Q1.Write a SQL query to fetch the student name who has appeared in only 1 subject.

Create table Student(Studid number,NAME varchar2(10),Subject varchar2(20),marks number);

insert into student values(1,'A','Phy','90');

insert into student values(1,'A','Che','95');

insert into student values(2,'B','Phy','80');

insert into student values(2,'B','Che','85');

insert into student values(3,'C','Phy','90');

insert into student values(4,'D','Phy','75');

insert into student values(4,'D','Che','90');

Q2.Write a SQL query to fetch the student name who has appeared in more than 1 subject.

Create table Student(Studid number,NAME varchar2(10),Subject varchar2(20),marks number);

insert into student values(1,'A','Phy','90');

insert into student values(1,'A','Che','95');

insert into student values(2,'B','Phy','80');

insert into student values(2,'B','Che','85');

insert into student values(3,'C','Phy','90');

insert into student values(4,'D','Phy','75');

insert into student values(4,'D','Che','90');

Q3.Write a SQL query to fetch the student name who has appeared in more than 1 subject and have scored marks greater than 85

Create table Student(Studid number,NAME varchar2(10),Subject varchar2(20),marks number);

insert into student values(1,'A','Phy','90');

insert into student values(1,'A','Che','95');

insert into student values(2,'B','Phy','80');

insert into student values(2,'B','Che','85');

insert into student values(3,'C','Phy','90');

insert into student values(4,'D','Phy','75');

insert into student values(4,'D','Che','90');

Q4.Write a query and show the details of students who have not failed in any subject.

Create table student(id number,name varchar2(10),subject varchar2(10),result varchar2(10));

insert into student values(1,'A','Maths','Pass');

insert into student values(1,'A','Phy','Pass');

insert into student values(1,'A','Chem','Fail');

insert into student values(2,'B','Maths','Pass');

insert into student values(2,'B','Phy','Fail');

insert into student values(2,'B','Chem','Fail');

insert into student values(3,'C','Maths','Pass');

insert into student values(3,'C','Phy','Pass');

insert into student values(3,'C','Chem','Pass');

Q5. Write an SQL query to fetch the below details:(Ignore subquestion-e,f,i,j)

a)Total profit of shopping store

b)Total profit deptwise

c)Total Profit monthwise

d)Total Profit location wise

e)Dept where profit has increased month over month

f)Dept where profit has decreased month over month

g)Location which is giving the least profit

h)Location which is giving the maximum profit

i)Dept and location where profit has continously increased month over month

j)Dept and Location where profit has continously decreased month over month

create table shopping(deptname varchar(50),location varchar(50),month varchar(50),profit int);

insert into shopping values('clothing','delhi','jan',200);

insert into shopping values('clothing','delhi','feb',300);

insert into shopping values('clothing','delhi','mar',500);

insert into shopping values('clothing','mumbai','jan',300);

insert into shopping values('clothing','mumbai','feb',200);

insert into shopping values('clothing','mumbai','mar',150);

insert into shopping values('clothing','chennai','jan',400);

insert into shopping values('clothing','chennai','feb',400);

insert into shopping values('clothing','chennai','mar',400);

insert into shopping values('sports','delhi','jan',200);

insert into shopping values('sports','delhi','feb',150);

insert into shopping values('sports','delhi','mar',100);

insert into shopping values('sports','mumbai','jan',300);

insert into shopping values('sports','mumbai','feb',400);

insert into shopping values('sports','mumbai','mar',500);

insert into shopping values('sports','chennai','jan',400);

insert into shopping values('sports','chennai','feb',500);

insert into shopping values('sports','chennai','mar',300);

insert into shopping values('appliances','delhi','jan',200);

insert into shopping values('appliances','delhi','feb',350);

insert into shopping values('appliances','delhi','mar',100);

insert into shopping values('appliances','mumbai','jan',350);

insert into shopping values('appliances','mumbai','feb',400);

insert into shopping values('appliances','mumbai','mar',500);

insert into shopping values('appliances','chennai','jan',400);

insert into shopping values('appliances','chennai','feb',500);

insert into shopping values('appliances','chennai','mar',200);

Q6. Write an SQL query to find the topper in each semester.

CREATE TABLE STUD(ID NUMBER,NAME VARCHAR2(10),SEMESTER NUMBER,SUBJECT VARCHAR2(10),MARKS NUMBER);

INSERT INTO STUD VALUES(1,'A',1,'PHYSICS',100);

INSERT INTO STUD VALUES(1,'A',2,'PHYSICS',150);

INSERT INTO STUD VALUES(1,'A',3,'PHYSICS',200);

INSERT INTO STUD VALUES(1,'A',4,'PHYSICS',250);

INSERT INTO STUD VALUES(1,'A',1,'CHEMISTRY',50);

INSERT INTO STUD VALUES(1,'A',2,'CHEMISTRY',250);

INSERT INTO STUD VALUES(1,'A',3,'CHEMISTRY',200);

INSERT INTO STUD VALUES(1,'A',4,'CHEMISTRY',350);

INSERT INTO STUD VALUES(2,'B',1,'PHYSICS',150);

INSERT INTO STUD VALUES(2,'B',2,'PHYSICS',250);

INSERT INTO STUD VALUES(2,'B',3,'PHYSICS',100);

INSERT INTO STUD VALUES(2,'B',4,'PHYSICS',200);

INSERT INTO STUD VALUES(2,'B',1,'CHEMISTRY',150);

INSERT INTO STUD VALUES(2,'B',2,'CHEMISTRY',150);

INSERT INTO STUD VALUES(2,'B',3,'CHEMISTRY',250);

INSERT INTO STUD VALUES(2,'B',4,'CHEMISTRY',300);

Q7. Write an SQL query to find the topper in each subject.

CREATE TABLE STUD(ID NUMBER,NAME VARCHAR2(10),SEMESTER NUMBER,SUBJECT VARCHAR2(10),MARKS NUMBER);

INSERT INTO STUD VALUES(1,'A',1,'PHYSICS',100);

INSERT INTO STUD VALUES(1,'A',2,'PHYSICS',150);

INSERT INTO STUD VALUES(1,'A',3,'PHYSICS',200);

INSERT INTO STUD VALUES(1,'A',4,'PHYSICS',250);

INSERT INTO STUD VALUES(1,'A',1,'CHEMISTRY',50);

INSERT INTO STUD VALUES(1,'A',2,'CHEMISTRY',250);

INSERT INTO STUD VALUES(1,'A',3,'CHEMISTRY',200);

INSERT INTO STUD VALUES(1,'A',4,'CHEMISTRY',350);

INSERT INTO STUD VALUES(2,'B',1,'PHYSICS',150);

INSERT INTO STUD VALUES(2,'B',2,'PHYSICS',250);

INSERT INTO STUD VALUES(2,'B',3,'PHYSICS',100);

INSERT INTO STUD VALUES(2,'B',4,'PHYSICS',200);

INSERT INTO STUD VALUES(2,'B',1,'CHEMISTRY',150);

INSERT INTO STUD VALUES(2,'B',2,'CHEMISTRY',150);

INSERT INTO STUD VALUES(2,'B',3,'CHEMISTRY',250);

INSERT INTO STUD VALUES(2,'B',4,'CHEMISTRY',300);

Q8.Write an SQL query to find the employee who is earning second highest salary.

create table HR(empid number,empname varchar2(50),dept varchar2(50),salary number);

insert into HR values(1,'A','HR',100);

insert into HR values(2,'B','HR',100);

insert into HR values(3,'C','HR',90);

insert into HR values(4,'D','TECH',250);

insert into HR values(5,'E','TECH',200);

insert into HR values(6,'F','TECH',190);

Q9.Write a SQL query to fetch the details of employee whose salary is least deptwise.

create table dept (dept\_id number,deptname varchar2(10),empname varchar2(10),salary number);

insert into dept values(1,'HR','A',100);

insert into dept values(1,'HR','B',200);

insert into dept values(1,'HR','C',300);

insert into dept values(1,'HR','X',NULL);

insert into dept values(2,'SALES','D',400);

insert into dept values(2,'SALES','E',500);

insert into dept values(2,'SALES','F',600);

insert into dept values(2,'SALES','Y',NULL);

insert into dept values(3,'TECH','G',700);

insert into dept values(3,'TECH','H',800);

insert into dept values(3,'TECH','I',900);

insert into dept values(3,'TECH','Z',NULL);

Q10.Write an SQL query to fetch the location with highest number of cases.

create table covid(location varchar2(50),days varchar2(10),cases number);

insert into covid values('DELHI','MON',100);

insert into covid values('DELHI','TUE',200);

insert into covid values('DELHI','WED',300);

insert into covid values('MUMBAI','MON',100);

insert into covid values('MUMBAI','TUE',100);

insert into covid values('MUMBAI','WED',300);

insert into covid values('CHENNAI','MON',100);

insert into covid values('CHENNAI','TUE',200);

insert into covid values('CHENNAI','WED',201);

Q11.Write a SQL query to find the overall topper.

Create table Student(Studid number,NAME varchar2(10),Subject varchar2(20),marks number);

insert into student values(1,'A','Phy','90');

insert into student values(1,'A','Che','95');

insert into student values(2,'B','Phy','80');

insert into student values(2,'B','Che','85');

insert into student values(3,'C','Phy','90');

insert into student values(4,'D','Phy','75');

insert into student values(4,'D','Che','90');

Q12. Write an SQL query to find the

a)Total txnamount

b)Total txnamount per month

c)Total txnamount per dept

d)Total txnamount per dept per month

e)Max txnamount per dept

f)Max txnamount per month

g)Min txnamount per dept

h)Max txnamount per month

i)avg txnamount of entire transaction

j)avg txnamount dept wise

i)avg txnamount month wise

j)dept with highest txnamount.

k)dept with least txnamount.

create table amazon(txnmonth varchar2(50),dept varchar2(50),txnamount int);

insert into amazon values('Jan','Grocery',100);

insert into amazon values('Feb','Grocery',300);

insert into amazon values('Mar','Grocery',200);

insert into amazon values('Jan','Clothes',700);

insert into amazon values('Feb','Clothes',500);

insert into amazon values('Mar','Clothes',200);

insert into amazon values('Jan','Sports',300);

insert into amazon values('Feb','Sports',400);

insert into amazon values('Mar','Sports',200);

Q13.Write a SQL query to find the details of employee whose salary is greater than the average salary of all the employees..

Create table employee(empid number,empname varchar2(10),salary number);

insert into employee values(1,'A',100);

insert into employee values(2,'B',200);

insert into employee values(3,'C',100);

insert into employee values(4,'D',300);

insert into employee values(5,'E',100);

insert into employee values(6,'F',400);