**THE PADS**

select concat(Name,'(',left(Occupation,1),')')

from occupations

order by name asc;

select concat('There are a total of ', count(occupation),' ', lower(occupation),'s.')

from occupations

group by occupation

order by count(occupation), occupation

**OCCUPATIONS**

select Doctor, Professor, Singer, Actor from (select name, occupation, row\_number() over(partition by occupation order by name asc) as row\_number from occupations)

PIVOT

(max(name) for occupation in ('Doctor' Doctor, 'Professor' Professor, 'Singer' Singer, 'Actor' Actor))

order by row\_number;

**BINARY TREE NODES**

select n,

case when p is null then 'Root'

when n in (select distinct p from bst) then 'Inner'

else 'Leaf'

end

from bst

order by n asc

**WEATHER OBSERVATION STATION 18**

select round(abs(max(lat\_n)-min(lat\_n)) + abs(max(long\_w)-min(long\_w)),4)

from station

**WEATHER OBSERVATION STATION 19**

select round(sqrt(power(max(lat\_n)-min(lat\_n),2) + power(max(long\_w)-min(long\_w),2)),4)

from station

**WEATHER OBSERVATION STATION 20 (Median)**

select round(avg(lat\_n),4)

from

(select \*, row\_number() over (order by lat\_n desc) as desc\_lat,

row\_number() over (order by lat\_n asc) as asc\_lat

from station) as a

where asc\_lat in (desc\_lat, desc\_lat+1, desc\_lat-1)

**THE REPORT**

SELECT CASE

WHEN Grades.Grade < 8 THEN 'NULL'

ELSE Students.Name

END

, Grades.Grade, Students.Marks

FROM STUDENTS

LEFT JOIN GRADES ON STUDENTS.MARKS >= MIN\_MARK

AND STUDENTS.MARKS <= MAX\_MARK

ORDER BY Grades.Grade DESC, Students.Name, Students.Marks;