**Now let's understand the differences between Inline Table Valued functions and Multi-statement Table Valued functions**

1. In an Inline Table Valued function, the RETURNS clause cannot contain the structure of the table, the function returns.

Where as, with the multi-statement table valued function, we specify the structure of the table that gets returned

2. Inline Table Valued function cannot have BEGIN and END block, where as the multi-statement function can have.

3. Inline Table valued functions are better for performance, than multi-statement table valued functions. If the given task, can be achieved using an inline table valued function, always prefer to use them, over multi-statement table valued functions.

4. It's possible to update the underlying table, using an inline table valued function, but not possible using multi-statement table valued function.  
  
**Updating the underlying table using inline table valued function:**  
This query will change **Sam** to **Sam1**, in the underlying table **tblEmployees**. When you try do the same thing with the multi-statement table valued function, you will get an error stating 'Object 'fn\_MSTVF\_GetEmployees' cannot be modified.'  
Update fn\_ILTVF\_GetEmployees() set Name='Sam1' Where Id = 1  
  
**Reason for improved performance of an inline table valued function:**  
Internally, SQL Server treats an inline table valued function much like it would a view and treats a multi-statement table valued function similar to how it would a stored procedure.