

SUSHMITHA Y V

1BM19CS165

LAB PROGAM-1:INSURANCE

```
create database
Insurance;

show databases;

use Insurance;

show tables;

create table PERSON(driverid varchar(20), dname
varchar(20), address varchar(40), primary key(driverid));
desc PERSON;

create table CAR(regno varchar(10), model varchar(10), year int
, primary key(regno));
desc CAR;

create table ACCIDENT(report_no int, adate date, location
varchar(20), primary key(report_no));
desc ACCIDENT;

create table OWNS(driverid varchar(10), regno varchar(10), primary
key(driverid, regno),
foreign key(driverid) references PERSON(driverid) on delete
cascade,
foreign key(regno) references CAR(regno) on delete cascade);

CREATE TABLE PARTICIPATED(driverid varchar(10), regno
varchar(10), report_no int,
damage_amt float, foreign key (driverid, regno) references
OWNS(driverid, regno)
ON DELETE CASCADE, foreign key (REPORT_NO) references
ACCIDENT(REPORT_NO) ON DELETE CASCADE);
desc PARTICIPATED;
```

```
insert into PERSON values('1111','Ramu','K.S.LAYOUT');

commit;

select* FROM PERSON;

insert into PERSON values('2222','John','INDIRANAGAR');

insert into PERSON values('3333','Priya','JAYANAGAR');

insert into PERSON values('4444','Gopal','WHITEFIELD');

insert into PERSON values('5555','Latha','VIJAYNAGAR');

commit;

insert into CAR values('KA04Q2301','MARUTHI-DX', 2000);

insert into CAR values('KA05P1000', 'FORDICON', 2000);

insert into CAR values('KA03L1234',' ZEN-VXI',1999);

insert into CAR values('KA03L9999', 'MARUTHI-DX', 2002);

insert into CAR values('KA01P4020', 'INDICA-VX', 2002);

commit;

select * from CAR;
```

```
insert into ACCIDENT values(12,'2002-06-01', 'M G ROAD');

insert into ACCIDENT values(200, '2002-12-10', 'DOUBLEROAD');

insert into ACCIDENT values(300, '1999-07-23', 'M G ROAD');

insert into ACCIDENT values(25000, '2000-06-11', 'RESIDENCY
ROAD');

insert into ACCIDENT values(26500, '2001-10-16', 'RICHMOND
CIRCLE');

commit;

select * from ACCIDENT;
```

```
insert into OWNS values('1111','KA04Q2301');

insert into OWNS values('1111', 'KA05P1000');

insert into OWNS values('2222', 'KA03L1234');

insert into OWNS values('3333', 'KA03L9999');

insert into OWNS values('4444', 'KA01P4020');

commit;
```

```
select * from OWNS;

insert into PARTICIPATED values('1111', 'KA04Q2301', 12 ,20000);
insert into PARTICIPATED values('2222', 'KA03L1234', 200, 500);
insert into PARTICIPATED values('3333', 'KA03L9999', 300,
10000);
insert into PARTICIPATED values('4444', 'KA01P4020', 25000
,2375);
insert into PARTICIPATED values('1111', 'KA05P1000', 26500
,70000);

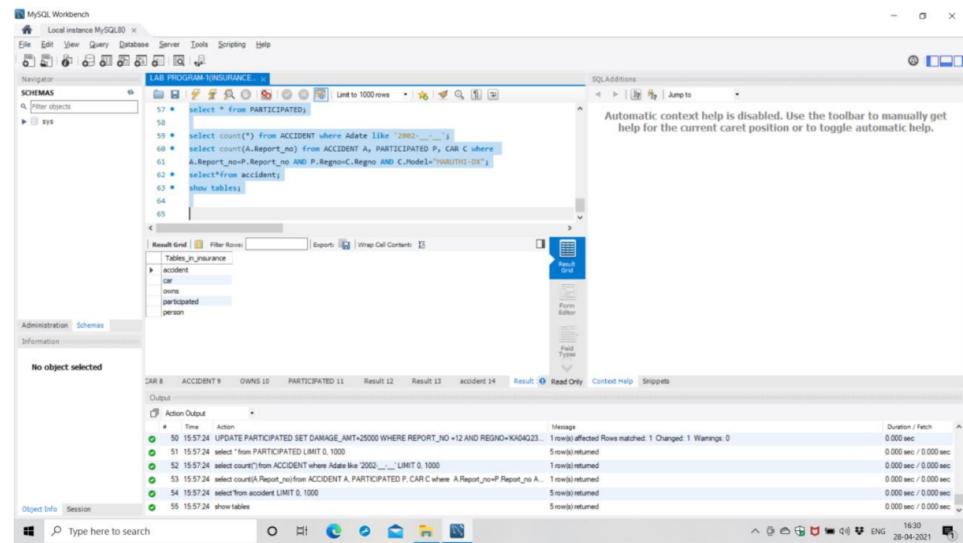
UPDATE PARTICIPATED SET DAMAGE_AMT=25000 WHERE REPORT_NO =12 AND
REGNO='KA04Q2301';
select * from PARTICIPATED;

select count(*) from ACCIDENT where Adate like '2002-__-__';
select count(A.Report_no) from ACCIDENT A, PARTICIPATED P, CAR C
where
A.Report_no=P.Report_no AND P.Regno=C.Regno AND
C.Model="MARUTHI-DX";
```

PROGRAM-1

INSURANCE-OUTPUT

SHOW TABLES:



The screenshot shows the MySQL Workbench interface. In the SQL Editor tab, there is a query window titled 'LAB PROGRAM-INSURANCE' containing the following SQL code:

```
select * from PARTICIPATED;
select count(*) from ACCIDENT where Adate like '2002-__-__';
select count(A.Report_no) from ACCIDENT A, PARTICIPATED P, CAR C where
A.Report_no=P.Report_no AND P.Regno=C.Regno AND C.Model='HARITHI-DX';
select * from accident;
show tables;
```

The Results Grid tab shows the output of the last two statements:

Tables_in_insurance
accident
car
owns
participated
person

At the bottom of the interface, the status bar displays '1630 28-04-2021'.

SELECT * FROM insurance.accident;

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS insurance

Tables accident car owner participated person

Columns report_no date location

Information

Table: accident

Columns:

report_no	int PK
date	date
location	varchar(20)

Action Output

Time Action

1 16:33:12 SELECT * FROM insurance.accident LIMIT 0, 1000

Result Grid

report_no	date	location
12	2002-06-01	M ROAD
200	2002-12-10	DOURJROAD
300	1999-07-23	M G ROAD
25000	2000-06-11	RESIDENCY ROAD
26300	2001-10-16	RICHMOND CIRCLE
...

Object Info Session

Type here to search

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Report No. 163312

Report Date: 28-04-2021

SELECT * FROM insurance.car;

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS insurance

Tables accident car owner participated person

Columns report_no date location

Information

Table: car

Columns:

report_no	int PK
date	date
location	varchar(20)

Action Output

Time Action

1 16:33:12 SELECT * FROM insurance.accident LIMIT 0, 1000

2 16:34:12 SELECT * FROM insurance.car LIMIT 0, 1000

Result Grid

regno	model	year
KAO-P4230	INDICA-VX	2002
KAO-33333	INDICA-VX	1999
KAO-3.9999	MARUTI-EKX	2002
KAO-Q2201	MARUTI-EKX	2000
KAO-SP2000	FORDECON	2000
...

Object Info Session

Type here to search

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Report No. 163412

Report Date: 28-04-2021

```
SELECT * FROM insurance.owns;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** LAB PROGRAM-INSURANCE
- Table:** owns
- Columns:** drivend, regno
- Result Grid:**

drivend	regno
4444	KAO-P4020
2222	KAO51234
3333	KAO-45999
1111	KAO-Q2301
1111	KAO-P1000
5555	5555
- Action Output:**

#	Time	Action	Message	Duration / Fetch
1	16:33:12	SELECT * FROM insurance.owns LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
2	16:34:12	SELECT * FROM insurance.car LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
3	16:34:51	SELECT * FROM insurance.owns LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

```
SELECT * FROM insurance.participated;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** LAB PROGRAM-INSURANCE
- Table:** participated
- Columns:** drivend, regno, report_no, damage_amt
- Result Grid:**

drivend	regno	report_no	damage_amt
1111	KAO-Q2301	12	25000
2222	KAO51234	200	500
3333	KAO-45999	300	10000
4444	KAO-P4020	2000	2375
1111	KAO-P1000	26500	7000
- Action Output:**

#	Time	Action	Message	Duration / Fetch
1	16:33:12	SELECT * FROM insurance.accident LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
2	16:34:12	SELECT * FROM insurance.car LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
3	16:34:51	SELECT * FROM insurance.owns LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
4	16:35:20	SELECT * FROM insurance.participated LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

```
SELECT * FROM insurance.person;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database schema with the **insurance** database selected. Under **Tables**, the **person** table is highlighted.
- SQL Editor:** Displays the query: `SELECT * FROM insurance.person;`
- Result Grid:** Shows the data from the **person** table:

drivnid	dname	address
1111	Renu	K.S.LAYOUT
2222	John	INDIANAGAR
3333	Priya	JAMIA NAGAR
4444	David	WHITEFIELD
5555	Latha	V.V.NAGAR
6666	OB	OB

Output: Shows the execution history of the query:

#	Action	Message	Duration / Fetch
1	1 16:33:12 SELECT * FROM insurance.person LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
2	2 16:34:12 SELECT * FROM insurance.car LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
3	3 16:34:51 SELECT * FROM insurance.cars LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
4	4 16:35:20 SELECT * FROM insurance.participated LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
5	5 16:35:50 SELECT * FROM insurance.person LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

SUSHMITA Y V

1BM19CS165

LAB-2 : BANKING ENTERPRISE DATABASE

BANKING ENTERPRISE DATABASE

```
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_Jantarmantar', 'Delhi', 20000);
commit;
select * from branch;

```

	branchname	branchcity	assets
▶	SBI_Chamrajpet	Bengaluru	50000
	SBI_Jantarmantar	Delhi	20000
	SBI_ParliamentRoad	Delhi	10000
	SBI_ResidencyRoad	Bengaluru	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_Jantarmantar', 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_Jantarmantar', 2000);
commit;
select * from bank_account;

```

	Accno	branchname	balance
▶	1	SBI_Chamrajpet	2000
	2	SBI_ResidencyRoad	5000
	4	SBI_ParliamentRoad	9000
	5	SBI_Jantarmantar	8000
	8	SBI_ResidencyRoad	4000
	9	SBI_ParliamentRoad	3000
	10	SBI_ResidencyRoad	5000
	11	SBI_Jantarmantar	2000

```

insert into bank_customer values ('Avinash', 'Bull_Temple_Road', 'Bangalore');
insert into bank_customer values ('Dinesh', 'BannerGatta_Road', 'Bangalore');
insert into bank_customer values ('Mohan', 'National_College_Road', 'Bangalore');
insert into bank_customer values ('Nikhil', 'Akbar_Road', 'Delhi');
insert into bank_customer values ('Ravi', 'Prithviraj_Road', 'Delhi');
commit;
select * from bank_customer;

```

customer_name	customer_street	customer_city
Avinash	Bull_Temple_Road	Bangalore
Dinesh	BannerGatta_Road	Bangalore
Mohan	National_College_Road	Bangalore
Nikhil	Akbar_Road	Delhi
Ravi	Prithviraj_Road	Delhi
NULL	NULL	NULL

```

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);
commit;
select * from depositer;

```

	CustomerName	Accno
▶	Avinash	1
	Dinesh	2
	Nikil	4
	Ravi	5
	Avinash	8
	Nikil	9
	Dinesh	10
	Nikil	11

```

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_Jantarmantar', 5000);
commit;
select * from loan;

```

	loannumber	branchname	Amount
▶	1	SBI_Chamrajpet	10000
	2	SBI_ResidencyRoad	20000
	3	SBI_ShivajiRoad	30000
	4	SBI_ParliamentRoad	40000
	5	SBI_Jantarmantar	30000
*	NULL	NULL	NULL

```

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);

```

Result Grid | Filter Rows: | Edit:

	CustomerName
▶	Dinesh
*	NULL

```
select distinct d.customer_name from depositer d where exists( select * from bank_account ba
where ba.accno=d.accno and exists (select * from branch b where b.branch_name =
ba.branch_name and b.branch_city='Delhi'));
```

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

	CustomerName
▶	Ravi
*	Nikil

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

Result Grid | Filter Rows: | Edit

	Accno	branchname	balance
▶	1	SBI_Chamrajpet	2000
	2	SBI_ResidencyRoad	5000
	4	SBI_ParliamentRoad	9000
	5	SBI_Jantarmantar	8000
	8	SBI_ResidencyRoad	4000
	9	SBI_ParliamentRoad	3000
	10	SBI_ResidencyRoad	5000
	11	SBI_Jantarmantar	2000

SUSHMITA Y V

1BM19CS165

LAB-3: SUPPLIER DATABASE

SUPPLIER DATABASE

```
create database Supplier;  
use Supplier;
```

```
create table Suppliers(  
sid varchar(20) ,  
sname varchar(20),  
city varchar(20),  
primary key(sid)  
);  
desc Suppliers;
```

```
create table Parts(  
pid integer,  
pname varchar(20),  
color varchar(20),  
primary key(pid)  
);  
desc Parts;
```

```
create table Catalog(  
sid varchar(20),  
pid integer,  
cost real,  
primary key(sid,pid),  
foreign key(sid) references Suppliers(sid),  
foreign key(pid) references Parts(pid)  
);  
desc Catalog;
```

```
insert into Suppliers values(10001,'Acme Widget','Bangalore');  
insert into Suppliers values(10002,'Johns','Kolkata');  
insert into Suppliers values(10003,'Vimal','Mumbai');  
insert into Suppliers values(10004,'Reliance','Delhi');  
insert into Suppliers values(10005,'Mahindra','Mumbai');  
select *from Suppliers;
```

Result Grid | Filter Rows: [] | Edit: [] | Export/Import: [] | W

sid	sname	city
10001	Acme Widget	Bangalore
10002	Johns	Kolkata
10003	Vimal	Mumbai
10004	Reliance	Delhi
10005	Mahindra	Mumbai
HULL	HULL	HULL

```
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');
select *from Parts;
```

Result Grid | Filter Rows: []

pid	pname	color
20001	Book	Red
20002	Pen	Red
20003	Pencil	green
20004	Mobile	green
20005	Charger	Black
HULL	HULL	HULL

```
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);
select *from 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
*	HULL	HULL	HULL

select distinct P.pname from Parts P, Catalog c where P.pid=C.pid;

Result Grid | Filter Rows: []

	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));

	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)));

	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');
```

pname
Mobile
Charger

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

sid
10002
10004

```
select s.sname ,p.pid from Suppliers s, Catalog c, Parts p where s.sid=c.sid and c.pid =p.pid and c.cost=(select max(ca.cost) from catalog ca where ca.pid=p.pid);
```

sname	pid
Acme Widget	20001
Johns	20001
Johns	20002
Reliance	20003
Acme Widget	20004
Acme Widget	20005

SUSHMITA Y V

1BM19CS165

LAB-4 : STUDENT FACULTY DATABASE

STUDENT FACULTY DATABASE

```
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));
desc student;

create table faculty(fid int, fname varchar(20), deptid int,primary key(fid));
desc faculty;

create table class(cname varchar(20), meetsat timestamp, room varchar(10), fid int,primary key
(cname),foreign key(fid) references faculty(fid));
desc class;

create table enrolled(snum int, cname varchar(20),primary key(snum,cname),
foreign key(snum) references student(snum),
foreign key(cname) references class(cname));
desc enrolled;

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);
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

```
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);
select * from faculty;

```

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

```

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', 12);
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);
select * from class;

```

	cname	meets_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
	HULL	HULL	HULL	HULL

```

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');

```

```
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));
```

	age	lvl
▶	19	Sr
	20	Jr
	21	Sr

SUSHMITA Y V

1BM19CS165

LAB PROGRAM-5: AIRLINE FLIGHT DATABASE

AIRLINE FLIGHT DATABASE

```
create database Airline_flight;
use Airline_flight;

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,
    fname 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);
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);
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
HULL	HULL	HULL

```

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);
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
HULL	HULL	HULL

```

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);
select * from certified;

```

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

```
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));
```

747
Dreamliner
Dream
707

```
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;
```

3500	701
120000	702

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

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;
```

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'));
```

ename
A

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

aname
747
Dreamliner
Boeing
Dream

```
select F.flno, F.departs
from flights F
Where F.flno in ( ( select F0.flno
from flights F0
where F0.fromplace = 'Bangalore' and F0.toplac = '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.deperts > 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.deperts > F0.arrives  
and F2.deperts > F1.arrives  
and extract(hour from F2.arrives) < 18));
```

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