function - functionName()

set of statements

will execute when we invoke it

Reusability

BuiltInLibrary Functions

Mathematical Functions

ABS() - Absolute Value - SELECT ABS(-97)

SQRT() - Squareroot - SELECT SQRT(1225)

SELECT SQRT(768)

POWER() - Expnential Power Val - SELECT POWER(3,10)

LOG() - Logarthimic value - SELECT LOG(10, 10)

SELECT \* FROM EMployee

Aggregation Functions

MAX() - Select MAX(Salary) from Employee

MIN() - SELECT MIN(Salary) 'Min Salary' from Employee

AVG() - SELECT AVG(SALARY) 'Average Salary' From Employee

SUM() - SELECT SUM(Salary) 'Total Salry' From Employee

COUNT()- SELECT COUNT(Eid) 'Total Employees' from Employee

select count(eid) 'Total EMP' from employee where deptid = 10

SELECT Ename 'Employee Name' From Employee

Date Functions

GETDATE() - Current Date - Select GetDate() 'Today'

default format will be yyyy-MM-dd hh: mm : SS

DATEPART() - gets specific part of the date

SELECT DATEPART(YEAR, GETDATE()) 'Current Year'

SELECT DATEPART(MONTH, GETDATE()) 'Current Year'

SELECT DATEPART(DAY, GETDATE()) 'Current Year'

emp Date of birth is 1990-08-13

DATEDIFF()

SELECT DATEDIFF(MONTH,'1990-08-13',GETDATE()) 'AGE IN MONTHS'

SELECT DATEDIFF(YEAR,'1990-08-13',GETDATE()) 'AGE'

DOJ - Date of Jpining

SELECT ENAME, SALARY , DATEDIFF(YEAR, DOJ, GETDATE) 'Exp' FROM Employee

DATEADD()

SELECT DATEADD(DAY,200, GETDATE())

SELECT DATEADD(MONTH,20, GETDATE())

String Function

UPPER() - SELECT UPPER(Ename) 'ENAME' From Employee

LOWER()

TRIM()

LTRIM() - Removes the spaces from Left side

RTRIM()

SUBSTRING() - SELECT SUBSTRING('HYDERABAD',3,2)

SELECT UPPER(SUBSTRING(Ename,2,2)) from Employee

REPLACE()

SELECT \* FROM Employee

GROUP BY

First group the data

then apply some aggregation functions on the grouped data

\* group by can not applied with out aggregation function

To check the count of employee based on gender

SELECT Gender, COUNT(Eid) 'Total Emp' FROM Employee GROUP BY Gender

to display the no of employees in each department

SELECT DeptId , COUNT(Eid) from Employee GROUP BY DeptId

To Display the no of employees under a magerID

SELECT MngrId , COUNT(Eid) from Employee GROUP BY MngrId

total salary to be on each department

SELECT DeptId, SUM(Salary) FROM Employee GROUP BY DeptId

SELECT DeptId, SUM(Salary) FROM Employee WHERE DeptId = 10 GROUP BY DeptId

\* Where condition must be before group by

SELECT DeptId, SUM(Salary) FROM Employee WHERE DeptId = 30 GROUP BY DeptId

HAVING - Used to apply condition on summarised result

display the total no of emplyess and deptID where the dept contains more tha 4 employees

SELECT DeptID, Count(Eid) 'Total Emp' from Employee Group by DeptId HAVING Count(Eid) >4

SELECT Gender, Count(Eid) 'Total Emp' FROM Employee Group by Gender Having Count(Eid)<=3

Display the DepartmentId and Total Salary of the employee where tatal salary greater tha 15000

select DeptId, sum(salary) "Total Salary" from employee

group by DeptId having sum(salary)>15000

Update employee set ename='Ravi', Salary = '8000' where eid = 1012

Select \* from Employee

Select \* from Department

Ename, Salary , DeptName

SELECT E.Ename, E.Salary, D.DeptName FROM Employee E

INNER JOIN Department D

ON E.DeptId = D.DeptId

\* By default will be considered as inner join

SELECT E.Ename, E.Salary, D.DeptName FROM Employee E

JOIN Department D

ON E.DeptId = D.DeptId

LEFT OUTER JOIN

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

SELECT E.Ename, E.Salary, D.DeptName FROM Employee E

LEFT OUTER JOIN Department D

ON E.DeptId = D.DeptId

RIGHT OUTER JOIN

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

SELECT E.Ename, E.Salary, D.DeptName FROM Employee E

RIGHT OUTER JOIN Department D

ON E.DeptId = D.DeptId

SELECT \* FROM Department

FULL OUTER JOIN

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

SELECT E.Ename, E.Salary, D.DeptName FROM Employee E

FULL OUTER JOIN Department D

ON E.DeptId = D.DeptId

SELECT \* FROM Employee

Update Employee set MngrId = 1012 Where Eid = 1012

SELF JOIN

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

its an inner join , where the same table is joining 2 times

Select eid , ename, mngrid from employee

E1 E2

EID ENAME MID EID ENAME MID

E1.EID, E1.ENAME, E2.ENAME

E1.MID = E2.EID

select E1.Eid, E1.Ename, E2.Ename from Employee E1

JOIN Employee E2

ON E1.MngrId = E2.Eid

CREATE TABLE [dbo].[HDDetails](

[Id] [int] IDENTITY(1,1) NOT NULL,

[MemoryCapacity] [varchar](50) NULL,

[Price] [int] NULL

) ON [PRIMARY]

GO

CREATE TABLE [dbo].[ProcessorDetails](

[Pid] [int] IDENTITY(1,1) NOT NULL,

[ProcessorName] [varchar](50) NULL,

[Price] [int] NULL

) ON [PRIMARY]

GO

INSERT INTO ProcessorDetails (ProcessorName ,Price) VALUES ('i3',4000)

INSERT INTO ProcessorDetails (ProcessorName ,Price) VALUES ('i5',5000)

INSERT INTO ProcessorDetails (ProcessorName ,Price) VALUES ('i7',7000)

INSERT INTO HDDetails (MemoryCapacity ,Price) VALUES ('250GB',2000)

INSERT INTO HDDetails (MemoryCapacity ,Price) VALUES ('520GB',3000)

INSERT INTO HDDetails (MemoryCapacity ,Price) VALUES ('1TB',4500)

Select \* from ProcessorDetails

Select \* from HDDetails

SELECT P.ProcessorName, H.MemoryCapacity, P.Price+H.Price 'TotalPrice' FROM ProcessorDetails P

CROSS JOIN

HDDetails H

TOP

FETCH NEXT

SKIP

CREATE TABLE DepartmentLocation (DeptId int , DeptLocation Varchar(100))

Insert into DepartmentLocation Values(10, 'Delhi'),(20,'Chennai'),(30, 'Pune'),(40,'Mumbai'),(50,'Hyderabad')

Select \* from DepartmentLocation

Select \* from Department

Select \* from Employee

Eid, Ename, Salary, DeptName, DeptLocation

JOINING more than 2 Tables

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

SELECT E.Eid, E.Ename, E.Salary, D.DeptName, DL.DeptLocation FROM Employee E

JOIN Department D ON E.DeptID = D.DeptId

JOIN DepartmentLocation DL ON E.DeptID = DL.DeptId

Subqueries

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

one query result is going to be an input for other query

Display all the employess who are working in the same department of 'rama'

Select \* from employee where deptid =(select deptid from employee where ename ='Rama')

Display all the employee details who are working under

the manger of employee with eid 1004

SELECT \* FROM EMPLOYEE WHERE MngrId=

( select MngrId from employee where Eid=1004)

USING VARIABLES

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

DECLARE @avgSal int

select @avgSal = avg(salary) from Employee

print @avgsal

Declare @totEmp int

Select @totEmp = count(eid) from employee

print @totEmp