

Difference between Stored procedures and User Defined functions[UDF]

Stored procedure

A stored procedure is a program (or procedure) which is physically stored within a database. They are usually written in a proprietary database language like PL/SQL for Oracle database or PL/PgSQL for PostgreSQL. The advantage of a stored procedure is that when it is run, in response to a user request, it is run directly by the database engine, which usually runs on a separate database server. As such, it has direct access to the data it needs to manipulate and only needs to send its results back to the user, doing away with the overhead of communicating large amounts of data back and forth.

User-defined function

A user-defined function is a routine that encapsulates useful logic for use in other queries. While views are limited to a single SELECT statement, user-defined functions can have multiple SELECT statements and provide more powerful logic than is possible with views.

In SQL Server 2000

User defined functions have 3 main categories

1. **Scalar-valued function** - returns a scalar value such as an integer or a timestamp.
Can be used as column name in queries
2. **Inline function** - can contain a single SELECT statement.
3. **Table-valued function** - can contain any number of statements that populate the table variable to be returned. They become handy when you need to return a set of rows, but you can't enclose the logic for getting this rowset in a single SELECT statement.

Differences between Stored procedure and User defined functions

1. UDF can be used in the SQL statements anywhere in the WHERE/HAVING/SELECT section where as Stored procedures cannot be.
2. UDFs that return tables can be treated as another rowset. This can be used in JOINS with other tables.
3. Inline UDF's can be thought of as views that take parameters and can be used in JOINS and other Rowset operations.
4. Of course there will be Syntax differences and here is a sample of that
Stored procedure

Code: SQL

```
CREATE PROCEDURE dbo.StoredProcedure1
/*
(
  @parameter1 datatype = default value,
  @parameter2 datatype OUTPUT
)
*/
AS
/* SET NOCOUNT ON */
RETURN
```

User defined functions

Code: SQL

```
CREATE FUNCTION dbo.Function1
(
/*
  @parameter1 datatype = default value,
  @parameter2 datatype
*/
)
RETURNS /* datatype */
AS
  BEGIN
    /* sql statement ... */
    RETURN /* value */
  END
```

Source: <http://www.go4expert.com/articles/difference-stored-procedures-user-t329/>