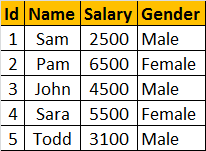
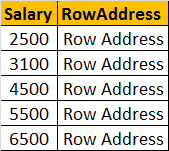
**Why indexes?**  
Indexes are used by queries to find data from tables quickly. Indexes are created on tables and views. Index on a table or a view, is very similar to an index that we find in a book.   
  
If you don't have an index in a book, and I ask you to locate a specific chapter in that book, you will have to look at every page starting from the first page of the book.   
  
On, the other hand, if you have the index, you lookup the page number of the chapter in the index, and then directly go to that page number to locate the chapter.   
  
Obviously, the book index is helping to drastically reduce the time it takes to find the chapter.   
  
In a similar way, Table and View indexes, can help the query to find data quickly.   
  
In fact, the existence of the right indexes, can drastically improve the performance of the query. If there is no index to help the query, then the query engine, checks every row in the table from the beginning to the end. This is called as Table Scan. Table scan is bad for performance.   
  
   
  
   
  
   
  
**Index Example:**At the moment, the Employees table, does not have an index on SALARY column.  
  
  
  
**Consider, the following query**  
Select \* from tblEmployee where Salary > 5000 and Salary < 7000  
  
To find all the employees, who has salary **greater than 5000 and less than 7000**, the query engine has to check each and every row in the table, resulting in a table scan, which can adversely affect the performance, especially if the table is large. Since there is no index, to help the query, the query engine performs an entire table scan.  
  
**Now Let's Create the Index to help the query:**Here, we are creating an index on Salary column in the employee table  
CREATE Index IX\_tblEmployee\_Salary   
ON tblEmployee (SALARY ASC)  
  
**The index stores salary of each employee, in the ascending order** as shown below. The actual index may look slightly different.  


**Now, when the SQL server has to execute the same query**, it has an index on the salary column to help this query. Salaries between the range of 5000 and 7000 are usually present at the bottom, since the salaries are arranged in an ascending order. SQL server picks up the row addresses from the index and directly fetch the records from the table, rather than scanning each row in the table. This is called as Index Seek.  
  
**An Index can also be created graphically using SQL Server Management Studio**

1. In the Object Explorer, expand the Databases folder and then specific database you are working with.   
2. Expand the Tables folder  
3. Expand the Table on which you want to create the index  
4. Right click on the Indexes folder and select New Index  
5. In the New Index dialog box, type in a meaningful name  
6. Select the Index Type and specify Unique or Non Unique Index  
7. Click the Add  
8. Select the columns that you want to add as index key  
9 Click OK  
10. Save the table   
  
**To view the Indexes**: In the object explorer, expand Indexes folder. Alternatively use sp\_helptext system stored procedure. The following command query returns all the indexes on tblEmployee table.  
**Execute sp\_helptext tblEmployee**   
  
**To delete or drop the index:**When dropping an index, specify the table name as well  
Drop Index tblEmployee.IX\_tblEmployee\_Salary