

1. VTU 5th SEM DBMS :: STUDENT

```
CREATE TABLE STUDENT
(
  SNUM INT PRIMARY KEY,
  NAME VARCHAR(10),
  MAJOR VARCHAR(10),
  LEVELS VARCHAR(5),
  AGE INT
);
```

```
CREATE TABLE FACULTY
(
  FID VARCHAR(4) PRIMARY KEY,
  FNAME VARCHAR(10),
  DEPT_ID INT
);
```

```
CREATE TABLE CLASS
(
  CNAME VARCHAR(10) PRIMARY KEY,
  MEETS_AT VARCHAR(10),
  ROOM VARCHAR(10),
  FID VARCHAR(4),
  FOREIGN KEY(FID) REFERENCES FACULTY(FID)
);
```

```
CREATE TABLE ENROLLED
(
  SNUM INT,
  CNAME VARCHAR(10) REFERENCES CLASS(CNAME),
  FOREIGN KEY(SNUM) REFERENCES STUDENT(SNUM)
);
```

```
INSERT IN TO STUDENT VALUES :::
INSERT INTO STUDENT VALUES (1, 'BLAKE', 'CS', 'JR', 20);
INSERT INTO STUDENT VALUES (2, 'JIM', 'EC', 'SR', 19);
INSERT INTO STUDENT VALUES (3, 'JOHN', 'EC', 'SR', 21);
INSERT INTO STUDENT VALUES (4, 'CHRIS', 'EE', 'JR', 20);
INSERT INTO STUDENT VALUES (5, 'JAKE', 'ME', 'SR', 20);
```

```
INSERT IN TO FACULTY VALUES :
INSERT INTO FACULTY VALUES (1, 'HARSHITH', 10);
INSERT INTO FACULTY VALUES (2, 'KAMBER', 20);
INSERT INTO FACULTY VALUES (3, 'NAVATHE', 30);
```

INSERT IN TO CLASS VALUES :

```
INSERT INTO CLASS VALUES ('2SEM', '10AM', '401', 1);
INSERT INTO CLASS VALUES ('3SEM', '12PM', '128', 2);
INSERT INTO CLASS VALUES ('4SEM', '12PM', '601', 3);
INSERT INTO CLASS VALUES ('6SEM', '2PM', '128', 1);
INSERT INTO CLASS VALUES ('5SEM', '8AM', '401', 1);
INSERT INTO CLASS VALUES ('7SEM', '9AM', '601', 1);
```

INSERT IN TO ENROLLED VALUES :

```
INSERT INTO ENROLLED VALUES (1, '4SEM');
INSERT INTO ENROLLED VALUES (2, '6SEM');
INSERT INTO ENROLLED VALUES (3, '3SEM');
INSERT INTO ENROLLED VALUES (4, '2SEM');
INSERT INTO ENROLLED VALUES (5, '5SEM');
INSERT INTO ENROLLED VALUES (2, '4SEM');
INSERT INTO ENROLLED VALUES (3, '4SEM');
INSERT INTO ENROLLED VALUES (1, '3SEM');
INSERT INTO ENROLLED VALUES (1, '2SEM');
INSERT INTO ENROLLED VALUES (4, '6SEM');
INSERT INTO ENROLLED VALUES (5, '3SEM');
INSERT INTO ENROLLED VALUES (1, '5SEM');
INSERT INTO ENROLLED VALUES (5, '4SEM');
INSERT INTO ENROLLED VALUES (4, '4SEM');
```

Queries:

1. Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. HARSHITH

```
SELECT DISTINCT S.NAME
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 = 'HARSHITH' AND S.LEVELS = 'JR';
```

NAME

CHRIS

BLAKE

2. 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 = 128
OR C.CNAME IN (SELECT E.CNAME
FROM ENROLLED E
GROUP BY E.CNAME
HAVING COUNT (*) > 4);

```

```

CNAME
-----
3SEM
4SEM
6SEM

```

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

```

SELECT DISTINCT S.NAME
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.MEETS_AT = C2.MEETS_AT);

```

```

NAME
-----
JAKE
JOHN
BLAKE

```

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

```

SELECT DISTINCT F.FNAME , FID
FROM FACULTY F
WHERE NOT EXISTS (
( SELECT DISTINCT C.ROOM FROM CLASS C )
MINUS
( SELECT DISTINCT C1.ROOM FROM CLASS C1 WHERE C1.FID = F.FID ));

```

```

FNAME  FID
-----
HARSHITH  1

```

5. 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 CLASS C, ENROLLED E
WHERE C.CNAME = E.CNAME
AND C.FID = F.FID);
```

FNAME

KAMBER

2. VTU 5th SEM DBMS: FLIGHT INFORMATION

```
create table aircraft
(
aid integer primary key,
aname varchar(15),
crange integer
);
```

```
create table flights
(
flno integer primary key references aircraft(aid),
fromplace varchar(15),
toplace varchar(15),
distance integer,
departs timestamp,
arrives timestamp,
price integer
);
```

```
create table employees
(
eid integer primary key,
ename varchar(15),
salary integer
);
```

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

```
INSERT IN TO AIRCRAFT VALUES::
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,'Dreamliner2',12000);
```

```
INSERT IN TO EMPLOYEES VALUES::
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',70000);
```

```
INSERT INTO CERTIFIED VALUES::
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(703,105);
insert into certified values(704,105);
insert into certified values(705,103);

```

```

insert into flights values(101,'Bangalore','Dehi',2500,TIMESTAMP '2005-05-13
07:15:31',TIMESTAMP '2005-05-13 07:15:31',5000);
insert into flights values(102,'Bangalore','Lucknow',3000,TIMESTAMP '2005-05-13
07:15:31',TIMESTAMP '2005-05-13 07:15:31',6000);
insert into flights values(103,'Lucknow','Dehi',500,TIMESTAMP '2005-05-13
07:15:31',TIMESTAMP '2005-05-13 07:15:31',3000);
insert into flights values(107,'Bangalore','Frankfurt',8000,TIMESTAMP '2005-05-13
07:15:31',TIMESTAMP '2005-05-13 07:15:31',60000);
insert into flights values(104,'Bangalore','Frankfurt',8500,TIMESTAMP '2005-05-13
07:15:31',TIMESTAMP '2005-05-13 07:15:31',75000);
insert into flights values(105,'Kolkata','Dehi',3400,TIMESTAMP '2005-05-13
07:15:31',TIMESTAMP '2005-05-13 07:15:31',7000);

```

i. 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 NOT IN (SELECT C.AID
FROM CERTIFIED C, EMPLOYEES E
WHERE C.EID = E.EID AND E.SALARY<80000);
or

```

```

SELECT DISTINCT A.ANAME
FROM AIRCRAFT a,CERTIFIED C, EMPLOYEES E
WHERE a.aid=c.aid and c.eid=e.eid AND E.SALARY<80000;

```

```

ANAME
-----
Dreamliner
Dreamliner2

```

ii. 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.CRANGE)

```

```

FROM CERTIFIED C, AIRCRAFT A
WHERE C.AID = A.AID
GROUP BY C.EID
HAVING COUNT (*) >= 3;

```

```

EID  MAX(A.CRANGE)
-----

```

```

701      3500
703      12000
704      12000
702      12000

```

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

```

select distinct ename
FROM EMPLOYEES E
WHERE E.SALARY < ( SELECT MIN (F.PRICE)
FROM FLIGHTS F
WHERE F.fromplace ='Bangalore' AND F.toplace = 'Frankfurt');

```

```

ENAME
-----

```

```

A
E

```

iv. 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 name,avgsalary
from (select a.aid,a.aname as name,avg(e.salary) as avgsalary
from aircraft a,certified c,employees e where a.aid=c.aid and e.eid=c.eid and a.crange>1000
group by a.aid,a.aname);

```

```

NAME          AVGSALARY
-----

```

```

747           75000
Dreamliner    113333.333
Dreamliner2   113333.333
707           50000
Boeing        96666.6667

```

5. 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 = 'Boeing';

```

ENAME

D

A

C

vi. 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.CRANGE >= (SELECT MIN (F.DISTANCE)
FROM FLIGHTS F
WHERE F.FROMplace = 'Bangalore' AND F.TOplace = 'Dehi');

```

AID

101

104

105

107

3rd DBMS::Student !!

```

create table student
(
  regno varchar(10) primary key,
  name varchar(10),
  major varchar(10),
  bdate date);

```

```

create table course
(
  cno int primary key,
  cname varchar(10),
  dept varchar(10));

```

```

create table enroll
(
  regno varchar(10),

```

```
cno int,  
    sem int,  
marks int,  
    foreign key (regno) references student(regno),  
    foreign key (cno) references course(cno)  
);
```

```
create table text  
(  
    isbn int primary key,  
    title varchar(10),  
    publisher varchar(10),  
    author varchar(10));
```

```
create table bookad  
(  
    cno int,  
    sem int,  
    isbn int,  
    foreign key(cno) references course(cno),  
    foreign key (isbn) references text(isbn)  
);
```


```
insert into student  
values('1ss04is400','Thor','BE','02sep1984');
```

```
insert into student  
values('1ss03is062','Spidey','BE','28jan1984');
```

```
insert into student  
values('1ss03is039','Goku','BE','21dec1983');
```

```
insert into student  
values('1ss03is009','Loki','BE','04may1984');
```

```
insert into student  
values('1ss03is006','Pikachu','BE','04sep1984');
```

```
insert into course  
values(1,'DBMS','CSE');
```

```
insert into course  
values(2,'c','CSE');
```

```
insert into course  
values(3,'FAFL','ISE');
```

```
insert into course  
values(4,'OS','ISE');
```

```
insert into course  
values(5,'MMC','ISE');
```

```
insert into enroll  
values('1ss04is400',05,6,78);
```

```
insert into enroll  
values('1ss03is062',02,6,78);
```

```
insert into enroll  
values('1ss03is039',05,6,78);
```

```
insert into enroll  
values('1ss03is009',04,6,78);
```

```
insert into enroll  
values('1ss03is006',01,6,78);
```

```
insert into text  
values(11,'database','pearson','navate');
```

```
insert into text  
values(12,'let us c','pearson','kanetkar');
```

```
insert into text  
values(13,'fafl','sapna','padmareddy');
```

```
insert into text  
values(14,'os','phi','galvin');
```

```
insert into text  
values(15,'mmc','ORAIN','ralf');
```

```
insert into bookad
values(1,6,11);
```

```
insert into bookad
values(2,6,12);
```

```
insert into bookad
values(2,6,13);
```

```
insert into bookad
values(2,6,14);
```

Q3:

```
insert into text
values(16,'c++','pearson','kanetkar');
```

1 row created.

```
insert into bookad
values(2,6,16);
```

1 row created.

Q4:

```
select c.cno,t.isbn,t.title
from course c,text t,bookad b
where c.cno=b.cno and
      t.isbn=b.isbn and
      c.dept='CSE' and c.cno in(select b.cno
                                from bookad b
                                group by b.cno
                                having count(*)>1)
group by c.cno ,t.isbn,t.title
order by c.cno,t.title;
```

RESULTS:

CNO	ISBN	TITLE
-----	------	-------

```

-----
2      16 c++
2      13 fafl
2      12 let us c
2      14 os
-----
-----

```

```

5.>select distinct c.dept
    from course c, bookad ba, text t
    where t.ISBN=ba.ISBN and ba.cno=c.cno and
          t.publisher='pearson';

```

RESULTS:

DEPT

CSE

```

-----
-----
select * from student;
select * from course;
select * from enroll;
select * from bookad;
select * from text;

```

4th DBMS::Book deler

```

create table author
(
aid int primary key,
name varchar (10) not null,
city varchar(10) not null,
country varchar(10) not null
);

```

```

create table publisher
(
pid int primary key,
name varchar(10) not null,
city varchar(10) not null,
country varchar(10) not null

```

```
);
```

```
create table category  
(  
  cid int primary key,  
  description varchar(10) not null  
);
```

```
create table catalog  
(  
  bid int primary key,  
  title varchar(10) not null,  
  aid not null,  
  pid not null,  
  cid not null,  
  year int not null,  
  price int not null,  
  foreign key(aid) references author(aid),  
  foreign key(pid) references publisher(pid),  
  foreign key(cid) references category(cid)  
);
```

```
create table odetails  
(  
  ono int primary key,  
  bid int not null,  
  quantity int not null,  
  foreign key(bid) references catalog(bid)  
);
```

AUTHOR

```
insert into author values (1,'ironman','mangalore','india');  
insert into author values (2,'goku','colombo','srilanka');  
insert into author values (3,'thor','sydney','australia');  
insert into author values (4,'ananya','bangalore','usa');  
insert into author values (5,'aparna','delhi','uk');
```

PUBLISHER

```
insert into publisher values(101,'tata','mangalore','srilanka');  
insert into publisher values(102,'nano','colombo','italy');  
insert into publisher values(103,'bmw','sydney','usa');  
insert into publisher values(104,'pearson','delhi','india');
```

```
insert into publisher values(105,'sapna','bangalore','uk');
```

CATEGORY

```
insert into category values (1001,'comp');
insert into category values (1002,'ele');
insert into category values (1003,'maths');
insert into category values (1004,'science');
insert into category values (1005,'mech');
```

CATALOG

```
insert into catalog values (111,'lib1',1,101,1001,2000,20);
insert into catalog values (112,'lib2',2,102,1002,2001,80);
insert into catalog values (113,'lib3',3,103,1003,2002,20);
insert into catalog values (114,'lib4',4,104,1001,2003,35);
insert into catalog values (115,'lib5',5,105,1004,2004,10);
insert into catalog values (116,'lib6',2,103,1005,2005,60);
insert into catalog values (117,'lib7',2,105,1002,2006,45);
insert into catalog values (118,'lib8',1,101,1001,2007,50);
```

odetailsinsert

```
insert into odetails values (10,111,2);
insert into odetails values (20,112,3);
insert into odetails values (30,111,5);
insert into odetails values (40,113,1);
insert into odetails values (50,114,2);
```

```
select * from author;
select * from catalog;
select * from odetails;
select * from category;
select * from publisher;
```

```
1.select a.aid,a.name,a.city,count(*) as count
from author a,catalog c
where a.aid=c.aid and c.year>2000
and c.price>=(select avg(price)
from catalog)
group by(a.aid,a.name,city)
having count(*)>=2;
```

```
2.select distinct(a.name)
from author a,catalog c,odetails odm
```

```
where a.aid=c.aid and odm.bid=c.bid
and exists(select od.bid,sum(od.quantity)
from odetails od
where od.bid=odm.bid
group by bid
having sum(od.quantity)>=all(select sum(quantity)
from odetails
group by bid));
```

OR \OR

```
select a.name from author a,
catalog c,odetails odm
where a.aid =c.aid and odm.bid=c.bid
and odm.quantity=(select max (quantity) from odetails);
```

```
3.update catalog
set price=1.1*price
where pid in (select pid
from publisher
where name='tata');
```

5.VTU 5th SEM DBMS::Bank Enterprise !!

```
create table branch
(
bname varchar(10) primary key,
```

```
bcity varchar(10),  
asset real  
);
```

```
create table account  
(  
accno int primary key,  
bname varchar(10),  
balance real not null,  
foreign key(bname) references branch(bname)on delete cascade  
);
```

```
create table customer  
(  
cname varchar(10) primary key,  
cstreet varchar(10) not null,  
city varchar(10) not null  
);
```

```
create table loan  
(  
lno int primary key,  
bname varchar(10) not null,  
amt real,  
foreign key(bname) references branch(bname)on delete cascade  
);
```

```
create table borrower  
(  
cname varchar(10),  
lno int,  
foreign key(cname) references customer(cname)on delete cascade,  
foreign key(lno) references loan(lno)on delete cascade  
);
```

```
create table depositor  
(  
cname varchar(10),  
accno int,  
foreign key (cname)references customer(cname)on delete cascade,  
foreign key(accno)references account(accno)on delete cascade  
);
```

```
insert into branch values('bit','bang',12000);
insert into branch values('jpnagar','cal',22000);
insert into branch values('mgroad','mum',33000);
insert into branch values('bsk','hyd',55000);
insert into branch values('rajaji','bang',99000);
insert into branch values('rsk','hyd',54000);
```

```
insert into account values(1,'bit',250);
insert into account values(2,'jpnagar',120);
insert into account values(3,'mgroad',10);
insert into account values(4,'bsk',100);
insert into account values(5,'rajaji',600);
insert into account values(6,'rajaji',600);
insert into account values(7,'bsk',600);
```

```
insert into customer values('hulk','bit','bang');
insert into customer values('ironman','jpnagar','bang');
insert into customer values('thor','mgroad','bang');
insert into customer values('loki','bsk','bang');
insert into customer values('batman','rajaji','bang');
insert into customer values('goku','ask','vell');
insert into customer values('vegeta','raji','italy');
```

```
insert into loan values(1,'bit',5000);
insert into loan values(2,'jpnagar',1500);
insert into loan values(3,'mgroad',10000);
insert into loan values(4,'bsk',3500);
insert into loan values(5,'rajaji',20000);
```

```
insert into borrower values('hulk',1);
insert into borrower values('ironman',2);
insert into borrower values('thor',3);
insert into borrower values('loki',4);
insert into borrower values('batman',5);
insert into borrower values('hulk',2);
```

```
insert into depositor values('hulk',1);
insert into depositor values('ironman',2);
insert into depositor values('thor',3);
insert into depositor values('loki',4);
insert into depositor values('batman',5);
```

```
insert into depositor values('batman',6);
insert into depositor values('loki',7);
```

```
select * from branch;
select * from account;
select * from depositor;
select * from customer;
select * from borrower;
select * from loan;
```

```
-----
1.>select distinct (cname),count (*)
from account a,depositor d
where a.accno=d.accno
and d.accno in (select accno from account where bname='bsk')
group by cname having count (*)>=2;
```

```
-----
2.>select d.cname
from account a,depositor d,branch b
where a.bname=b.bname
and d.accno=a.accno
and b.bcity='cal'
having count(distinct b.bname)
=(select count (bname) from
branch where bcity='cal')
group by cname;
```

OR

```
select d.cname
from depositor d,account a,branch b
where ((d.accno=a.accno)and
(a.bname=b.bname) and
(b.bcity='cal'))
group by d.cname
having count(unique(a.bname))=(select count(b.bname)
from branch b
where b.bcity='cal');
```

```
-----
3.>delete from account
where bname in
(select bname
from branch
```

where bcity='hyd');