3.)
Customer to Customer_index: 4ms
Customer to Customer_noindex: 0ms

Supplier to Supplier_index: 5ms
Supplier to Supplier_noindex: 0ms

Orders to Orders_index: 106ms
Orders to Orders_noindex: 11ms

Lineitem to Lineitem_index: 166ms
Lineitem to lineitem_noindex: 35ms

Using the index, it has to copy all the data down and then go through it with the index position making it run longer then the table with no index

4.)
CREATE INDEX customer_mktsegment_index ON customer_index(c_mktsegment); = 2ms
CREATE INDEX lineitem_returnflag_index ON lineitem_index(l_returnflag); = 37ms
CREATE INDEX orders_priority_index ON orders_index(o_orderpriority); = 14ms

We can see that with more data inside the column it will take longer to process the index

5.)
Q1.)
customer_Index: 1ms
Customer_noindex: 2ms

Q2.)
Supplier_index: 0ms
Supplier_noindex: 1ms

Q3.)
Lineitem_index: 32ms
Lineitem_noindex: 34ms

Q4.)
Lineitem_index: 99ms
Lineitem_noindex: 112ms.

Q5.)
customer_Index: 2ms
Customer_noindex: 3ms

Q6.)
Supplier & order & customer_index: 1867ms
Supplier & order & customer_noindex: 2008ms

Q7.)
Supplier & order_index: 1886ms
Supplier & order_noindex: 1540ms

Q8.)
Supplier_index: 1ms
Supplier_noindex: 2ms

Q9.)
Supplier_index: 0ms
Supplier_noindex: 2ms

Q10.)
Customer & order_index: 90ms
Customer & order_noindex: 94ms

Q11.)
lineitem & order & customer_index: 438ms
lineitem & order & customer_noindex: 297ms

Q12.)
Customer & order_index: 61ms
Customer & order_noindex: 69ms

Q13.)
customer_Index: 3ms
Customer_noindex: 4ms

Q14.)
Customer & order_index: 22ms
Customer & order_noindex: 35ms

Q15.)
Supplier & order_index: 2100ms
Supplier & order_noindex: 1660ms

Analyzing the runtimes of the queries, we can see that the queries running the indexes will run faster, but there seemed to be some strange occurances. When there is more than one index being used in query, there was an increase in time compared to the queries that did not run with indexes. When it comes to running queries with only the use of the index, or less joint tables, or less data in general, the queries will run faster with the indexes.

6.)
Lineitem_index: 234ms
Lineitem_noindex: 103ms

With the update, since the index will scan through the data and through the position, updating will take longer with the index.

7.)
Supplier_index: 0 ms
Supplier_noindex: 0ms

There seems to be no change when deleting, I tried doing bigger updates, and multiple runs of the updates but there was no change in either runtimes.