In this video we will discuss Row\_Number function in SQL Server. This is continuation to [Part 108](http://csharp-video-tutorials.blogspot.com/2015/09/over-clause-in-sql-server.html). Please watch [Part 108](http://csharp-video-tutorials.blogspot.com/2015/09/over-clause-in-sql-server.html) from [SQL Server tutorial](https://www.youtube.com/playlist?list=PL08903FB7ACA1C2FB) before proceeding.   
  
   
  
**Row\_Number function**

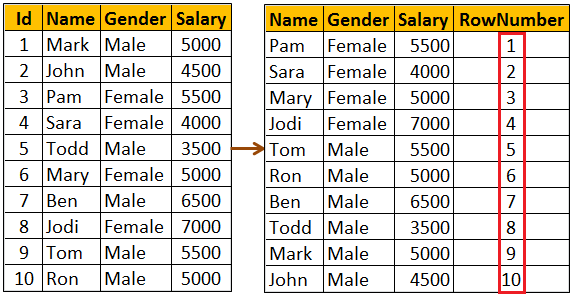
* Introduced in SQL Server 2005
* Returns the sequential number of a row starting at 1
* ORDER BY clause is required
* PARTITION BY clause is optional
* When the data is partitioned, row number is reset to 1 when the partition changes

**Syntax :** ROW\_NUMBER() OVER (ORDER BY Col1, Col2)

**Row\_Number function without PARTITION BY :** In this example, data is not partitioned, so ROW\_NUMBER will provide a consecutive numbering for all the rows in the table based on the order of rows imposed by the ORDER BY clause.

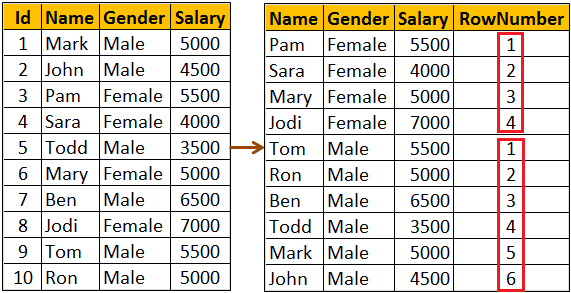
SELECT Name, Gender, Salary,

        ROW\_NUMBER() OVER (ORDER BY Gender) AS RowNumber

FROM Employees  
  
   
  
**Please note :** If ORDER BY clause is not specified you will get the following error  
The function 'ROW\_NUMBER' must have an OVER clause with ORDER BY  
  
**Row\_Number function with PARTITION BY :** In this example, data is partitioned by Gender, so ROW\_NUMBER will provide a consecutive numbering only for the rows with in a parttion. When the partition changes the row number is reset to 1.

SELECT Name, Gender, Salary,

        ROW\_NUMBER() OVER (PARTITION BY Gender ORDER BY Gender) ASRowNumber

FROM Employees  
  
   
  
**Use case for Row\_Number function :** Deleting all duplicate rows except one from a sql server table.   
  
Discussed in detail in[Part 4](http://csharp-video-tutorials.blogspot.com/2014/05/part-4-delete-duplicate-rows-in-sql.html) of [SQL Server Interview Questions and Answers video series](https://www.youtube.com/playlist?list=PL6n9fhu94yhXcztdLO7i6mdyaegC8CJwR).