Create table Using Query

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

create table Employee(

EmpNo bigint ,EName varchar(50) ,DeptNo bigint ,Job varchar(10),HireDate datetime)

create table Dept(

DeptNo bigint ,DName varchar(20) ,Loc varchar(20) )

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

Create,Insert,Delete,Update,Drop

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

Table Creation:-

1.Give Table Name

TS\_Company\_Master

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

Id

Name

Remarks

TS\_Group\_Master

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

Id

Name

Remarks

TS\_Item\_Master

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

Id

Name

Company\_id

Group\_id

Rate

Remarks

TS\_Purchse\_Header

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

Id

Purchse\_no

Trns\_Date

Total

Remarks

TS\_Purchase\_Detail

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

Id

Header\_id

Sl\_no

Item\_id

Qty

rate

amount

Insert Query(TS\_Company\_Master)

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

insert into TS\_Company\_Master (id,name,remarks) values (1,'P&G','')

insert into TS\_Company\_Master (id,name,remarks) values (2,'XENICAL','')

insert into TS\_Company\_Master (id,name,remarks) values (3,'J&J','')

insert into TS\_Company\_Master (id,name,remarks) values (4,'JAC','')

Insert Query(TS\_Group\_Master)

insert into TS\_Group\_Master (id,name,remarks) values (1,'INHALER','')

insert into TS\_Group\_Master (id,name,remarks) values (2,'DROPS','')

insert into TS\_Group\_Master (id,name,remarks) values (3,'POWDER','')

insert into TS\_Group\_Master (id,name,remarks) values (4,'OINTMENT','')

Delete Query(TS\_Company\_Master)

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

delete from TS\_Company\_Master

Delete Query(TS\_Company\_Master)

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

delete from TS\_Company\_Master where id=1

Delete Query(TS\_Group\_Master)

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

delete from TS\_Group\_Master where name='INHALER'

Update Query(TS\_Company\_Master)

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

update TS\_Company\_Master set name='P&G' where id=2

update Employee set EName=(select EName from Employee where EmpNo=2000)

Drop

---------

drop table TS\_Company\_Master

Create a new table based on an existing table

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

select id,name into Copy\_Company from TS\_Company\_Master

select \* into Copy\_TS\_Company\_Master from TS\_Company\_Master

Alter A table

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

Alter table columnexisting data type to another

Alter table table\_name Alter column column\_name datatype

Eg: Alter table ss Alter column i varchar(30)

Add new Column intto the table

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

Alter table table\_name add newcolumn\_name datatype

Alter table ss add New\_colum bigint

Renamae a table name

exec sp\_rename oldtable,new\_name

rename databse

exec Sp\_renamedb old name,new name

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

A: % (Wildcard - Character(s) to Match) (Transact-SQL)

Matches any string of zero or more characters. This wildcard character can be used as either a prefix or a suffix. For more information, see Pattern Matching in Search Conditions.

Like %,-

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

- = reperesent single value

% = represent any value

6. select \* from TS\_Company\_Master where name like '%a%'

o/p fetch datas which contain the letter ‘a’ in any position.

7. select \* from TS\_Company\_Master where name like 'j%'

O/p fetch datas begins with letter ‘j’

8. select \* from TS\_Company\_Master where name like '%j'

O/P fetch datas ending with letter ‘j’

9. select \* from TS\_Company\_Master where name like '\_a%'

O/p fetch datas which contains letter ‘a’ at position 2, ‘\_’ indicates position

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

B: UNION (Transact-SQL)

Combines the results of two or more queries into a single result set that includes all the rows that belong to all queries in the union.

The UNION operation is different from using joins that combine columns from two tables.

UNION

Specifies that multiple result sets are to be combined and returned as a single result set.

ALL

Incorporates all rows into the results. This includes duplicates. If not specified, duplicate rows are removed.

Union & Union All

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

1. select field\_name1,2,3 from table\_name1

Union

select field\_name1,2,3 from table\_name2

2. SELECT Name, Id FROM TS\_Company\_Master

UNION ALL

SELECT Name, id FROM TS\_group\_Master

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

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

1. select \* from table\_name1,table\_name2

Mathematic Fumctions

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

1. ABS- Returns the absolute and positive value.

Syntax - ABS ( numeric )

Eg:

SELECT ABS(-1.0) -- Value = 1.0

SELECT ABS(0.0) -- Value = 0.0

SELECT ABS(1.0) -- Value = 1.0

2. FLOOR- Returns the largest integer less than or equal to the given numeric expression.

Syntax - FLOOR( numeric)

Eg:

SELECT FLOOR(32.15) -- Value = 32

SELECT FLOOR(-32.15) -- Value = -33

SELECT FLOOR($32.15) -- Value = 32.0000

3. CEILING- Returns the smallest integer greater than, or equal to the input value

Syntax - CEILING( numeric)

Eg:

SELECT CEILING(32.15) -- Value = 33

SELECT CEILING(-32.15) -- Value = -32

SELECT CEILING($32.15) -- Value = 33.00

4. ROUND- Return rounded value.

Syntax - ROUND( numeric, length)

Eg:

SELECT ROUND(32.15 ,1) -- Value = 32.20

SELECT ROUND(32.14 ,1) -- Value = 32.10

SELECT ROUND(32.14998 ,4) -- Value = 32.15000

SQL Aggregate Functions

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

1. AVG - Average value of columns

Eg:

Select AVG(Salary) FROM A

2. COUNT - number of rows

Eg:

Select COUNT(\*) FROM A

SELECT COUNT (DISTINCT Salary) FROM CUSTOMERS

3. MAX - Maximun or Highest number in a column

Eg:

SELECT MAX(Salary) FROM CUSTOMERS

4. MIN - Minimum or Lowest number in a column

Eg:

SELECT MIN(Salary) FROM CUSTOMERS

5. SUM - Total number in a column

Eg:

SELECT SUM(Salary) FROM CUSTOMERS

SQL String Functions

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

1. ASCII - Returns the ASCII code value of a keyboard button and the rest etc (@,R,9,\*) .

Syntax - ASCII ( character)

Eg:

SELECT ASCII('a')

2. SPACE -Returns spaces in your SQL query (you can specific the size of space).

Syntax - SPACE ( integer)

Eg:

SELECT ('SQL') + SPACE(1) + ('String')+SPACE(1)+('Functions')

3.REPLACE-Replaces all occurrences of the string2 in the string1 with string3.

Syntax - REPLACE ( 'string1' , 'string2' , 'string3' )

Eg:

SELECT REPLACE('All Function' , 'All', 'SQL')

4.LEFT-Returns left part of a string with the specified number of characters.

Syntax - LEFT ( string , integer)

Eg:

SELECT LEFT('TravelYourself', 6)

5.RIGHT-Returns right part of a string with the specified number of characters.

Syntax - RIGHT( string , integer)

Eg:

SELECT RIGHT('TravelYourself', 6)

6. REPLICATE-Repeats string for a specified number of times.

Syntax - REPLICATE (string, integer)

Eg:

SELECT REPLICATE('Sql', 2)

7.SUBSTRING-Returns part of a string.

Syntax - SUBSTRING ( string, startindex , length )

Eg:

SELECT SUBSTRING('SQLServer', 4, 3)

8. LEN-Returns number of characters in a string.

Syntax - LEN( string)

Eg:

SELECT LEN('SQLServer')

9. REVERSE-Returns reverse a string.

Syntax - REVERSE( string)

Eg:

SELECT REVERSE('SQLServer')

10. LOWER-Convert string to lowercase.

Syntax - LOWER( string )

Eg:

SELECT LOWER('SQLSERVER')

11. UPPER-Convert string to Uppercase.

Syntax - UPPER( string )

Eg:

SELECT UPPER('sqlserver')

12.LTRIM-Returns a string after removing leading blanks on Left side.

Syntax - LTRIM( string )

Eg:

SELECT LTRIM(' sqlserver')

13. RTRIM -Returns a string after removing leading blanks on Right side.

Syntax - RTRIM( string )

Eg:

SELECT RTRIM('SqlServer ')

and ,or , in ,<,>,>=,<=,not,between

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

1. select \* from TS\_Company\_Master where name='veena' and id=3

2. select \* from TS\_Company\_Master where name='veena' or id=8

3. SELECT Name, Remarks

FROM TS\_Company\_Master

WHERE name IN ('Ammu', 'Veena','priya', 'Jenifar')

4. SELECT Name, Remarks

FROM TS\_Company\_Master

WHERE name not IN ('Ammu', 'Veena','priya', 'Jenifar')

5. select \* from TS\_Purchse\_detail where header\_id in(select id from TS\_Purchse\_Header where purchase\_no not in('A001','A003'))

6. select \* from TS\_Company\_Master where id >=2

7. select \* from TS\_Company\_Master where id >2

8. select \* from TS\_Company\_Master where id <=2

9. select \* from TS\_Company\_Master where id <2

10. select \* from TS\_Company\_Master where id = 2

11. select \* from TS\_Company\_Master where id between 2 and 4

gRoup by

--------

select name from TS\_Company\_Master group by name

select name,Count(\*) from TS\_Company\_Master group by name

select name,count(name) from TS\_Company\_Master where name='ammu' group by name

Condition having

------------------------+

select name,count(\*) from TS\_Company\_Master group by name having max(id)>4

select name from TS\_Company\_Master group by name having avg(id)>1

-----------

Ascending,Desc

ORder by

select name from TS\_Company\_Master order by name

select name from TS\_Company\_Master order by name desc

select \* from ts\_company\_master where order by name desc

Distinct

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

select distinct(name) from TS\_Company\_Master

cast,convert

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

SELECT CAST(10.6496 AS int)

SELECT CAST(10.3496847 AS money)

SELECT 'The salary is ' + CAST(salary AS varchar(12)) AS Salary

FROM employee

WHERE salary BETWEEN 2000 AND 3000