

## 1.Creating tables

```
CREATE SCHEMA insurance;

CREATE table person(
    driverid char(10),
    name char(20),
    address char(20),
    primary key(driverid)
);

CREATE table car(
    regnum char(10),
    model char(30),
    year int,
    primary key(regnum)
);

CREATE table accident(
    reportnum int,
    accdate date,
    location char(20),
    primary key(reportnum)
);

CREATE table owns(
    driverid char(10),
    regnum char(10),
    primary key(driverid,regnum),
    foreign key(driverid) references person(driverid),
    foreign key(regnum) references car(regnum)
);

CREATE table participated(
    driverid char(10),
    regnum char(10),
    reportnum int,
```

```

damageamount int,
primary key(driverid,regnum,reportnum),
foreign key(driverid) references person(driverid),
foreign key(regnum) references owns(regnum),
foreign key(reportnum) references accident(reportnum)
);

```

2.Inserting 5 tuples into each table

```

insert into person(driverid,name,address)

```

```

values

```

```

('A01', 'Richard', 'Srinivas nagar'),
('A02', 'Pradeep', 'Rajajinagar'),
('A03', 'Smith', 'Ashoknagar'),
('A04', 'Venu', 'N.R Colony'),
('A05', 'John', 'Hanumanthnagar');

```

```

INSERT INTO car (regnum, model, year) VALUES

```

```

('KA052250', 'Indica', 1990),
('KA031181', 'Lancer', 1957),
('KA095477', 'Toyota', 1998),
('KA053408', 'Honda', 2008),
('KA041702', 'Audi', 2005);

```

```

insert into accident(reportnum,accdate,location) values

```

```

(0011, '2003-01-01', 'Mysore Road'),
(0012, '2004-02-02', 'South end circle'),
(0013, '2003-01-21', 'Bull temple road'),
(0014, '2008-02-17', 'Mysore Road'),
(0015, '2005-03-04', 'Kanakapura Road');

```

```

INSERT INTO owns(driverid, regnum)

```

```

VALUES

```

```

('A01', 'KA052250'),
('A02', 'KA053408'),
('A04', 'KA031181'),
('A03', 'KA095477'),

```

```

('A05', 'KA041702');

INSERT INTO participated (driverid, regnum, reportnum, damageamount)
VALUES

('A01', 'KA052250', 0011, 1000),
('A02', 'KA053408', 0012, 50000),
('A03', 'KA095477', 0013, 25000),
('A04', 'KA031181', 0014, 3000),
('A05', 'KA041702', 0015, 5000);

```

3.Update the damage

amount to 25000 for the car with a specific reg-num(example 'KA053408') for which the accident report number was 12.


```

UPDATE participated

    set damageamount = 25000

    where regnum = "KA053408" AND reportnum=12;

```

	28 11:31:25	UPDATE participated	set damageamount = 25000	where regnum = "KA...	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.015 sec
---	-------------	---------------------	--------------------------	-----------------------	--	-----------

4.Add a new accident to the

database.

```

insert into accident values('0016','2008-02-12','Jalahalli');

```

5.Find the total number of

people who owned cars that involved in accidents in 2008.

```

SELECT COUNT(DISTINCT DRIVER_ID) FROM ACCIDENT, PARTICIPATED
WHERE ACCIDENT.REPORT_NUM = PARTICIPATED.REPORT_NUM
AND ACCIDENT_DATE LIKE '%08';

```

```

81 • SELECT COUNT(DISTINCT driverid) FROM accident, participated
82 WHERE accident.reportnum = participated.reportnum
83 AND accdate LIKE '%08';

```

<	
Result Grid	Filter Rows: <input type="text"/>
Export:	Wrap Cell Content: <a href="#">IA</a>
	COUNT(DISTINCT driverid)
▶	0

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

```

select count(*) from participated
join car on car.regnum=participated.regnum
where car.model='Audi';

```

```

75 • select count(*) from participated
76 join car on car.regnum=participated.regnum
77 where car.model='Audi';
78 • select count(*) from accident

```

<	
Result Grid	Filter Rows: <input type="text"/>
Export:	Wrap Cell Content: <a href="#">IA</a>
	count(*)
▶	1

## LAB-2

### BANKING ENTERPRISE

```

CREATE schema
banking_enterprise;

```

```

CREATE table branch(
branch_name varchar(30),

```

```

branch_city varchar(30),
assets real,
primary key(branch_name)
);

create table bankaccount(
accno int,
branch_name varchar(30),
balance real,
primary key(accno),
foreign key(branch_name) references branch(branch_name)
);

create table bankcustomer(
customername varchar(30),
customerstreet varchar(30),
customercity varchar(30),
primary key(customername)
);

create table depositer(
customername varchar(30),
accno int,
primary key(customername,accno),
foreign key(customername) references bankcustomer(customername),
foreign key(accno) references bankaccount(accno)
);

create table loan(
loannum int,
branch_name varchar(30),
amount real,
foreign key(branch_name) references branch(branch_name)
);

insert into branch(branch_name,branch_city,assets) values
('SBI_jayanagar','Bangalore','60000'),

```

```

('SBI_basavangudi','Bangalore','50000'),
('SBI_shivajinagar','Mumbai','10000'),
('SBI_jantarmentar','Dehli','20000'),
('SBI_parilmentroad','Dehli','30000');
insert into bankaccount(accno,branch_name,balance) values
('1','SBI_jayanagar','3000'),
('2','SBI_shivajinagar','2000'),
('3','SBI_jantarmentar','3000'),
('4','SBI_parilmentroad','1000'),
('5','SBI_basavangudi','4000'),
('6','SBI_shivajinagar','1000');
insert into bankaccount(accno,branch_name,balance) values
('7','SBI_jayanagar','1000'),
('8','SBI_jayanagar','3000');
insert into bankaccount(accno,branch_name,balance) values
('9','SBI_basavangudi','4000');
insert into bankcustomer(customername,customerstreet,customercity)
values
('rishab','bulltempleroad','Bangalore'),
('surabhi','highstreet','Bangalore'),
('kunal','avenueroad','Bangalore'),
('priyanka','akbarroad','Dehli'),
('rohan','prithviroad','Mumbai');
insert into depositer(customername,accno) values
('kunal','1'),
('surabhi','4'),
('priyanka','6'),
('rishab','3'),
('rohan','5'),
('kunal','2');
insert into depositer(customername,accno) values
('kunal','7');

```

```

insert into depositer(customername,accno) values
('kunal','9');

insert into loan(loannum,branch_name,amount) values
('1','SBI_jayanagar','2000'),
('2','SBI_shivajinagar','3000'),
('3','SBI_jantarmantar','1000'),
('4','SBI_basavangudi','4000'),
('5','SBI_parilmentroad','5000');

select customername from depositer
join bankaccount on depositer.accno=bankaccount.accno where
bankaccount.branch_name='SBI_jayanagar'
group by depositer.customername having
count(depositer.customername)>=2;

```

```

71 • select customername from depositer
72 join bankaccount on depositer.accno=bankaccount.accno where bankaccount.branch_name='SBI_jayanagar'
73 group by depositer.customername having count(depositer.customername)>=2
74
75

```

customername
kunal

```

select customername from depositer
join bankaccount on depositer.accno=bankaccount.accno
join branch on branch.branch_name=bankaccount.branch_name where
branch.branch_city='Bangalore'
group by depositer.customername having count(distinct
branch.branch_name)=(select count(branch.branch_name) from branch
where branch_city='Bangalore');

```

```

78 • select customername from depositer
79 join bankaccount on depositer.accno=bankaccount.accno
80 join branch on branch.branch_name=bankaccount.branch_name where branch.branch_city='Bangalore'
81 group by depositer.customername having count(distinct branch.branch_name)=(select count(branch.branc
82

```

customername
kunal

```

delete from bankaccount where branch_name in(select branch_name
from branch where branch_city='Dehli');

```

## LAB-3

### SUPPLIER DATABASE

```
create
schema
supplie
r;

create table suppliers(
sid int,

sname varchar(250),

address varchar(300),

primary key(sid)

);

create table parts(
pid int,

pname varchar(250),

color varchar(250),

primary key(pid)

);

create table catalog(
sid int,

pid int,

cost real,

foreign key(sid) references suppliers(sid),

foreign key(pid) references parts(pid)

);

INSERT into suppliers(sid,sname,address) values
('1001','johns','bangalore'),
('1002','vimal','mumbai'),
('1003','reliance','dehli'),
('1004','acme widget','kolkata');

insert into parts(pid,pname,color) values
('2001','book','red'),
('2002','pen','red'),
```



```

('2003','pencil','green'),
('2004','mobile','green'),
('2005','charger','black');

insert into catalog(sid,pid,cost) values

('1001','2001','10'),
('1001','2002','10'),
('1001','2003','30'),
('1001','2004','10'),
('1001','2005','10'),
('1002','2001','10'),
('1002','2002','20'),
('1003','2003','30'),
('1004','2003','40');

select distinct parts.pname from parts , catalog
where parts.pid= catalog.pid;

```

Result Grid		Filter Rows:	
	pname		
▶	book		
	pen		
	pencil		
	mobile		
	charger		

```

select s.sname from suppliers s

where s.sid in (select c.sid from catalog c group by c.sid having
count(distinct (c.pid))=(select count(p.pid) from parts p));

```

Result Grid		Filter Rows:	
	sname		
▶	johns		



```
select s.sname from suppliers s
where s.sid in (select ca.sid from catalog ca,parts p
where ca.pid=p.pid and p.color='red' group by ca.sid having
count(ca.pid)=(select count(*) from parts p where p.color='red'));
```

Result Grid		Filter Rows:	
	sname		
▶	johns		
	vimal		



Resets all sorted col

```
select ca.pid from catalog ca where ca.sid=(select s.sid from suppliers
s
where s.sname ='acme widget')
having (select count(c.pid) from catalog c
where c.pid=ca.pid)=1;
```

```
select distinct c.sid,c.pid from catalog c where c.cost > (select  
avg(ca.cost) from catalog ca  
where ca.pid=c.pid);
```

Result Grid					Filter Rows
	sid	pid			
▶	1002	2002			
	1004	2003			

```
select s.sname from suppliers s where s.sid in (select c.sid from catalog  
c  
where c.cost=(select max(cost) from catalog ca where ca.pid=c.pid));
```

Result Grid				Filter Rows
	sname			
▶	johns			
	vimal			
	acme widget			

```
create
schema
student
faculty
;
```

```
create table student(
    snum int,
    sname varchar(200),
    major varchar(200),
    lvl varchar(200),
    age int,
    primary key(snum));
```

```
create table class(
    cname varchar(200),
    meetstat time,
    room varchar(200),
    fid int,
    primary key(cname),
    foreign key(fid) references faculty(fid));
```

```
create table enrolled(
    snum int,
    cname varchar(200),
    foreign key(snum) references student(snum),
    foreign key(cname) references class(cname));
```

```
create table faculty(
    fid int,
```

```
fname varchar(200),  
deptid int,  
primary key (fid));
```

```
insert into student(snum,sname,major,lvl,age) values  
( '1','john','CS','sr','19'),  
( '2','steve','CS','jr','20'),  
( '3','dan','CV','sr','20'),  
( '4','brandon','CS','jr','20'),  
( '5','matt','CS','jr','20'),  
( '6','jamie','CS','sr','21');
```

```
insert into faculty(fid,fname,deptid) values  
( '11','harish','1000'),  
( '12','MV','1000'),  
( '13','david','1001'),  
( '14','noah','1002'),  
( '15','timothy','1000');
```

```
insert into class(cname,meetstat,room,fid) values  
( 'Class1','12/11/15 10:15:16','R1','14'),  
( 'Class10','12/11/15 10:15:16','R128','14'),  
( 'Class2','12/11/15 10:15:20','R2','12'),  
( 'Class3','12/11/15 10:15:25','R3','11'),  
( 'Class4','12/11/15 20:15:20','R4','14'),  
( 'Class5','12/11/15 20:15:20','R3','15'),  
( 'Class6','12/11/15 13:20:20','R2','14'),  
( 'Class7','12/11/15 10:10:10','R3','14');
```

```
insert into enrolled(snum,cname) values
```

```
('1','class1'),
```

```
('2','class1'),
```

```
('3','class3'),
```

```
('4','class3'),
```

```
('5','class4');
```

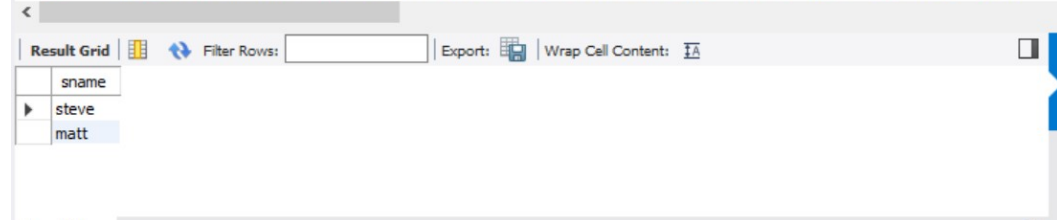
i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by "name"

```
select distinct s.sname
```

```
from student s,enrolled e,faculty f,class c
```

```
where s.snum=e.snum and e.cname=c.cname and f.fid=c.fid and  
f.fname='noah' and s.lvl='jr';
```

```
54 • select distinct s.sname  
55 from student s,enrolled e,faculty f,class c  
56 where s.snum=e.snum and e.cname=c.cname and f.fid=c.fid and f.fname='noah' and s.lvl='jr';
```



sname
steve
matt

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

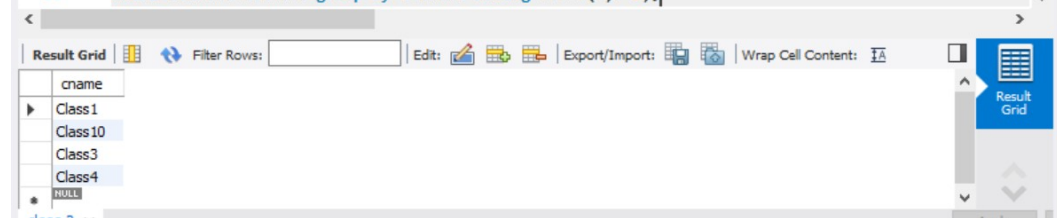
```
select cname from class c
```

```
where c.room='R128'
```

```
or c.cname in (select e.cname from enrolled e,class
```

```
where c.cname=e.cname group by e.cname having count(*)>=5);
```

```
57 • select cname from class c  
58 where c.room='R128'  
59 or c.cname in (select e.cname from enrolled e,class  
60 where c.cname=e.cname group by e.cname having count(*)>=5);
```



cname
Class1
Class10
Class3
Class4
NULL

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

```
select s.snum 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.meetstat=c2.meetstat);
```

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

```
select f.fname from faculty f

where not exists (

select distinct c.room from class c

where c.room not in (

select c1.room from class c1 where c1.fid=f.fid

)

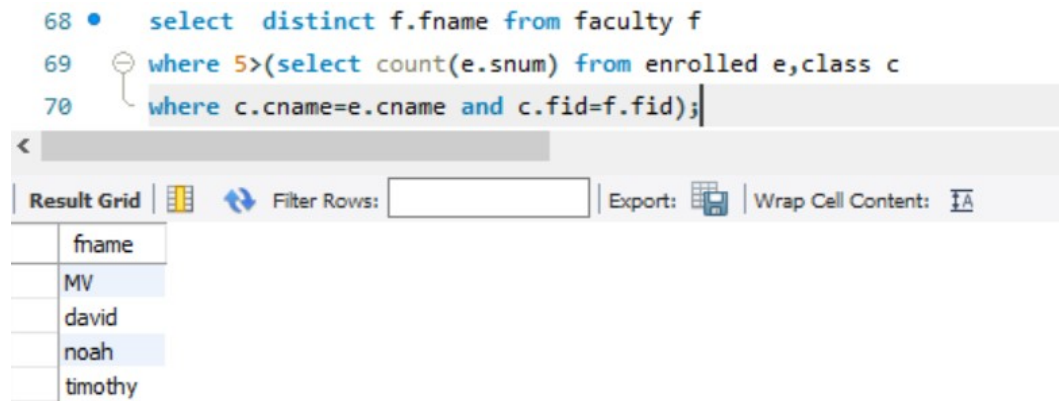
);
```

```
65 • select f.fname from faculty f
66   where not exists (
67     select distinct c.room from class c
68     where c.room not in (
69       select c1.room from class c1 where c1.fid=f.fid
70     )
71   );
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
fname				
noah				

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

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



The screenshot shows a SQL query editor with the following code:

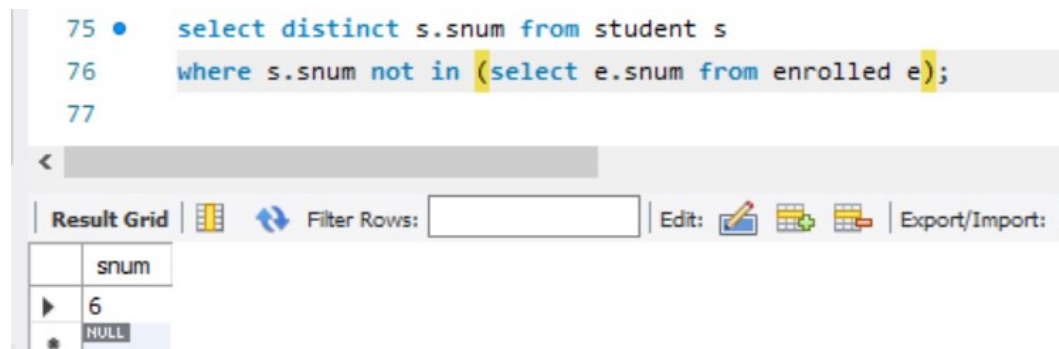
```
68 • select distinct f.fname from faculty f
69   where 5 > (select count(e.snum) from enrolled e, class c
70   where c.cname=e.cname and c.fid=f.fid);
```

Below the editor is a 'Result Grid' showing the output of the query:

fname
MV
david
noah
timothy

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

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



The screenshot shows a SQL query editor with the following code:

```
75 • select distinct s.snum from student s
76   where s.snum not in (select e.snum from enrolled e);
77
```

Below the editor is a 'Result Grid' showing the output of the query:

snum
6
NULL

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



```
where s1.age=s2.age
group by s2.lvl,s2.age))
```

```
74 • select s.age,s.lvl from student s
75     group by s.age,s.lvl
76     having s.lvl in(select s1.lvl from student s1
77                     where s1.age=s.age
78                     group by s1.lvl,s1.age
79                     having count(*)>=all(select count(*)from student s2
80                                         where s1.age=s2.age
81                                         group by s2.lvl,s2.age));
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	age	lvl			
▶	19	sr			
	20	jr			
	21	sr			

## LAB-5

```
create
databas
e
flightd
b;

create table flights(
    flno int,
    fromplace varchar(15),
    toplace varchar(15),
    distance int,
```

```

        departs datetime,
        arrives datetime,
        price int,
        primary key (flno)
    );

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

create table employees (
        eid int,
        ename varchar(15),
        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
flights(flno,fromplace,toplace,distance,departs,arrives,price) values
(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),
(107, 'Bangalore', 'Frankfurt', 8000, '2013-05-05 07:15:31', '2013-05-05
22:15:31', 60000),
(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),
(108, 'Lucknow', 'Kolkata', 1000, '2013-05-05 11:30:30', '2013-05-05
15:20:30', 10000);
```

```
insert into aircraft(aid,aname,cruisingrange)values
```

```
(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);
```

```
insert into employees(eid,ename,salary)values
```

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

```
insert into certified (eid,aid) values
```

```
(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);

```

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
    )
);

```

```

75 • select distinct a.aname from aircraft a where a.aid in (
76     select c.aid from certified c, employees e where
77     c.eid = e.eid and not exists(
78         select * from employees e1 where e1.eid=e.eid and e1.salary<80000
79     )
80 );
81

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	aname			
▶	747			
	Dreamliner			
	Dream			
	707			

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 max(a.cruisingrange),
```

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

```
--  
83 • select max(a.cruisingrange),  
84       c.eid from certified c, aircraft a where c.aid = a.aid group by c.eid having count(c.eid)>3;  
85
```

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	max(a.cruisingrange)	eid
▶	3500	701
	120000	702

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

```
select ename from employees where salary <(  

```

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

```
86 • select ename from employees where salary <(  
87       select min(price) from flights where fromplace='Bangalore' and toplace='Frankfurt');  
88
```

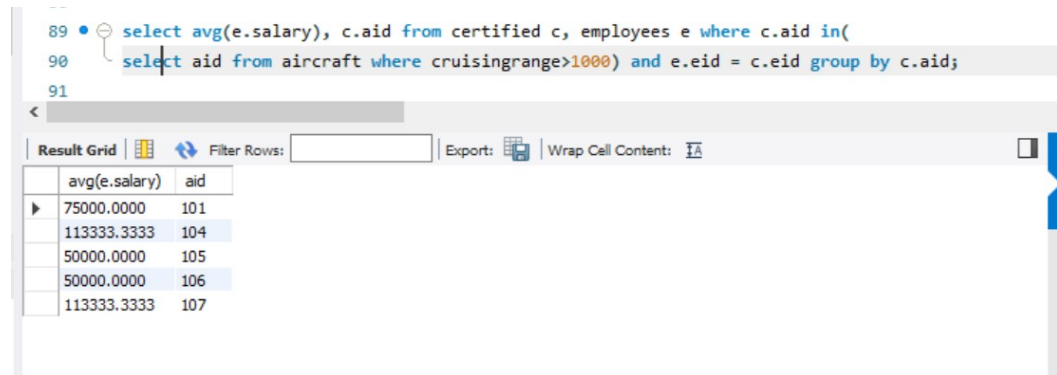
<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	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 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;
```

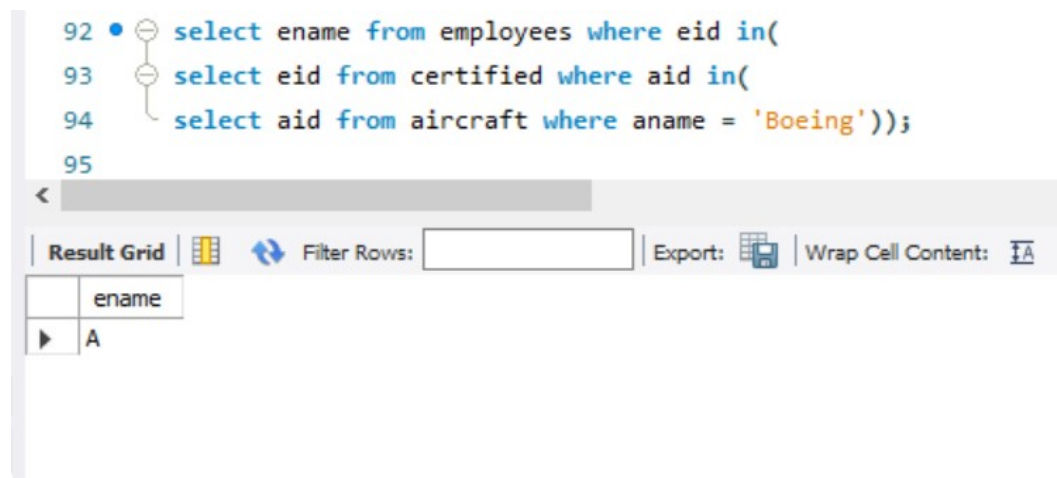


The screenshot shows a SQL query editor with a query on lines 89-91. Below the query is a 'Result Grid' with columns 'avg(e.salary)' and 'aid'. The grid contains five rows of data.

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

Find the names of pilots certified for some Boeing aircraft.

```
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 SQL query editor with a query on lines 92-95. Below the query is a 'Result Grid' with a single column 'ename'. The grid contains one row with the value 'A'.

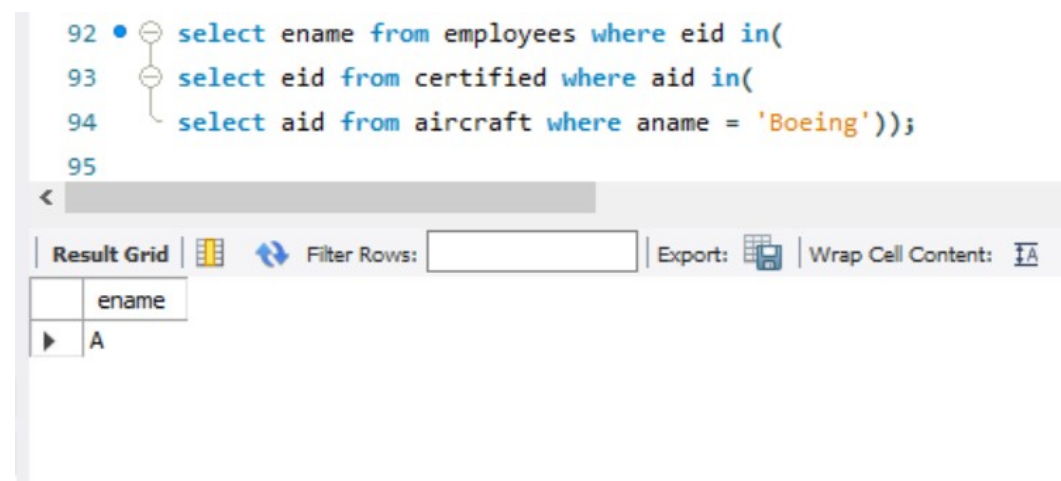
ename
A

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

```
select aname from aircraft
```

```
where cruisingrange > any (select distance from flights where  
fromplace='Bangalore'
```

```
and toplace='Delhi');
```



The screenshot shows a SQL IDE interface. The query editor contains the following SQL code:

```
92 • select ename from employees where eid in(  
93 select eid from certified where aid in(  
94 select aid from aircraft where aname = 'Boeing'));  
95
```

Below the query editor is a toolbar with icons for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. The 'Result Grid' icon is active, and a small table is displayed below it:

	ename
▶	A

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.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));

```

Result Grid			Filter Rows:
	fno	departs	
▶	102	2013-05-05 07:15:31	
	106	2013-05-05 01:15:30	