**ON DELETE CASCADE** means that if the parent record is deleted, any child records are also deleted. ...

**ON UPDATE CASCADE** means that if the parent primary key is changed, the child value will also change to reflect that

------------------------

Recursive relation example:-

create table employees(

employee\_id integer primary key,

manager\_id integer,

foreign key (department\_id) references departments(employee\_id) #recursive relation

)

**---------------------------------------------------**

**OVER()** clause to define a set of rows

**PARTITION BY** clause to define a subset of data in a partition

**Order by in partition** clause to sort

Dense rank don’t have gap between rank.

<https://www.sqlshack.com/overview-of-sql-rank-functions/>

----------------------------------------------------

Diff date – Diffdate(‘month’, tdate, GETDATE()) 🡪 return diff between cureent vs given date like tdate

PIVOT operator we can very easily transform rows to columns

'City'+ cast(row\_number() over(partition by Country order by Country)

as varchar(10)) ColumnSequence

🡪 can declare variable type in alias

cast() 🡪 convert type of the variable

---------------------------------------------------------

where isnumeric**(value) = 1 🡪 query will return only numeric**

GROUP BY DepartmentName ORDER BY COUNT(\*) DESC 🡪 can apply directly count in order by

----------------------------------------------------------

cast(dob as Date) as date\_part 🡪 convert 19-10-2012 12:12:09 to 19-10-2012

day(dob) 🡪 will return 19

month(dob) 🡪 10

DATEADD( DAY, -1, CAST(GETDATE() AS DATE) ) 🡪 return yesterday date

DATEDIFF(YEAR, HireDate, GETDATE()) 🡪 return year difference between Hire date and today date

strftime('%m',at.date\_event) 🡪 return month of the date

datetime(a.dob, ‘-1 days’) 🡪 return yesterday date

----------------------------------------------------------

For deletion with parent child relation firslty always delete child thn parent

CHARINDEX('M',Name) = 1

LEFT(Name, 1) = 'M'

SUBSTRING(Name, 1, 1) = 'M'

Country IN ('Germany', 'France', 'UK');

**BETWEEN** 10 AND 20

**UNION** selects only distinct values.

**UNION** ALL to also select duplicate values

**EXISTS** operator returns TRUE if the subquery returns one or more records

**Any** return true if any matched result found in the subquery

**All** return true if all matched result found in the subquery

<https://www.w3schools.com/sql/func_sqlserver_month.asp>

----------------------------

Question 1: How to find nth highest salary in sql

SELECT TOP 1 \* from (select TOP 5 \* from Employees order by Salary desc) RESULT order by Salary asc;

-------------------------------------

dense\_rank():-

with result as (

select salary,

dense\_rank() over (order by salary desc) as denserank ##create row number over salary

from employees

)

select TOP 1 \* from result where denserank = 4 order by Salary

-------------------------------------

ROW\_NUMBER():-

Another option is row number but for same salary it will create different row number unlike dense rank

ROW\_NUMBER() OVER (ORDER BY SALARY DESC) AS ROWNUMBER

<https://www.youtube.com/watch?v=fvPddKyHxpQ&list=PL6n9fhu94yhXcztdLO7i6mdyaegC8CJwR&index=1>

**Questions 2: SQL query to get organization hierarchy**

select e.EmployeeName as ManagerName, null from Employees as e

where e.ManagerID is null

union

select e1.EmployeeName as ManagerName, e.EmployeeName as EmployeeName from Employees as e

join Employees as e1 on e.EmployeeId = e1.ManagerID

-----------------------------------

Declare @ID int ;

Set @ID =7;

WITH EmployeeCTE as(

select EmployeeId, EmployeeName, ManagerID

from Employees where ManagerID is NULL

union all

select e.EmployeeID, e.EmployeeName, e.ManagerID from Employees as e

join EmployeeCTE as ec on

ec.EmployeeId = e.ManagerID

)

select ec.EmployeeName, ISNULL(ee.EmployeeName, 'No Boss') from EmployeeCTE

as ec left join EmployeeCTE as ee

on ee.EmployeeID = ec.ManagerID

**Part 4 Delete duplicate rows in sql**

with ec as (

select \*, ROW\_NUMBER() OVER(PARTITION BY ID ORDER BY ID DESC) RowNumber from Employees

)

select \* from ec WHERE RowNumber = 1

delete from (select \*, ROW\_NUMBER() OVER(PARTITION BY ID ORDER BY ID DESC) rn from Employees) Employees --------------- not working

**Part 5 SQL query to find employees hired in last n months**

select \* from Employees where DATEDIFF(YEAR, HireDate, GETDATE()) between 0 and 3

**Part 7 SQL query to find rows that contain only numerical data**

select \* from Employees where isnumeric**(value) = 1**

**-----------------------------------------------------------**

**SQL Query to find department with highest number of employees**

select top 1 DepartmentName from Employees as e join Departments as d on e.DepartmentID = d.DepartmentID group by DepartmentName order by count(DepartmentName) desc

--------------------------

**Without like doncditoin, first character with ‘M’**

SELECT \* FROM Students WHERE CHARINDEX('M',Name) = 1

SELECT \* FROM Students WHERE LEFT(Name, 1) = 'M'

SELECT \* FROM Students WHERE SUBSTRING(Name, 1, 1) = 'M'