**TSQL EXAM**

1.       How would you limit values that can be entered in the column of table that based on values in other table.

* **Use a where clause to filter the result set.**

2.       In what scenario you would prefer normalized tables over deformalized table and vice versa and why?

* **Normalized tables perform better under write intensive conditions. The updates are very fast because the data to be updated is located at a single place and there are no duplicates. Similarly the inserts are very fast because the data has to be inserted at a single place and does not have to be duplicated.**
* **De-normalized tables perform better under read intensive conditions. The data is present in the same table so there is no need for any joins, hence the selects are very fast.**

3.       How will you List all the table and column names in a database which have any column of xml data type.

**SELECT T.NAME AS [TABLE NAME], C.NAME AS [COLUMN NAME], P.NAME AS [DATA TYPE]**

**FROM SYS.OBJECTS AS T**

**JOIN SYS.COLUMNS AS C**

**ON T.OBJECT\_ID=C.OBJECT\_ID**

**JOIN SYS.TYPES AS P**

**ON C.SYSTEM\_TYPE\_ID=P.SYSTEM\_TYPE\_ID**

**WHERE T.TYPE\_DESC='USER\_TABLE'**

**and P.NAME ='xml';**

4.       If data content of table belongs to group of users (HR, Marketing,IT etc…), what can be done to expose limited data to that group/team which belong to them and not others.

* Use GroupBy Clause and WHERE clause

5.       If column is defined to be Identity in the table, how can you enter new values of your choice which has not been entered.

* **--SET IDENTITY\_INSERT to ON.**

**SET IDENTITY\_INSERT dbo.TableName ON;**

**GO**

**-- Insert an explicit ID value of your choice.**

**INSERT INTO dbo.TableName (ID, Name) VALUES (3, 'Value');**

**GO**

6.       How would you search for record where a given column contains “%”.

* select \* from TableName

where Columnname LIKE ('%%%')

OR

* select \* from TableName

where Columnname LIKE ('%[%]%')

OR

* select \* from TableName

where Columnname LIKE '%\%%' escape '\'

7.       Arrange Filter criteria in order of their performance of execution

a.       Col1 like ‘abc%’

b.      Col1 like ‘a%yz’

c.       Col1=’abcyz’

d.      Col1 like ‘%yz’

1. **c.       Col1=’abcyz’**
2. **a.       Col1 like ‘abc%’**
3. **b.      Col1 like ‘a%yz’**
4. **d.      Col1 like ‘%yz’**

8.       Write a Stored Proc which takes input of date and following three output date:

a.       Same day of last week ( if input date is 2014/12/01 result should be 2014/11/24)

b.      Same day of Same week of Last month (if input date is 2014/12/01 result should be

c.       Same day of Same week of same month of last year (if input date is 2014/12/01 result 2014/11/03) should be 2014/12/02)

CREATE procedure [dbo].[DateCalculations]

@AsOfDate as datetime

as

begin

set nocount on

--declare @AsOfDate as datetime =getdate(),

declare @SameDayOfLastWeek as date,

@SameDayOfSameWeekOfLastMonth as datetime,

@SameDayOfSameWeekOfSameMonthOfLastYear as datetime

if @AsOfDate is null or @AsOfDate > cast(getdate() as date)

set @AsOfDate = cast(getdate() as date)

select

dateadd(week, -1, @AsOfDate) as SameDayOfLastWeek ,

dateadd(week, -datepart(dw, @AsOfDate) +1 , @AsOfDate) as SameDayOfSameWeekOfLastMonth,

dateadd(week,-52,@AsOfDate) as SameDayOfSameWeekOfSameMonthOfLastYear

end