<https://www.youtube.com/watch?v=0GTeUHkdZtc>

select \* from employee where dept=”IT”   
If the more employees are there in IT then scan will happen. To make it seek   
select \* from employee where dept=”IT” option(optimize for (dept=”IT”))

Index Scan retrieves all the rows from the table.   
Index Seek retrieves selective rows from the table.

**Index Scan:**  
Since a scan touches every row in the table whether or not it qualifies, the cost is proportional to the total number of rows in the table. Thus, a scan is an efficient strategy if the table is small or if most of the rows qualify for the predicate.

**Index Seek:**  
Since a seek only touches rows that qualify and pages that contain these qualifying rows, the cost is proportional to the number of qualifying rows and pages rather than to the total number of rows in the table.

If there is no index, then you might see a Table Scan (Index Scan) in the execution plan.

Index seeks are generally preferred for the highly selective queries. What that means is that the query is just requesting a fewer number of rows or just retrieving the other 10 (some documents says 15 percent) of the rows of the table.

In general query optimizer tries to use an Index Seek which means that optimizer has found a useful index to retrieve recordset. But if it is not able to do so either because there is no index or no useful indexes on the table then SQL Server has to scan all the records that satisfy query condition.