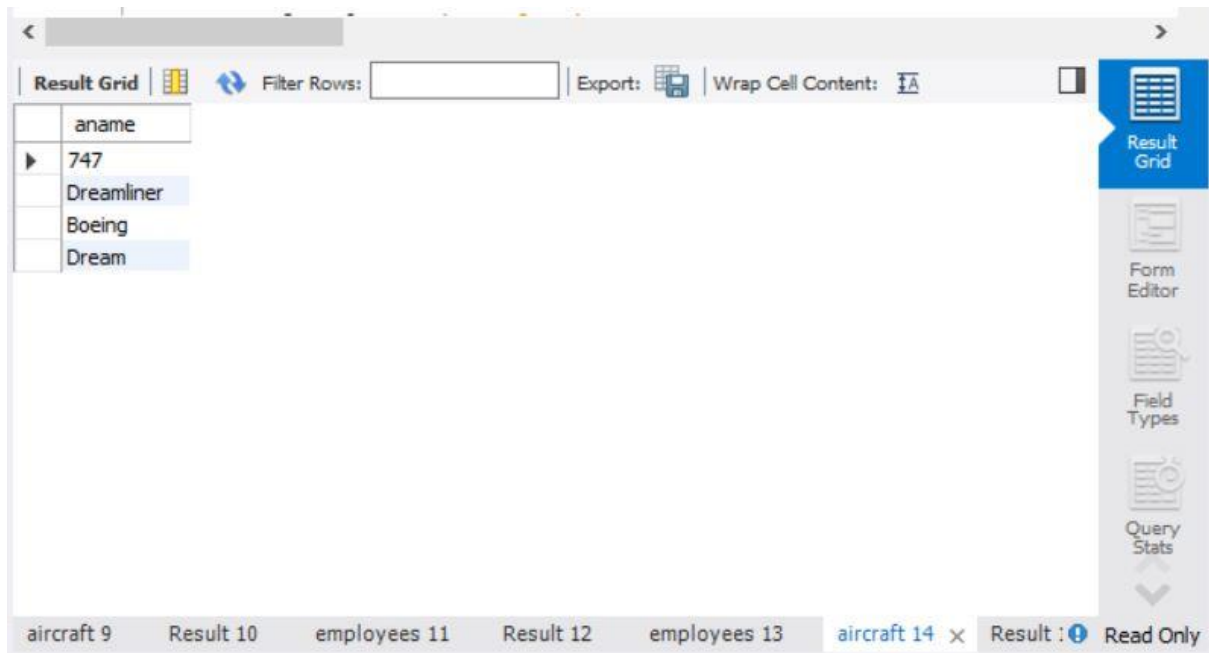
avg(e.salary)	aid
75000.0000	101
113333.3333	104
50000.0000	105
50000.0000	106
113333.3333	107

```
select ename from employees where eid in(  
select eid from certified where aid in(  
select aid from aircraft where aname = 'Boeing'));
```

The screenshot shows a database query tool interface. At the top, there is a toolbar with buttons for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. Below the toolbar, a table with one column 'ename' and one row containing the value 'A' is displayed. On the right side, there is a vertical toolbar with buttons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Stats'. At the bottom, there is a tab bar with tabs labeled 'aircraft 9', 'Result 10', 'employees 11', 'Result 12', 'employees 13 x', 'aircraft 14', 'Result 1', and 'Read Only'.

ename
A

```
select aname from aircraft where cruisingrange > any (select distance from  
flights where fromplace='Bangalore' and toplace='Delhi');
```

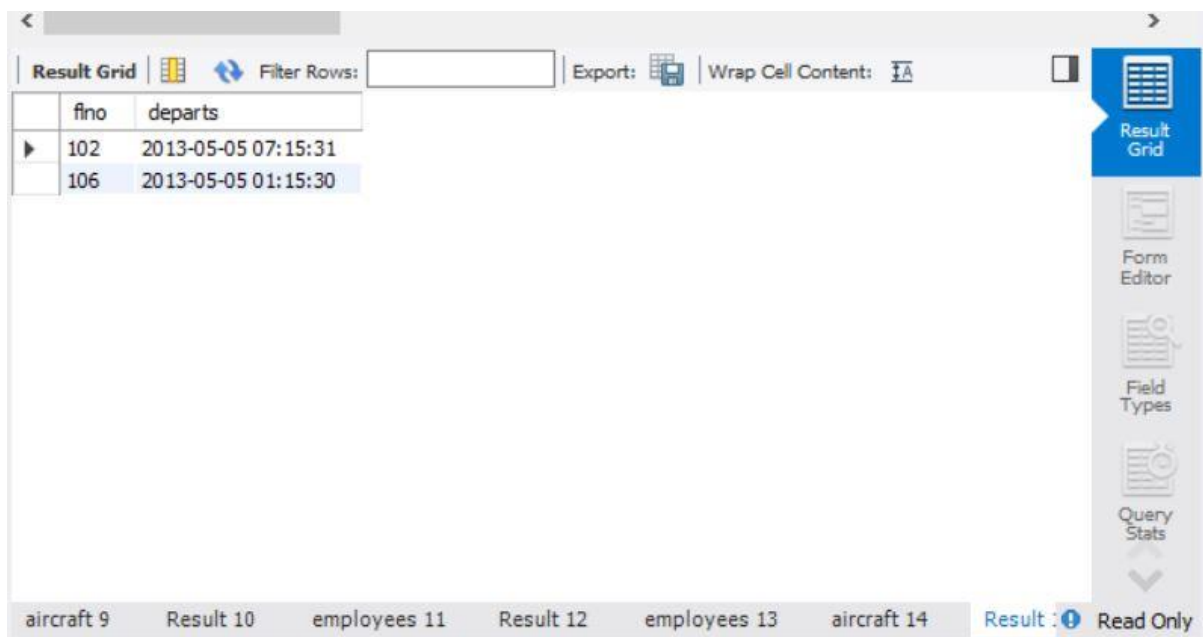


```

SELECT F.flno, F.departs
FROM flights F
WHERE F.flno IN ( ( SELECT F0.flno
FROM flights F0
WHERE F0.fromplace = 'Bangalore' AND F0.toplace = 'Kolkata'
AND extract(hour from F0.arrives) < 18 )
UNION
( SELECT F0.flno
FROM flights F0, flights F1
WHERE F0.fromplace = 'Bangalore' AND F0.toplace <> 'Kolkata'
AND F0.toplace = F1.fromplace AND F1.toplace = 'Kolkata'
AND F1.departs > F0.arrives
AND extract(hour from F1.arrives) < 18)
UNION
( SELECT F0.flno

```

```
FROM flights F0, flights F1, flights F2
WHERE F0.fromplace = 'Bangalore'
AND F0.toplace = F1.fromplace
AND F1.toplace = F2.fromplace
AND F2.toplace = 'Kolkata'
AND F0.toplace <> 'Kolkata'
AND F1.toplace <> 'Kolkata'
AND F1.departs > F0.arrives
AND F2.departs > F1.arrives
AND extract(hour from F2.arrives) < 18));
```



The screenshot shows a database query result grid. The grid has two columns: 'fno' and 'departs'. The first row has values '102' and '2013-05-05 07:15:31'. The second row has values '106' and '2013-05-05 01:15:30'. The interface includes a toolbar with 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content' options. A sidebar on the right contains icons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Stats'. The bottom status bar shows 'aircraft 9', 'Result 10', 'employees 11', 'Result 12', 'employees 13', 'aircraft 14', and 'Result : Read Only'.

fno	departs
102	2013-05-05 07:15:31
106	2013-05-05 01:15:30