/\*

Tables:

STUDIES, SOFTWARE and PROGRAMMER

\*/

select \* from studies

select \* from software

select \* from programmer

--1. Find out the selling cost average for packages developed in Pascal.

select DEVELOPIN,AVG(SCOST) from Software

where DEVELOPIN='PASCAL'

group by DEVELOPIN

--2. Display the names and ages of all programmers.

select pname,(year(getdate())-year(dob)) as age from Programmer

--3. Display the names of those who have done the DAP Course.

select \* from Studies

where COURSE='DAP'

--4. Display the names and date of birth of all programmers born in January.

select Programmer.PNAME,Programmer.DOB from Programmer

where month(DOB)=1

--5. What is the highest number of copies sold by a package?

select title,sum(sold) as num\_copies from Software

group by title

order by sum(sold) desc

--6. Display lowest course fee.

select distinct(COURSE),course\_fee from Studies

order by course\_fee asc

--7. How many programmers have done the PGDCA Course?

select COURSE,COUNT(COURSE) AS COUNT from Studies

where COURSE='PGDCA'

GROUP BY COURSE

--8. How much revenue has been earned through sales of packages developed in C?

SELECT Software.DEVELOPIN,sum(scost \* sold) as revenue FROM Software

where DEVELOPIN='c'

group by DEVELOPIN

--9. Display the details of the software developed by Ramesh.

select \* from Software

where pname='ramesh'

--10. How many programmers studied at Sabhari

select \* from Studies

where INSTITUTE='sabhari'

--11. Display details of packages whose sales crossed the 2000 mark.

select TITLE,DEVELOPIN,(scost\*sold)-(DCOST) as sales from Software

where (scost\*sold)-(DCOST)>2000

--12. Display the details of packages for which development costs have been recovered.

select TITLE,DEVELOPIN,(SCOST\*SOLD)-DCOST AS PROFIT from Software

WHERE (SCOST\*SOLD)-DCOST>0

--LOSSED PRODUCTS

select TITLE,DEVELOPIN,(SCOST\*SOLD)-DCOST AS LOSS from Software

WHERE (SCOST\*SOLD)-DCOST<0

--13. What is the cost of the costliest software development in Basic?

SELECT TITLE,DEVELOPIN,DCOST FROM Software

WHERE DEVELOPIN='BASIC'

ORDER BY DCOST DESC

--14. How many packages have been developed in dBase?

SELECT DEVELOPIN,COUNT(\*) AS COUNT FROM Software

WHERE DEVELOPIN='DBASE'

GROUP BY DEVELOPIN

--15. How many programmers studied in Pragathi?

SELECT \* FROM STUDIES

WHERE INSTITUTE='PRAGATHI'

--16. How many programmers paid 5,000 to 10,000 for their course?

select count(\*) as course\_fee from Studies

where COURSE\_FEE between 5000 and 10000

--17. What is the average course fee?

select avg(COURSE\_FEE) as avg\_course\_fee from Studies

--18. Display the details of the programmers knowing C.

select \* from Programmer

where PROF1='C' or PROF2 ='c'

--19. How many programmers know either COBOL or Pascal?

select \* from Programmer

where PROF1='COBOL' or PROF1='Pascal'or PROF2='Pascal' or PROF2='COBOL'

--20. How many programmers don’t know Pascal and C?

select count(\*) as counts from Programmer

where PROF1 not in ('Pascal','C') and PROF2 not in ('Pascal','C')

--21. How old is the oldest male programmer?

select dense\_rank() over (order by datediff(yy,dob,getdate()) desc) as age\_rank,

pname,datediff(yy,dob,getdate()) as age,

datediff(YY,doj,getdate()) as workexp

from Programmer

order by datediff(YY,dob,getdate()) desc

--Juliana is oldest programer

--22. What is the average age of female programmers?

ALTER TABLE Programmer

ADD age AS DATEDIFF(YY, dob, GETDATE());

select avg(age) as female\_avg\_age from Programmer

where GENDER='f'

--23. Calculate the experience in years for each programmer and display with

--their names in descending order.

select pname,DATEDIFF(yy,doj,getdate()) as experience from Programmer

order by pname desc

select pname,DATEDIFF(yy,doj,getdate()) as experience from Programmer

order by DATEDIFF(yy,doj,getdate()) desc

--24. Who are the programmers who celebrate their birthdays during the

--current month?

select pname,dob from Programmer

where month(dob)=month(getdate())

--25. How many female programmers are there?

select count(\*) female\_programers from Programmer

where GENDER='f'

-- 26. What are the languages studied by male programmers?

select PNAME,PROF1,PROF2 from Programmer

where gender='M'

-- alternative

select distinct(PROF1) from Programmer

where GENDER='m'

union

select distinct(prof2) from Programmer

where GENDER='m'

--27. What is the average salary

select avg(salary) as average\_salary from Programmer

--28. How many people draw a salary between 2000 to 4000?

select count(\*) as count\_of\_people\_salary\_2000\_to\_4000 from Programmer

where salary between 2000 and 4000

--29. Display the details of those who don’t know Clipper, COBOL or Pascal.

select \* from programmer

where prof1 not in ('Clipper','COBOL','Pascal')

and prof2 not in ('Clipper','COBOL','Pascal')

--30. Display the cost of packages developed by each programmer.

select pname,sum(dcost) total\_development\_cost from software

group by pname

order by sum(dcost) asc

--31. Display the sales value of the packages developed by each programmer.

select Pname,sum(scost\*sold) Sales\_by\_Developer from Software

group by PNAME

order by sum(SCOST\*SOLD) desc

--32. Display the number of packages sold by each programmer.

select PNAME,count(\*) as Software\_count from Software

group by PNAME

order by count(\*) desc

--33. Display the sales cost of the packages developed by each programmer language wise.

select DEVELOPIN,sum(SCOST\*SOLD) as Total\_Revenue from software

group by DEVELOPIN

order by Total\_Revenue desc

--34. Display each language name with the average development cost,

--average selling cost and average price per copy.

select DEVELOPIN,

avg(DCOST) as AVG\_DCOST,

sum(SCOST\*SOLD)/sum(sold) as AVG\_SCOST,

avg(SCOST) AS AVG\_PRICE\_PER\_COPY

from Software

GROUP BY DEVELOPIN

--alternative

select DEVELOPIN,

avg(DCOST) as AVG\_DCOST,

avg(SCOST\*SOLD) as AVG\_SCOST,

avg(SCOST) AS AVG\_PRICE\_PER\_COPY

from Software

GROUP BY DEVELOPIN

--35. Display each programmer’s name and the costliest and cheapest

--packages developed by him or her.

select PNAME,iif(max(dcost),max(dcost),min(dcost)) from Software

group by PNAME

WITH CostliestPackages AS (

SELECT pname, TITLE, DCOST,

ROW\_NUMBER() OVER (PARTITION BY pname ORDER BY DCOST DESC) AS rank\_max

FROM Software

),

CheapestPackages AS (

SELECT pname, TITLE, DCOST,

ROW\_NUMBER() OVER (PARTITION BY pname ORDER BY DCOST ASC) AS rank\_min

FROM Software

)

SELECT C.pname,

C.title AS COSTLIEST\_PACKAGE,

H.title AS CHEAPEST\_PACKAGE

FROM CostliestPackages C

JOIN CheapestPackages H ON C.pname = H.pname

WHERE C.rank\_max = 1 AND H.rank\_min = 1;

--36. Display each institute’s name with the number of courses and the average cost per course.

select INSTITUTE,count(course) as NUMBER\_OF\_COURSE,avg(course\_fee) AS AVERAGE\_COURSE from Studies

group by INSTITUTE

--37. Display each institute’s name with the number of students.

SELECT INSTITUTE,COUNT(PNAME) AS NUMBER\_OF\_STUDENTS FROM Studies

GROUP BY INSTITUTE

--38. Display names of male and female programmers along with their gender.

SELECT PNAME,IIF(GENDER='M','Male','Female') as gender FROM Programmer

--39. Display the name of programmers and their packages.

select p.PNAME,s.TITLE as Packages from Programmer as p

inner join Software as s

on p.PNAME=s.PNAME

--40. Display the number of packages in each language except C and C++.

select DEVELOPIN,count(title) as NO\_OF\_PACKAGES from Software

WHERE DEVELOPIN NOT IN ('CPP','C')

GROUP BY DEVELOPIN

--41. Display the number of packages in each language for which development cost is less than 1000.

SELECT DEVELOPIN,COUNT(TITLE) AS NO\_OF\_PACKAGES FROM SOFTWARE

WHERE DCOST<1000

GROUP BY DEVELOPIN

--42. Display the average difference between SCOST and DCOST for each package.

SELECT TITLE,AVG(DCOST-SCOST) FROM SOFTWARE

GROUP BY TITLE

--43. Display the total SCOST, DCOST and the amount to be recovered for

--each programmer whose cost has not yet been recovered.

SELECT

PNAME,TITLE,

SCOST\*SOLD AS TOTAL\_SELLING\_COST,

DCOST,

((SCOST\*SOLD)-DCOST) AS RECOVERED\_MONEY

FROM Software

WHERE ((SCOST\*SOLD)-DCOST)<0

--44. Display the highest, lowest and average salaries for those earning more than 2000.

SELECT MAX(SALARY) AS HIGHEST\_SALARY,

MIN(SALARY) AS LOWEST\_SALARY,

AVG(SALARY) AS AVERAGE\_SALARY FROM Programmer

WHERE SALARY>2000

--45. Who is the highest paid C programmer

SELECT TOP 1 PNAME,SALARY FROM Programmer

WHERE PROF1='C' OR PROF2='C'

ORDER BY SALARY DESC

--46. Who is the highest paid female COBOL programmer?

SELECT PNAME,SALARY FROM Programmer

WHERE GENDER='F' AND (PROF1='COBOL' OR PROF2='COBOL')

ORDER BY SALARY DESC

--47. Display the names of the highest paid programmers for each language.

WITH MaxSalaries AS (

SELECT S.DEVELOPIN, MAX(P.SALARY) AS HIGHEST\_SALARY

FROM Programmer AS P

INNER JOIN Software AS S

ON P.PNAME = S.PNAME

GROUP BY S.DEVELOPIN

)

SELECT P.PNAME, S.DEVELOPIN, P.SALARY AS HIGHEST\_SALARY

FROM Programmer AS P

INNER JOIN Software AS S

ON P.PNAME = S.PNAME

INNER JOIN MaxSalaries MS

ON S.DEVELOPIN = MS.DEVELOPIN AND P.SALARY = MS.HIGHEST\_SALARY;

--48. Who is the least experienced programmer?

select TOP 1 PNAME,DATEDIFF(yy,DOJ,getdate()) as EXPIRIENCE from Programmer

ORDER BY EXPIRIENCE

--49. Who is the most experienced male programmer knowing PASCAL?

SELECT \* FROM Programmer

--50. Which language is known by only one programmer?

select \* from Programmer

where PROF1 in ('FOXPRO','CPP','ORACLE') or PROF2 in ('FOXPRO','CPP','ORACLE')

select prof,sum(no\_of\_programmers) as total\_programmers

from (

--count for prof1

select prof1 as prof, count(prof1) as no\_of\_programmers

from Programmer

group by PROF1

union all

--count for prof2

select prof2 as prof, count(prof2) as no\_of\_programmers

from programmer

group by prof2

) as combined

group by prof

having sum(no\_of\_programmers)=1

--51. Who is the above programmer referred in 50?

select prof,sum(no\_of\_programmers) as total\_programmers

from (

--count for prof1

select prof1 as prof, count(prof1) as no\_of\_programmers

from Programmer

group by PROF1

union all

--count for prof2

select prof2 as prof, count(prof2) as no\_of\_programmers

from programmer

group by prof2

) as combined

group by prof

having sum(no\_of\_programmers)=1

--solution:

select \* from Programmer

where PROF1 in ('FOXPRO','CPP','ORACLE') or PROF2 in ('FOXPRO','CPP','ORACLE')

--52. Who is the youngest programmer knowing dBase?

select top 1 \* from Programmer

where PROF1='dbase' or prof2='dbase'

order by age asc

--53. Which female programmer earning more than 3000 does not know C, C++, Oracle or dBase?

select \* from Programmer

where (GENDER='f' and salary >3000) and

(PROF1 not in ('c','cpp','oracle','dBase') and PROF2 not in ('c','cpp','oracle','dBase'))

--54. Which institute has the most number of students?

select top 1 INSTITUTE,count(institute) no\_of\_students from Studies

group by INSTITUTE

order by no\_of\_students desc

--55. What is the costliest course?

select top 1 \* from Studies

order by COURSE\_FEE desc

--56. Which course has been done by the most number of students?

select top 1 with ties course,

(count(course)) as num\_of\_students from Studies

group by course

order by num\_of\_students desc

--57. Which institute conducts the costliest course?

select top 1 \* from Studies

order by COURSE\_FEE desc

--58. Display the name of the institute and the course which has below

--average course fee.

select INSTITUTE,course,COURSE\_FEE from Studies

where COURSE\_FEE<(select avg(COURSE\_FEE) from Studies)

--59. Display the names of the courses whose fees are within 1000 (+ or -) of the average fee.

with average\_fees as

(select avg(course\_fee) as avgfee from Studies)

select course,COURSE\_FEE from Studies

cross join average\_fees

where COURSE\_FEE between (avgfee-1000) and (avgfee+1000)

--60. Which package has the highest development cost?

select top 1 with ties \* from Software

order by DCOST desc

--61. Which course has below average number of students?

create view students\_countbyCourse

as select COURSE,count(course) as count\_of\_students from Studies group by COURSE

select course,count\_of\_students from students\_countbyCourse

where count\_of\_students<(select avg(count\_of\_students) from students\_countbyCourse)

--62. Which package has the lowest selling cost?

select top 1 with ties \* from Software

order by SCOST asc

--63. Who developed the package that has sold the least number of copies?

select top 1 with ties \* from Software

order by sold asc

--64. Which language has been used to develop the package which has the

--highest sales amount?

select TITLE,DEVELOPIN,(SCOST\*SOLD) as total\_sales from Software

where SCOST\*sold=(select max(scost\*sold) from Software)

--65. How many copies of the package that has the least difference between development and selling cost were sold?

select TITLE,DEVELOPIN,SOLD as Copies,(SCOST-DCOST) as Profit from software

order by abs(SCOST-DCOST) asc

--66. Which is the costliest package developed in Pascal?

select top 1 with ties \* from Software

where DEVELOPIN='pascal'

order by DCOST desc

--67. Which language was used to develop the most number of packages?

select top 1 with ties DEVELOPIN,COUNT(DEVELOPIN) as num\_of\_packages from Software

group by DEVELOPIN

order by num\_of\_packages desc

--68. Which programmer has developed the highest number of packages?

select top 1 with ties PNAME,COUNT(title) num\_of\_packages from Software

group by PNAME

order by num\_of\_packages desc

--69. Who is the author of the costliest package?

select top 1 with ties PNAME,TITLE,DEVELOPIN,DCOST from Software

order by DCOST desc

--70. Display the names of the packages which have sold less than the

--average number of copies.

select TITLE,SOLD as COPIES from Software

WHERE SOLD<(SELECT AVG(SOLD) FROM Software)

ORDER BY COPIES DESC

--71. Who are the authors of the packages which have recovered more than

--double the development cost?

SELECT PNAME,TITLE,DEVELOPIN,(SCOST\*SOLD) AS TOTAL\_SALES,DCOST FROM Software

WHERE (SCOST\*SOLD)>(2\*DCOST)

--72. Display the programmer names and the cheapest packages developed

--by them in each language.

SELECT PNAME,DEVELOPIN,MIN(DCOST) AS DCOST FROM Software

GROUP BY PNAME,DEVELOPIN

ORDER BY DCOST

--73. Display the language used by each programmer to develop the highest

--selling and lowest selling package.

WITH PackageSales AS (

SELECT PNAME, DEVELOPIN, SOLD,

ROW\_NUMBER() OVER (PARTITION BY PNAME ORDER BY SOLD DESC) AS RankHighest,

ROW\_NUMBER() OVER (PARTITION BY PNAME ORDER BY SOLD ASC) AS RankLowest

FROM Software

)

SELECT PNAME,

MAX(CASE WHEN RankHighest = 1 THEN DEVELOPIN END) AS LanguageForHighestSelling,

MAX(CASE WHEN RankLowest = 1 THEN DEVELOPIN END) AS LanguageForLowestSelling

FROM PackageSales

GROUP BY PNAME;

--74. Who is the youngest male programmer born in 1965?

SELECT TOP 1 WITH TIES \* FROM Programmer

WHERE YEAR(DOB)=1965 AND GENDER='M'

ORDER BY DOB DESC

--75. Who is the oldest female programmer who joined in 1992?

SELECT TOP 1 WITH TIES \* FROM Programmer

WHERE YEAR(DOJ)=1992 AND GENDER='F'

ORDER BY DOB ASC

--76. In which year was the most number of programmers born?

SELECT TOP 1 WITH TIES YEAR(DOB) AS YEAR,COUNT(PNAME) AS NUM\_OF\_PROGRAMMERS FROM Programmer

GROUP BY YEAR(DOB)

ORDER BY NUM\_OF\_PROGRAMMERS DESC

--77. In which month did the most number of programmers join?

SELECT TOP 1 WITH TIES DATENAME(MM,DOJ) AS MONTH,COUNT(PNAME) AS PROGRAMMER\_COUNT FROM Programmer

GROUP BY DATENAME(MM,DOJ)

ORDER BY PROGRAMMER\_COUNT DESC

--78. In which language are most of the programmer’s proficient?

select prof,sum(no\_of\_programmers) as total\_programmers

from (

--count for prof1

select prof1 as prof, count(prof1) as no\_of\_programmers

from Programmer

group by PROF1

union all

--count for prof2

select prof2 as prof, count(prof2) as no\_of\_programmers

from programmer

group by prof2

) as combined

group by prof

-- alternative method

-- CREATING VIEW

CREATE VIEW TOTAL\_PROGRAMES AS

SELECT PROF1 AS PROF,COUNT(PROF1) AS NUM\_OF\_PROGRAMMERS FROM Programmer

GROUP BY PROF1

UNION ALL

SELECT PROF2 AS PROF,COUNT(PROF2) AS NUM\_OF\_PROGRAMMERS FROM Programmer

GROUP BY PROF2

--TACKLING THE SOLUTION

SELECT PROF,SUM(NUM\_OF\_PROGRAMMERS) AS COUNT\_PROGRAMMERS FROM TOTAL\_PROGRAMES

GROUP BY PROF

ORDER BY COUNT\_PROGRAMMERS DESC

--79. Who are the male programmers earning below the average salary of

--female programmers?

select \* from Programmer

where GENDER='m' and SALARY<( Select AVG(SALARY) from Programmer where GENDER='F')

--80. Who are the female programmers earning more than the highest paid Males?

select \* from Programmer where gender='f' and

salary>(select max(salary) from programmer where gender='m')

--81. Which language has been stated as the proficiency by most of the programmers?

select top 1 with ties prof,sum(count\_language) as counts from

(select prof1 as prof,count(prof1) as count\_language from programmer

group by prof1

union all

select prof2 as prof,count(prof2) as count\_language from programmer

group by prof2

) as combined

group by prof

order by counts desc

--82. Display the details of those who are drawing the same salary.

with salary\_count as

(select salary,count(salary) as count from programmer

group by salary)

select p.pname,p.salary,p.age from programmer as p

inner join salary\_count as s

on p.salary=s.salary

where s.count >1

--83. Display the details of the software developed by the male programmers

--earning more than 3000.

select s.pname,p.salary,s.title,s.developin,s.scost,s.dcost,s.sold from software as s

inner join programmer as p

on s.pname=p.pname

where p.salary>3000 and p.gender='m'

--84. Display the details of the packages developed in Pascal by the female

--programmers.

select s.pname,p.gender,s.title,s.developin,s.scost,s.dcost,s.sold

from software as s inner join programmer as p

on s.pname=p.pname

where s.developin='pascal' and p.gender='f'

--85. Display the details of the programmers who joined before 1990.

select \* from programmer

where year(doj)<1990

--86. Display the details of the software developed in C by the female

--programmers at Pragathi.

select s.pname,p.gender,s.title,s.developin,s.scost,s.dcost,s.sold,st.institute from

software as s inner join programmer as p

on s.pname=p.pname

inner join studies as st

on s.pname=st.pname

where s.developin='c' and p.gender='f' and st.institute='pragathi'

--87. Display the number of packages, number of copies sold and sales value

--of each programmer institute wise.

select st.institute,s.pname,count(s.title) as number\_of\_packages,

sum(s.sold) as num\_of\_copies,sum(s.scost\*s.sold) as total\_sales

from software as s inner join studies as st

on s.pname=st.pname

group by st.institute,s.pname

--88. Display the details of the software developed in dBase by male

--programmers who belong to the institute in which the most number of

--programmers studied.

;with intitute\_wise\_students\_count as

(select top 1 institute,count(pname) as counts from studies

group by institute

order by counts desc)

select s.pname,p.gender,s.title,s.developin,s.scost,s.dcost,s.sold,st.institute

from software as s inner join programmer as p

on s.pname=p.pname

inner join studies as st

on s.pname=st.pname

where s.developin='dbase'

and p.gender='m' and st.institute in (select institute from intitute\_wise\_students\_count)

--89. Display the details of the software developed by the male programmers

--born before 1965 and female programmers born after 1975.

select s.pname,p.gender,p.dob,s.title,s.developin,s.scost,s.dcost,s.sold

from software as s inner join programmer as p

on s.pname=p.pname

where (p.gender='m' and year(p.dob)<1965) or (p.gender='f' and year(p.dob)>1975)

--Note: we have one employee who born before 1965 but he not develped any packages

--90. Display the details of the software that has been developed in the

--language which is neither the first nor the second proficiency of the

--programmers.

; with prof\_with\_ranks as(

select prof,DENSE\_RANK()over(order by counts desc) as ranks from

(select prof,SUM(count\_of\_language) as counts from

(

select prof1 as prof,COUNT(prof1) as count\_of\_language from Programmer

group by PROF1

union all

select prof2 as prof,COUNT(prof2) as count\_of\_language from Programmer

group by PROF2

)as combined

group by prof

) as ss)

select \* from Software as s

inner join prof\_with\_ranks as pr

on s.DEVELOPIN=pr.prof

where pr.ranks>2

--91. Display the details of the software developed by the male students at

--Sabhari.

select s.PNAME,st.INSTITUTE,p.GENDER,s.DEVELOPIN,s.TITLE,s.DCOST,s.SCOST,s.SOLD from

Software as s

inner join Programmer as p

on s.PNAME=p.PNAME

inner join Studies as st

on s.PNAME=st.PNAME

where p.GENDER='m' and st.INSTITUTE='Sabhari'

--92. Display the names of the programmers who have not developed any

--packages.

select \* from Programmer

where PNAME not in (select pname from Software)

--93. What is the total cost of the software developed by the programmers of

--Apple?

select st.INSTITUTE,SUM(s.dcost) as 'Development Cost' from

Software as s inner join Studies as st

on s.PNAME=st.PNAME

where st.INSTITUTE='Apple'

group by st.INSTITUTE

--94. Who are the programmers who joined on the same day?

select \* from programmer

where doj in

(select doj from

(select DOJ,COUNT(doj) as counts from Programmer

group by DOJ

having COUNT(doj)>1) as ss)

--95. Who are the programmers who have the same Prof2?

select \* from Programmer

where PROF2 in (select PROF2 from

(select PROF2,COUNT(prof2) as counts

from Programmer

group by prof2

having COUNT(prof2)>2) as aa))

--96. Display the total sales value of the software institute wise.

select INSTITUTE,SUM(SCOST\*SOLD) as Total\_sales

from Software inner join Studies

on Software.PNAME=Studies.PNAME

group by INSTITUTE

order by Total\_sales desc

--97. In which institute does the person who developed the costliest package

--study?

select Software.PNAME,Studies.INSTITUTE from Software inner join Studies

on Software.PNAME=Studies.PNAME

where SCOST=(select MAX(SCOST) from Software)

--98. Which language listed in Prof1, Prof2 has not been used to develop any

--package?

select prof1 from Programmer

where PROF1 not in (select DEVELOPIN from Software)

union

select prof2 from Programmer

where PROF2 not in (select developin from Software)

--99. How much does the person who developed the highest selling package

--earn and what course did he/she undergo?

select p.PNAME,p.SALARY,s.TITLE,st.COURSE,st.INSTITUTE,s.DCOST from Programmer as p

inner join Software as s

on p.PNAME=s.PNAME

inner join Studies as st

on p.PNAME=st.PNAME

where s.DCOST=(select MAX(dcost) from Software)

--100. What is the average salary for those whose software sales is more than

--50,000?

select AVG(salary) avg\_salary from Programmer inner join Software

on Programmer.PNAME=Software.PNAME

where SCOST\*SOLD>500000

--101. How many packages were developed by students who studied in

--institutes that charge the lowest course fee?

select st.INSTITUTE,COUNT(\*) total\_DPackages from Software as s

inner join Studies as st

on s.PNAME=st.PNAME

where st.COURSE\_FEE=(select MIN(course\_fee) from Studies)

group by st.INSTITUTE

--102. How many packages were developed by the person who developed the

--cheapest package? Where did he/she study?

select s.PNAME,st.INSTITUTE,COUNT(s.TITLE) as num\_of\_packages from Software as s

inner join Studies as st

on s.PNAME=st.PNAME

where s.PNAME in(

select pname from Software where DCOST=(select MIN(dcost) from Software))

group by s.PNAME,st.INSTITUTE

--103. How many packages were developed by female programmers earning

--more than the highest paid male programmer?

select distinct(COUNT(\*)) as number\_of\_female\_employees

from Software as s inner join Programmer as p

on s.PNAME=p.PNAME

where p.GENDER='f' and p.SALARY>(select MAX(salary) from Programmer where GENDER='m')

--104. How many packages are developed by the most experienced

--programmers from BDPS?

select COUNT(\*) as 'number of packages' from

Software as s inner join Programmer as p

on s.PNAME=p.PNAME

inner join Studies as st

on s.PNAME=st.PNAME

where st.INSTITUTE='BDPS' and

p.DOJ=(select MIN(doj) from Programmer as p

inner join Studies as st

on p.PNAME=st.PNAME

where st.INSTITUTE='bdps')

--105. List the programmers (from the software table) and the institutes they

--studied at.

select distinct(s.PNAME),st.INSTITUTE from Software as s

inner join Studies as st

on s.PNAME=st.PNAME

--106. List each PROF with the number of programmers having that PROF

--and the number of the packages in that PROF.

; with table\_prof as(

select prof,SUM(count\_prof) as counts from

(select prof1 as prof,COUNT(prof1) as count\_prof from Programmer

group by PROF1

union all

select prof2 as prof,COUNT(prof2) as count\_prof from Programmer

group by PROF2

) as combined

group by prof)

select s.DEVELOPIN,tp.counts as Number\_of\_employees,COUNT(s.developin) as number\_of\_packages

from Software as s

inner join table\_prof as tp

on s.DEVELOPIN=tp.prof

group by s.DEVELOPIN,tp.counts

--107. List the programmer names (from the programmer table) and the

--number of packages each has developed

select p.PNAME,COUNT(s.TITLE) as number\_of\_packages from Programmer as p

inner join Software as s

on p.PNAME=s.PNAME

group by p.PNAME