Decision Factors on the SQL Statements for Indexes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SQL Statements | | Index Type | Table | Column(S) | Fill Factor |
| 1 | The column(s) in the where statement | Both index types | ProductTBL | ProductPrice  Clustered 🡪 Support the range predicate |  |
| *1a* | *The selectivity[[1]](#footnote-1) of the where statement* | *Both index types* | ProductTBL | ProductPrice |  |
| 2 | The column(s) in the join statements | Both index types (FK prefer to use a clustered index) | OrderDetailsTBL | ProductID 🡪 Clustered support the join |  |
| 3 | The column(s) in the order by statement | Clustered index only | None |  |  |
| 4 | The column(s) in the group by statement | Clustered index only | None |  |  |
| 5 | Covering index 🡪 index on the columns in the where statement, **include** other columns that are in the select statement, put a **filter** to match the where statement. | Non clustered only | Include 🡪  Filter 🡪 | N/A |  |

Decision Factors on the tables statistics for Indexes

|  |  |  |
| --- | --- | --- |
| 1 | Is the table a big table |  |
| 2 | Data density 🡪 the percentage of unique rows in the column(s) in the table |  |
| 3 | Distribution of the data in the column(s) in the table |  |
| 4 | Column width 🡪 too wide too much processing to maintain the index |  |

1. The percentage of rows in the table that this where statement is likely to return – High selectivity means a small number of rows are returned so this is when an index is useful as it avoids a full table scan [↑](#footnote-ref-1)