**Name : Maiyani Gautam Sureshbhai**

**Branch : CSE**

**Sub : DBMS**

**Seat No : 312025**

**Batch : Y-batch**

**Prn : 2020033800100853**

**E-mail :** [**gautammaiyani2332@gmail.com**](mailto:gautammaiyani2332@gmail.com)

**Mob. No: 9067548332**

SUPPLIER(SID, SNAME, CITY)

PART(PID, PNAME, COLOR)

CATALOG(SID, PID, COST)

1. Create the above tables with necessary constraints.

create table supplier(

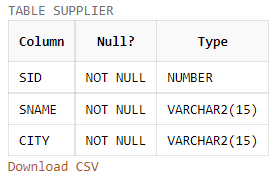
sid int primary key not null,

sname varchar(15) not null,

city varchar(15) not null

);

desc supplier



create table part(

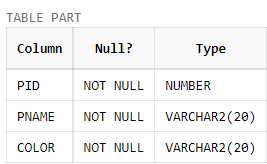
pid int primary key,

pname varchar(20) not null,

color varchar(20) not null

);

desc part



create table catalog(

sid int not null,

pid int not null,

cost int not null,

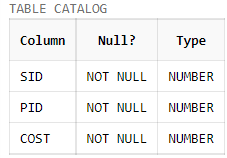
constraint fk\_sid primary key(sid,pid),

foreign key(sid) references Supplier(sid) delete on cascade,

foreign key(pid) references Part(pid) delete on cascade

);

desc catalog



* Inserting Values Into the Table..

insert into supplier(sid,sname,city) values(1,'Raj','Mumbai');

insert into supplier(sid,sname,city) values(2,'Joy','Delhi');

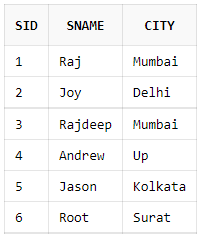
insert into supplier(sid,sname,city) values(3,'Rajdeep','Mumbai');

insert into supplier(sid,sname,city) values(4,'Andrew','Up');

insert into supplier(sid,sname,city) values(5,'Jason','Kolkata');

insert into supplier(sid,sname,city) values(6,'Root','Surat');

select \* from supplier;



insert into part(pid,pname,color) values(10,'p-1','Red');

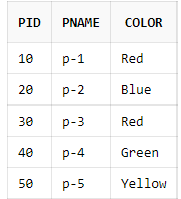
insert into part(pid,pname,color) values(20,'p-2','Blue');

insert into part(pid,pname,color) values(30,'p-3','Red');

insert into part(pid,pname,color) values(40,'p-4','Green');

insert into part(pid,pname,color) values(50,'p-5','Yellow');

select \* from part;



insert into catalog(sid,pid,cost) values(1,10,200);

insert into catalog(sid,pid,cost) values(2,10,350);

insert into catalog(sid,pid,cost) values(1,20,100);

insert into catalog(sid,pid,cost) values(3,30,1000);

insert into catalog(sid,pid,cost) values(3,10,200);

insert into catalog(sid,pid,cost) values(4,50,20);

insert into catalog(sid,pid,cost) values(2,50,600);

insert into catalog(sid,pid,cost) values(5,40,400);

insert into catalog(sid,pid,cost) values(5,20,2000);

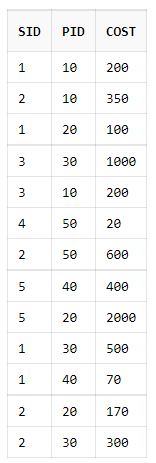
insert into catalog(sid,pid,cost) values(1,30,500);

insert into catalog(sid,pid,cost) values(1,40,70);

insert into catalog(sid,pid,cost) values(2,20,170);

insert into catalog(sid,pid,cost) values(2,30,300);

select \* from catalog;



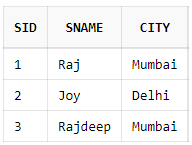
1. **Display details of suppliers who supply red colored parts.**

select \* from supplier

where sid in

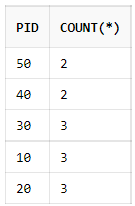
(select distinct(sid) from catalog where pid in

(select pid from part where color='Red'));



1. **Display total number of suppliers for each part.**

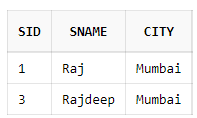
select pid,count(\*) from catalog group by pid;



1. **Display suppliers based in Mumbai and whose names have ‘j’ as the third character.**

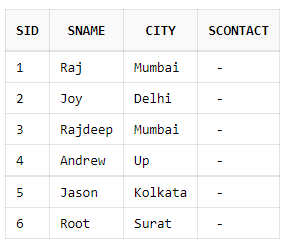
select \* from supplier

where upper(city)='MUMBAI' and upper(sname) like '\_\_J%';



1. **Add a column SCONTACT to the table SUPPLIER which always has ‘0’(zero) as the first character.**

alter table supplier add scontact number(10)constraint ck\_count check(scontact like '0%');



1. **Create a composite index on table SUPPLIER for the columns SNAME and CITY.**

create index idx on supplier (sname,city);



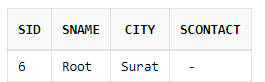
1. **Display all those suppliers who supply more than three parts.**

select sid from catalog group by sid having count(\*)>3;



1. **Display suppliers who are not supplying any part.**

select \* from supplier where sid not in(select distinct(sid)from catalog);



1. **Display the average cost of all green parts.**

select avg(cost) from catalog where pid in(select pid from part where upper(color)='GREEN');



**PART-B**

**CATEGORY(CAT\_NO, CAT\_TYPE, EXPIRY\_DT)**

**ITEM (ITEMNO, IDESC, UNIT\_OF\_MEASURE, QTY\_ON\_HAND, WEIGHT, CAT\_NO)**

1. Create the above tables with necessary constraints.

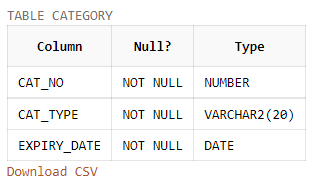
create table Category(

cat\_no int constraint pk\_cat\_no primary key not null,

cat\_type varchar(20) not null,

expiry\_date date not null);

desc Category



create table Item(

itemno int constraint pk\_item\_no primary key not null,

idesc varchar(10) not null,

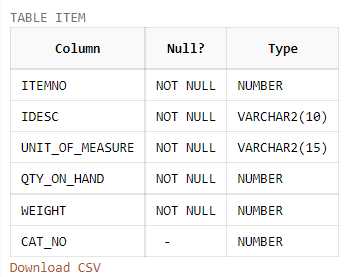
unit\_of\_measure varchar(15) not null,

qty\_on\_hand int not null,

weight int not null,

cat\_no constraint fk\_cat\_no references Category(cat\_no));

desc Item



* Inserting values into the table

insert into Category values(1,'Dairyfood','1-NOV-2023');

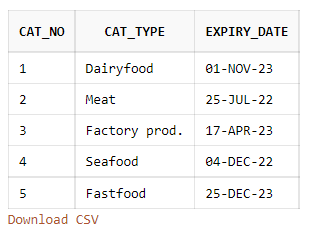
insert into Category values(2,'Meat','25-JULY-2022');

insert into Category values(3,'Factory prod.','17-APR-2023');

insert into Category values(4,'Seafood','4-DEC-2022');

insert into Category values(5,'Fastfood','25-DEC-2023');

select \* from Category;



insert into Item values(111,'Tea','Kilograms',300,50,3);

insert into Item values(112,'Fish','Kilograms',20,10,4);

insert into Item values(113,'Milk','Liters',100,50,1);

insert into Item values(114,'Noodles','Kilograms',5,10,5);

insert into Item values(115,'Prep.Meat','Kilograms',25,20,2);

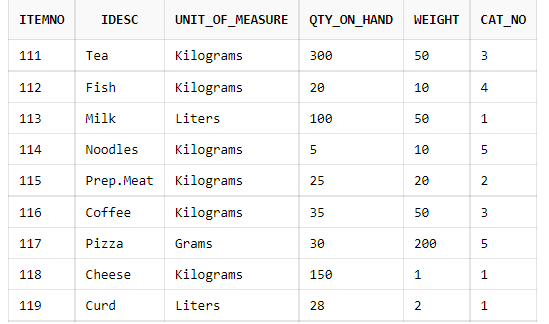
insert into Item values(116,'Coffee','Kilograms',35,50,3);

insert into Item values(117,'Pizza','Grams',30,200,5);

insert into Item values(118,'Cheese','Kilograms',150,1,1);

insert into Item values(119,'Curd','Liters',28,2,1);

select \* from Item;



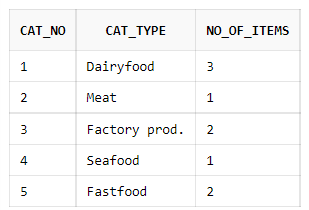
1. Add a constraint to the ITEM table which will allow the QTY\_ON\_HAND to have values between 100 and 1000 only.

alter table item add constraint ck\_qh check(qty\_on\_hand between 100 and 1000);



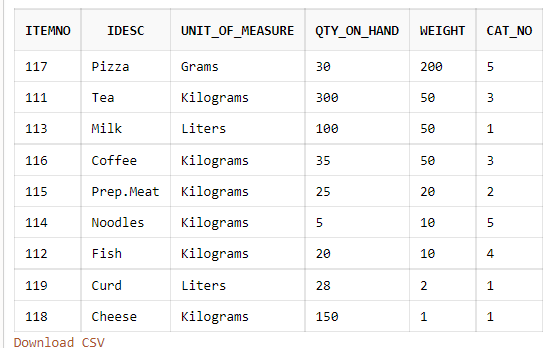
1. Display how many items are there in each category.

select i.cat\_no, c.cat\_type, count(\*) as No\_OF\_ITEMS from item i ,category c where i.cat\_no = c.cat\_no group by(i.cat\_no,c.cat\_type) order by(i.cat\_no) ;



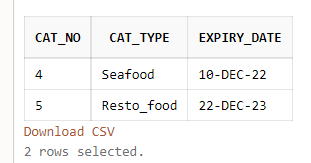
1. Display the items in descending order of their weights.

select \* from Item order by weight desc;



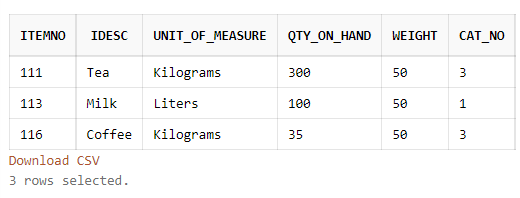
1. Display categories which expire in the month of December.

select \* from Category where (to\_char(expiry\_date,'Mon') = 'Dec');



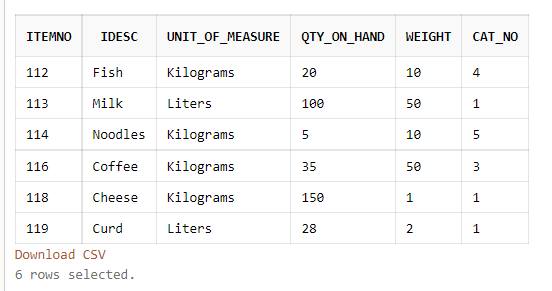
1. Display items which have the same weight as that of ITEMNO 116.

select \* from item where weight=(select weight from item where itemno =116);



1. Display items which do not have the alphabet ‘A’ or ‘T’ in their IDESC.

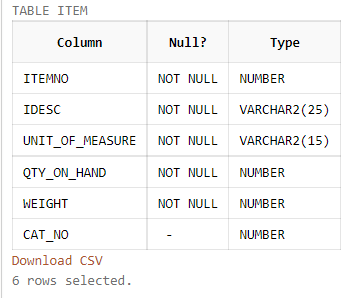
select \* from item where upper(idesc) not like '%A%' and upper(idesc) not like '%T%';



1. Increase the size of the column IDESC by making it a variable length of 25 characters.

alter table item modify idesc varchar(25) ;

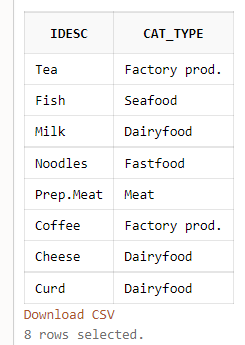
desc item



1. Display the item details and the category details of items which have a UNIT\_OF \_MEASURE as Kilogram, Meters or Litres.

select i.idesc , c.cat\_type from item i , category c

where i.cat\_no=c.cat\_no and upper(i.unit\_of\_measure)in('KILOGRAMS','LITERS','METERS');



**ASSIGNMENT-1(b)**

**PART-1**

**TEAM\_MASTER (TEAM\_ID, TEAM,NAME)**

**PLAYER(TEAM\_ID, PID, PNAME, PBIRTHDATE)**

**SCORE(TEAM\_ID, PID, RUNS\_MADE, OUT\_TYPE, B\_ID, BTEAM\_ID)**

**BALLING (TEAM\_ID, B\_ID, OVERS, MAIDEN, RUNS\_GIVEN, WICKETS\_TAKEN)**

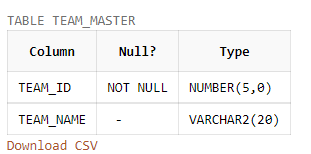
1. Write DDLs to create the above table with necessary constraints.

create table Team\_Master(

team\_id number(5) constraint pk\_team\_id primary key ,

team\_name varchar2(20)

); desc Team\_Master



create table Player(

team\_id number(5),

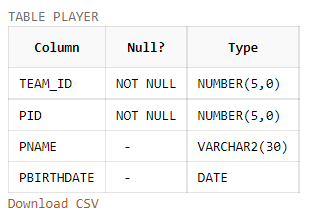
pid number(5),

pname varchar2(30),

pbirthdate date,

constraint fk\_team\_id foreign key(team\_id) references Team\_Master(team\_id) on delete cascade ,

constraint pk\_team\_id\_pid primary key(team\_id,pid)); desc Player



create table Score(

team\_id number(5),

pid number(5),

runs\_made number(5),

out\_type varchar2(20),

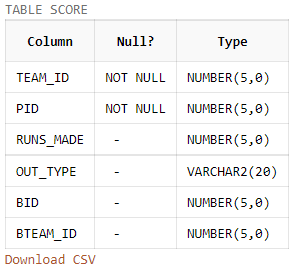
bid number(5),

bteam\_id number(5),

constraint fk1\_team\_id\_pid foreign key(team\_id,pid) references Player(team\_id,pid) ,

constraint fk2\_bteam\_id\_bid foreign key(bteam\_id,bid) references Player(team\_id,pid) ,

constraint pkS\_team\_id\_pid primary key(team\_id,pid) ); desc Score



create table Balling(

team\_id number(5),

bid number(5),

overs number(10),

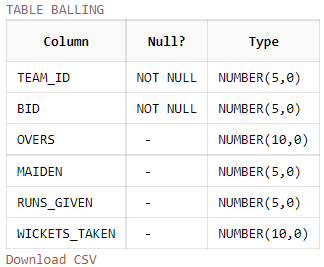
maiden number(5),

runs\_given number(5),

wickets\_taken number(10),

constraint fk\_team\_id\_bid foreign key(team\_id,bid) references Player(team\_id,pid),

constraint pk\_team\_id\_bid primary key(team\_id,bid)); desc Balling



insert into Team\_Master values(101,'India');

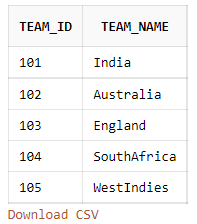
insert into Team\_Master values(102,'Australia');

insert into Team\_Master values(103,'England');

insert into Team\_Master values(104,'SouthAfrica');

insert into Team\_Master values(105,'WestIndies');

Select \* from Team\_Master;



insert into Player values(101,1,'MS Dhoni',to\_date('07-07-1981','dd-mm-yyyy')) ;

insert into Player values(101,2,'Virat Kohli',to\_date('05-11-1988','dd-mm-yyyy')) ;

insert into Player values(101,3,'Ravindra Jadeja',to\_date('06-12-1988','dd-mm-yyyy'))

insert into Player values(102,1,'Steve Smith',to\_date('01-11-1988','dd-mm-yyyy')) ;

insert into Player values(102,2,'David Warner',to\_date('08-12-1984','dd-mm-yyyy')) ;

insert into Player values(102,3,'Glen Johnsen',to\_date('21-09-1995','dd-mm-yyyy')) ;

insert into Player values(103,1,'Eion Morgan',to\_date('17-04-1982','dd-mm-yyyy')) ;

insert into Player values(103,2,'Ben Stokes',to\_date('20-02-1989','dd-mm-yyyy')) ;

insert into Player values(103,3,'Moin Ali',to\_date('18-06-1987','dd-mm-yyyy')) ;

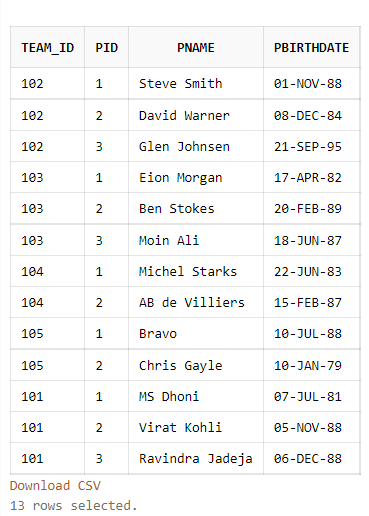
insert into Player values(104,1,'Michel Starks',to\_date('22-06-1983','dd-mm-yyyy')) ;

insert into Player values(104,2,'AB de Villiers',to\_date('15-02-1987','dd-mm-yyyy')) ;

insert into Player values(105,1,'Bravo',to\_date('10-07-1988','dd-mm-yyyy')) ;

insert into Player values(105,2,'Chris Gayle',to\_date('10-01-1979','dd-mm-yyyy')) ;

select \* from Player;



insert into Score values(101,1,101,'Bowled',2,103) ;

insert into Score values(101,2,79,'Caught',1,105) ;

insert into Score values(101,3,34,'Caught',2,103) ;

insert into Score values(102,1,36,'Stumped',3,103) ;

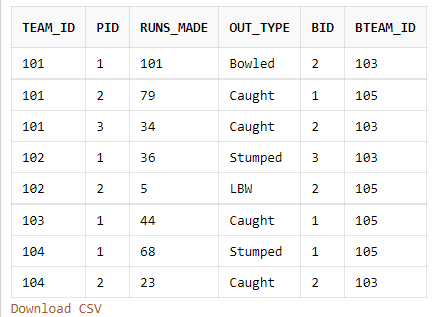
insert into Score values(102,2,5,'LBW',2,105) ;

insert into Score values(103,1,44,'Caught',1,105) ;

insert into Score values(104,1,68,'Stumped',1,105) ;

insert into Score values(104,2,23,'Caught',2,103) ;

Select \*from Score;



insert into Balling values(103,2,4,0,28,2);

insert into Balling values(105,1,4,0,22,2);

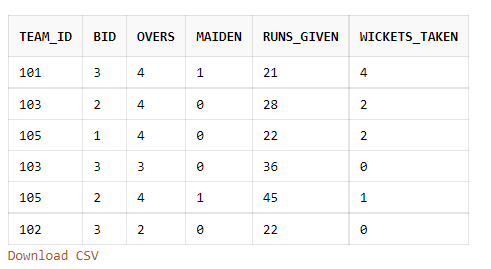
insert into Balling values(103,3,3,0,36,0);

insert into Balling values(105,2,4,1,45,1);

insert into Balling values(102,3,2,0,22,0);

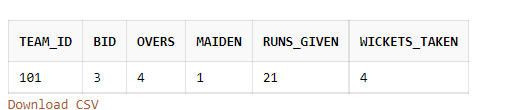
insert into Balling values(101,3,4,1,21,4);

Select \*from Balling;



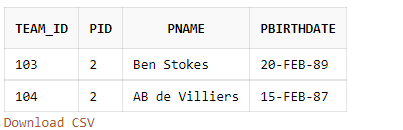
1. Display details about the bowler who took the maximum wickets.

select \* from Balling where wickets\_taken = (select max(wickets\_taken) from Balling);



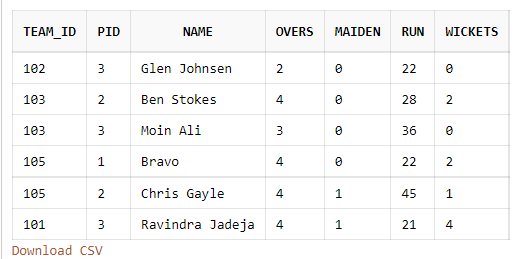
1. Which players are born in the month of February and have scored more than 25 runs?

select \* from Player where pid in (select pid from Score where runs\_made > 25) and (to\_char(pbirthdate,'Mon') = 'Feb') ;



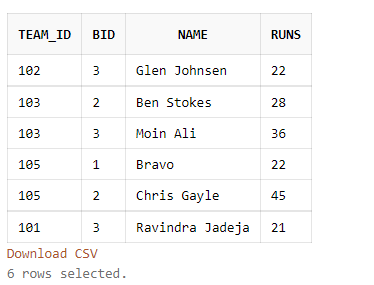
1. Display details about players who bowled at the most three maiden overs.

select p.team\_id,pid,pname as Name,overs,maiden,runs\_given as Run,wickets\_taken as Wickets from Player p , Balling b where p.pid = b.bid and p.team\_id = b.team\_id and maiden <= 3 ;



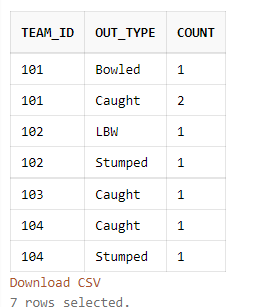
1. Display the total runs given by each bowler.

select p.team\_id,bid,pname as Name,runs\_given as Runs from Player p , Balling b where p.pid = b.bid and p.team\_id = b.team\_id ;



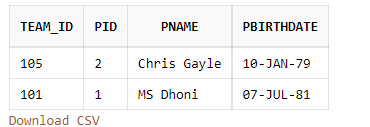
1. Display team wise how many players got out with lbw, stumping, catch etc.

select team\_id,out\_type,count(\*) as count from Score group by (team\_id,out\_type) order by team\_id ;



1. Display details about players who are more than 40 years old.

select \* from Player where round(sysdate-pbirthdate) > 40\*365;



1. Display the bowlers who are also batsmen.

select p.team\_id, p.pid, pname as “ALL ROUNDERS” from Player p,Score s , Balling b where s.pid=b.bid and s.team\_id=b.team\_id and p.pid=s.pid and p.team\_id=s.team\_id ;



**PART-2**

**CANDIDATE (CID, CNAME, CADDRESS, CBIRTH\_DT)**

**TEST (TID, TNAME, TOT\_MRKS, PASS\_MKS)**

**TEST\_CENTRE (TCID, LOCATION, MGR, CAPACITY)**

**TEST\_TAKEN (CID, TID, TCID, TEST\_DT, SCORE)**

1. Design the above tables with appropriate constraints.

create table Candidate(

cid number(5) constraint pk\_cid primary key,

cname varchar2(25),

caddress varchar2(30),

cbirth\_dt date );

create table Test(

tid number(5) constraint pk\_tid primary key,

tname varchar2(25),

tot\_marks number(5),

pass\_marks number(5) constraint ck\_pm check(pass\_marks < tot\_marks));

create table Test\_Centre(

tcid number(5) constraint pk\_tcid primary key,

location varchar2(25),

mgr varchar2(25),

capacity number(5) );

create table Test\_Taken(

cid number(5),

tid number(5),

tcid number(5),

test\_dt date,

score number(5),

constraint fk\_cid foreign key(cid) references candidate(cid),

constraint fk\_tid foreign key(tid) references test(tid),

constraint fk\_tcid foreign key(tcid) references test\_centre(tcid),

constraint pk\_cid\_tid\_tcid primary key(cid,tid,tcid));

* INSERTING VALUE IN TABLE

insert into Candidate values(1,'Gautam Maiyani','Kirti Stambh','14-Nov-2002');

insert into Candidate values(2,'Raval Krish ','Akota','07-JUN-2001');

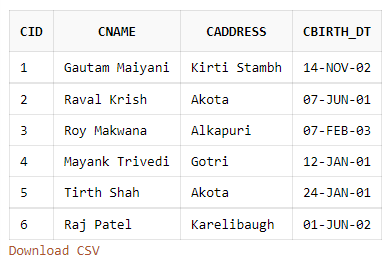
insert into Candidate values(3,'Roy Makwana','Alkapuri','07-FEB-2003');

insert into Candidate values(4,'Mayank Trivedi','Gotri','12-JAN-2001');

insert into Candidate values(5,'Tirth Shah','Akota','24-JAN-2001');

insert into Candidate values(6,'Raj Patel','Karelibaugh','01-JUN-2002');

select \* from Candidate;



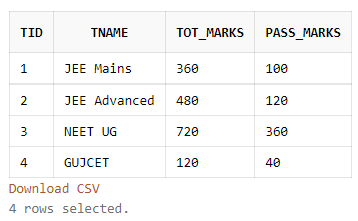
insert into test values(1,'JEE Mains',360,100) ;

insert into test values(2,'JEE Advanced',480,120) ;

insert into test values(3,'NEET UG',720,360) ;

insert into test values(4,'GUJCET',120,40) ;

SELECT\* from test;



insert into Test\_Centre values(1,'Gotri','Raj Shah',200) ;

insert into Test\_Centre values(2,'Alkapuri','Nilesh Patel',100) ;

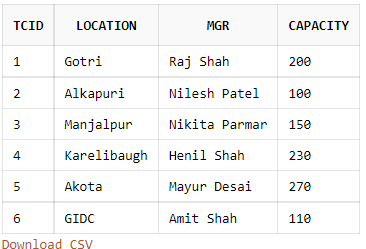
insert into Test\_Centre values(3,'Manjalpur','Nikita Parmar',150) ;

insert into Test\_Centre values(4,'Karelibaugh','Henil Shah',230) ;

insert into Test\_Centre values(5,'Akota','Mayur Desai',270) ;

insert into Test\_Centre values(6,'GIDC','Amit Shah',110) ;

SELECT\* from Test\_Centre;



insert into Test\_Taken values(1,1,4,'15-JUNE-2020',220);

insert into Test\_Taken values(1,2,4,'20-MAY-2020',155);

insert into Test\_Taken values(1,4,1,'20-APR-2020',118);

insert into Test\_Taken values(2,3,2,'25-JUNE-2020',580);

insert into Test\_Taken values(3,3,2,'20-MAY-2020',490);

insert into Test\_Taken values(4,1,1,'20-APR-2020',120);

insert into Test\_Taken values(4,2,4,'15-MAY-2020',25);

insert into Test\_Taken values(4,4,3,'25-APR-2020',99);

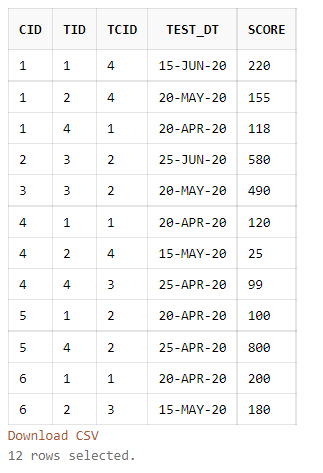
insert into Test\_Taken values(5,1,2,'20-APR-2020',100);

insert into Test\_Taken values(5,4,2,'25-APR-2020',70);

insert into Test\_Taken values(6,1,1,'20-APR-2020',200);

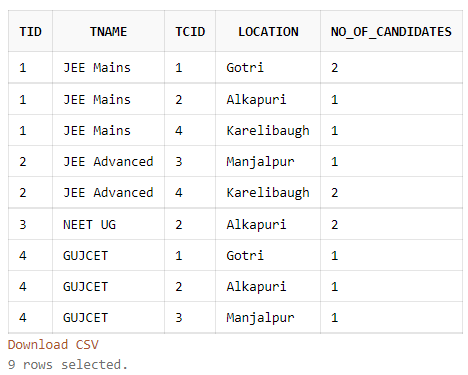
insert into Test\_Taken values(6,2,3,'15-MAY-2020',180);

SELECT\* from Test\_Taken;



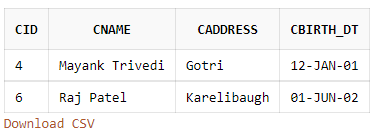
1. Display the number of candidates who have appeared for each test at each test centre.

select tt.tid, tname, tt.tcid, location, count(\*) as No\_Of\_Candidates from Test\_Taken tt , Test t,Test\_Centre tc where tt.tid = t.tid and tt.tcid = tc.tcid group by(tt.tid,tt.tcid,tname,location) order by(tid);



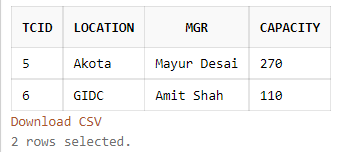
1. Display candidate details for those candidates who have scored as many marks as Vineet for tests conducted at Manjalpur.

select c.\* from Candidate c ,Test\_Centre tc, Test\_Taken tt where c.cid = tt.cid and tc.tcid = tt.tcid and upper(location) = 'MANJALPUR' and score in (select score from test\_taken tt1 ,Candidate c1 where tt1.cid = c.cid and upper(cname) = 'RAJ PATEL') ;



1. Display details about test centers where no tests have been conducted.

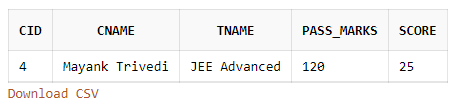
select \* from Test\_Centre where tcid not in (select distinct(tcid) from Test\_Taken ) order by tcid ;



1. Display details of candidates who have failed**.**

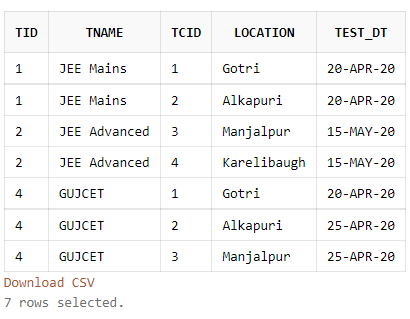
select c.cid, c.cname, t.tname, t.pass\_marks, tt.score from Candidate c , Test\_Taken tt ,Test t

where c.cid = tt.cid and t.tid = tt.tid and score < pass\_marks ;



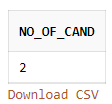
1. For tests, which have been conducted between 20-04-2020 and 15-05-2020, show details of the tests as well as the test centers.

select tt.tid, t.tname, tt.tcid, tc.location, test\_dt from Test\_Centre tc, Test\_Taken tt, Test t where tc.tcid = tt.tcid and t.tid = tt.tid and test\_dt between '20-APR-2020' and '15-MAY-2020' group by (t.tname, tt.tid, tt.tcid, tc.location, test\_dt) order by tt.tid,tt.tcid ;



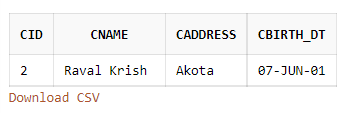
1. How many candidates appeared in the ‘JEE MAINS’ test at Surat in the month of April?

select count(\*) as No\_Of\_Cand from Test\_Taken where to\_char(test\_dt,'Mon')='Apr' and tid in (select tid from Test where upper(tname) = 'JEE MAINS') and tcid = (select tcid from Test\_Centre where upper(location) = 'GOTRI') group by tid ;



1. Display details about candidates who appeared for tests in the same month as the month in which they were born.

select \* from Candidate where cid in (select c.cid from Candidate c ,Test\_Taken tt where tt.cid = c.cid and to\_char(cbirth\_dt,'Mon') = to\_char(test\_dt,'Mon')) ;



1. Display the details about candidates who have scored the highest in each test, test centre wise.

select \* from candidate where cid in (select cid from test\_taken where (score,cid,tid,tcid) in (select max(score),cid,tid,tcid from test\_taken group by (score,cid,tid,tcid))) ;

