**SQL Assignment 4**

1. Explain different types of views. Demonstrate with suitable examples

Ans. A View in SQL as a logical subset of data from one or more tables. Views are used to restrict data access. A View contains no data of its own but its like window through which data from tables can be viewed or changed. The table on which a View is based are called BASE Tables.

There are 2 types of Views in SQL: Simple View and Complex View. Simple views can only contain a single base table. Complex views can be constructed on more than one base table. In particular, complex views can contain: join conditions, a group by clause, a order by clause.

When user wants data or some columns from same table then simple view is used. For an example if there is employee table which has Employee\_num,Employee\_name,salary columnns and we just need to see Employee\_num,Employee\_name then user can create a simple view

1. What is the difference between function and stored procedure? Write syntax for creating functions and stored procedures.

Ans. Differences

* 1. The function must return a value but in Stored Procedure it is optional. Even a procedure can return zero or n values.
  2. Functions can have only input parameters for it whereas Procedures can have input or output parameters.
  3. Functions can be called from Procedure whereas Procedures cannot be called from a Function.
  4. Procedures cannot be utilized in a SELECT statement whereas Function can be embedded in a SELECT statement.

Syntax

* For creating Function

CREATE [OR REPLACE] FUNCTION function\_name [parameters]

RETURN return\_datatype; {IS, AS}

Declaration\_section <variable,constant> ;

BEGIN

Execution\_section

Return return\_variable;

EXCEPTION

exception section

Return return\_variable;

END

* For creating Store Procedures

CREATE [OR REPLACE] PROCEDURE procedure\_name (<Argument> {IN, OUT, IN OUT} <Datatype>,…)

IS

Declaration section<variable, constant> ;

BEGIN

Execution section

EXCEPTION

Exception section

END

1. What is an index in SQL? What are the different types of indexes in SQL?

Ans. A SQL index is used to retrieve data from a database very fast.

A SQL index is a quick lookup table for finding records users need to search frequently. An index is small, fast, and optimized for quick lookups. It is very useful for connecting the relational tables and searching large tables.

Two types of Indexes

* + - 1. **Clustered**

Clustered index is the type of indexing that established a physical sorting order of rows.Suppose you have a table Student\_info which contains ROLL\_NO as a primary key than Clustered index which is self created on that primary key will sort the Student\_info table as per ROLL\_NO. Clustered index is like Dictionary, in the dictionary sorting order is alphabetical there is no separate index page.

* + - 1. **Non – Clustered**

The Non-Clustered index is an index structure separate from the data stored in a table that reorders one or more selected columns. The non-clustered index is created to improve the performance of frequently used queries not covered by clustered index. It’s like a textbook, the index page is created separately at the beginning of that book.

1. Showcase an example of exception handling in SQL stored procedure.

Ans. Error handling in SQL Server gives us control over the Transact-SQL code. For example, when things go wrong, we get a chance to do something about it and possibly make it right again. SQL Server error handling can be as simple as just logging that something happened, or it could be us trying to fix an error. It can even be translating the error in SQL language because we all know how technical SQL Server error messages could get making no sense and hard to understand. Luckily, we have a chance to translate those messages into something more meaningful to pass on to the users, developers, etc

1. Create a SQL function to split strings into rows on a given character?

Input String: Stephen;peter;berry;Olivier;caroline;

|  |
| --- |
| Stephen |
| Peter |
| Berry |
| Oliver |
| Caroline |

Ans.

CREATE FUNCTION SplitString

(

@Input NVARCHAR(MAX),

@Character CHAR(1)

)

RETURNS @Output TABLE (

Item NVARCHAR(1000)

)

AS

BEGIN

DECLARE @StartIndex INT, @EndIndex INT

SET @StartIndex = 1

IF SUBSTRING(@Input, LEN(@Input) - 1, LEN(@Input)) <> @Character

BEGIN

SET @Input = @Input + @Character

END

WHILE CHARINDEX(@Character, @Input) > 0

BEGIN

SET @EndIndex = CHARINDEX(@Character, @Input)

INSERT INTO @Output(Item)

SELECT SUBSTRING(@Input, @StartIndex, @EndIndex - 1)

SET @Input = SUBSTRING(@Input, @EndIndex + 1, LEN(@Input))

END

RETURN

END

GO

1. What is a temporary and a variable table? Write suitable syntax to create temporary tables and variable tables.

Ans. Temporary Tables are physically created in the tempdb database. These tables act as the normal table and also can have constraints, index like normal tables.

Table Variable acts like a variable and exists for a particular batch of query execution. It gets dropped once it comes out of batch. It is created in the memory database but may be pushed out to tempdb.

Syntax

To create temporary table

CREATE TABLE #EmpDetails (id INT, name VARCHAR(25))

To create variable table

Declare @<Variable\_Name> TABLE(

Column\_Name [Data\_Type],

Column\_Name [Data\_Type],

Column\_Name [Data\_Type],

......)