ASSIGNMENT #4 REPORT

Student Name and CCID:

A. ZiQing Ma, 1499476

By submitting this assignment, the students named above confirm that they have worked on it themselves without any help by other people. If any external resources were used please state which ones and how they were used:

PART 1

Task A (1) (no index):

Cardinality of Table Parts	**Average Processing time for index free Q1 (ms)**	**Personal Machine Processing Time (ms)**
100	1.55023	0.05950
1000	2.94600	0.11903
10,000	4.04902	0.64480
100,000	24.63741	7.39038
1,000,000	244.58351	74.10245

Cardinality of Table Parts	**Average Processing time for index free Q2 (ms)**	**Personal Machine Processing Time (ms)**
100	1.84650	0.07938
1000	3.18230	0.16863
10,000	4.14389	1.24000
100,000	28.33202	14.18562

1,000,000	283.28395	109.31840

Task B (2):

Compare, contrast, and explain the trends observable in both tables above (Task A)
The average processing time increases proportionally with the size of the database. This is expected behavior since the queries must read all the rows in the table. The average processing time for Q2 is higher than Q1 for the corresponding instance of the database. This is because needsPart is not a primary key like partNumber which is unique.

Task C (3) (using index):

Cardinality of Table Parts	**Average Processing time for indexed Q1 (ms)**	**Personal Machine Processing Time (ms)**
100	1.84105	0.05952
1000	2.94978	0.11905
10,000	3.59469	0.66464
100,000	22.77260	6.00119
1,000,000	243.17923	70.82877

Cardinality of Table Parts	**Average Processing time for indexed Q2 (ms)**	**Personal Machine Processing Time (ms)**
100	1.83255	0.06944
1000	2.91430	0.10911
10,000	3.67164	0.65472

100,000	22.99454	5.95200
1,000,000	235.78401	70.73655

Task D (4):

Compare, contrast, and explain the trends observable in both tables above (Task C)
Similar to Task A, the average processing time increases proportionally with size of the database again for both queries. However, the processing times for both queries are about equivalent now after indexing. Interestingly, since both have similar times suggests that indexing helped Q2 more than it helped Q1. This is due to the fact that needsPart was indexed and equality searches are done on needsPart in Q2.

Task E (5):

Compare, contrast, and explain the trends observed in Task D to the trends observed in Task B. Discuss the cost-benefit of the index space cost and query performance.
A4v100: 8KB ==> 12KB A4v1k: 32KB ==> 56KB A4v10k: 236KB ==> 416KB A4v100k: 2MB ==> 3MB A4v1M: 21MB ==> 39MB The processing times were significantly less for both queries after indexing. However, the times obtained from Q1 were slightly less than before indexing. In conclusion, the benefits of an index are not worth the space cost for Q1, but much more worthwhile for Q2.

PART 2

Task F (6) (no index):

Cardinality of Table Parts	Average Processing time for index-free Q3 (ms)	Personal Machine Processing Time (ms)
100	1.47173	0.05952
1000	2.74579	0.20832
10,000	4.00952	1.76577
100,000	27.86391	24.55199
1,000,000	379.91939	240.26260

Task G (7) (using index):

Cardinality of Table Parts	Average Processing time for indexed Q3 (ms)	Personal Machine Processing Time (ms)
100	1.64415	0.04959
1000	1.70196	0.03968
10,000	1.30629	0.04960
100,000	1.46978	0.29709
1,000,000	241.57815	8.92797

Task H (8):

Compare, contrast, and explain the trends observed in Task F to the trends observed in Task G. Discuss the cost-benefit of the index space cost and query performance.
A4v100: 8KB ==> 12KB A4v1k: 32KB ==> 52KB A4v10k: 236KB ==> 308KB

A4v100k: 2MB ==> 3MB
A4v1M: 21MB ==> 35MB

The index drastically increased performance as the size of the database increases. This is because madeIn is indexed and hence easier for GROUP BY to search. For example, on my machine, the average processing time went from 240 ms to ~9 ms for cardinality of 1M after indexing. The index is definitely worth the cost as the performance/space ratio is so high.

PART 3

Task I (9) (no index):

Cardinality of Table Parts	**Average Processing time for no-index Q4 (ms)**	**Personal Machine Processing Time (ms)**
100	1.68579	0.06944
1000	3.54705	0.19839
10,000	5.02722	1.30942
100,000	33.32946	15.07838
1,000,000	342.88874	153.36318

Task J (10):

Define an index that you believe will optimize Q4 and explain why you think so.
CREATE INDEX idxMadeIn ON Parts (madeIn); Improves performance by indexing madeIn because of equality search in the WHERE clause and within the subquery.

Task K (11) (using index):

Cardinality of Table Parts	Average Processing time for indexed Q4 (ms)	Personal Machine Processing Time (ms)
100	2.02020	0.05952
1000	3.00232	0.11990
10,000	3.79922	0.66456
100,000	23.46678	7.48966
1,000,000	454.92272	75.59056

Task L (12):

Compare, contrast, and explain the trends observed in Task K to the trends observed in Task I. Discuss the cost-benefit of the index space cost and query performance.
A4v100: 8KB ==> 12KB A4v1k: 32KB ==> 52KB A4v10k: 236KB ==> 380KB A4v100k: 2MB ==> 3MB A4v1M: 21MB ==> 35MB Similar to other tasks without indexing, the average processing time increases proportionally with the size of the database.

PART 4

Task M (13) (no index):

Cardinality of Table Parts	Average Processing time for index-free Q5 (ms)	Personal Machine Processing Time (ms)
100	2.54872	0.40671

1000	43.25018	35.31521
10,000	4063.12886	3545.36061
100,000	N/A	N/A
1,000,000	N/A	N/A

Task N (14) (no index):

Cardinality of Table Parts	**Average Processing time for index-free Q6 (ms)**	**Personal Machine Processing Time (ms)**
100	1.62705	0.06946
1000	2.17791	0.46624
10,000	8.24502	5.53535
100,000	81.09519	73.60644
1,000,000	2186.18197	3453.42236

Task O (15):

Compare, contrast, and explain the trends observed in Task M to the trends observed in Task N
Similar to other tasks without indexing, the average processing time increases proportionally with the size of the database.

Task P (16):

Define an index that you believe will optimize Q6 and explain why you think so
CREATE INDEX idxNeedsPart ON Parts (needsPart);

Task Q (17) (with index):

Cardinality of Table Parts	**Average Processing time for indexed Q6 (ms)**	**Personal Machine Processing Time (ms)**
100	1.79729	0.05953
1000	1.69155	0.20757
10,000	3.97082	2.10303
100,000	38.03020	31.09918
1,000,000	1934.24807	3730.82743

Task R (18):

Compare, contrast, and explain the trends observed in Task N to the trends observed in Task Q. Discuss the cost-benefit of the index space cost and query performance.
A4v100: 8KB ==> 12KB A4v1k: 32KB ==> 56KB A4v10k: 236KB ==> 416KB A4v100k: 2MB ==> 3MB A4v1M: 21MB ==> 39MB Index does not seem worthwhile once cardinality is 1M.