Title

Report #6

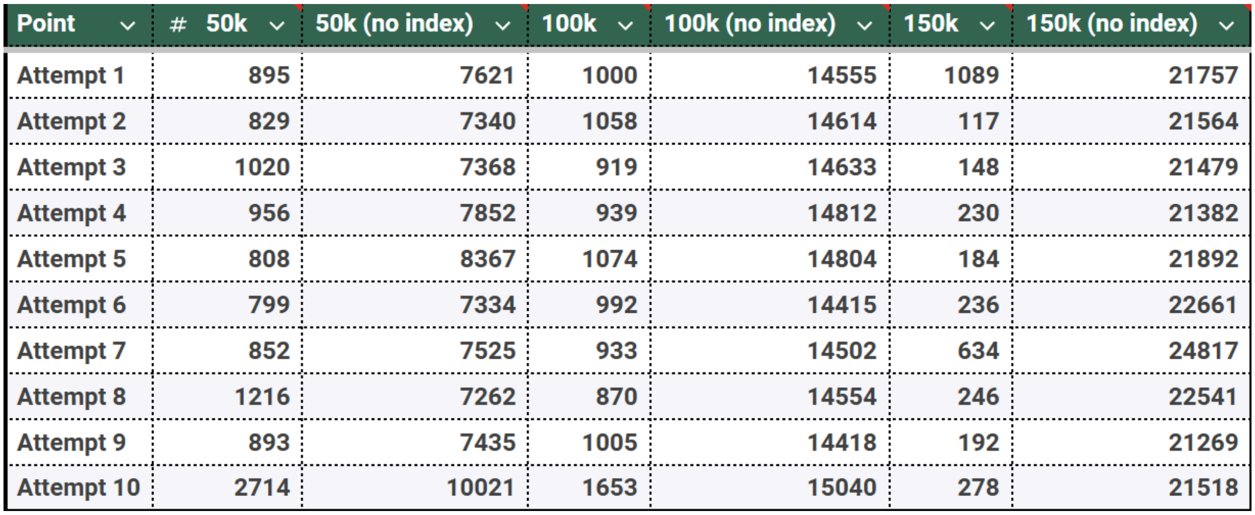
- **Title:** DB Assignment 6
- **Your Name:** Antonio J. Cima
- **Date:** 12/13/2024

As a side note, again sorry for the delay, I got really busy with finals and thought I would have more time to complete this assignment, nonetheless, this all should work fingers crossed!

Information

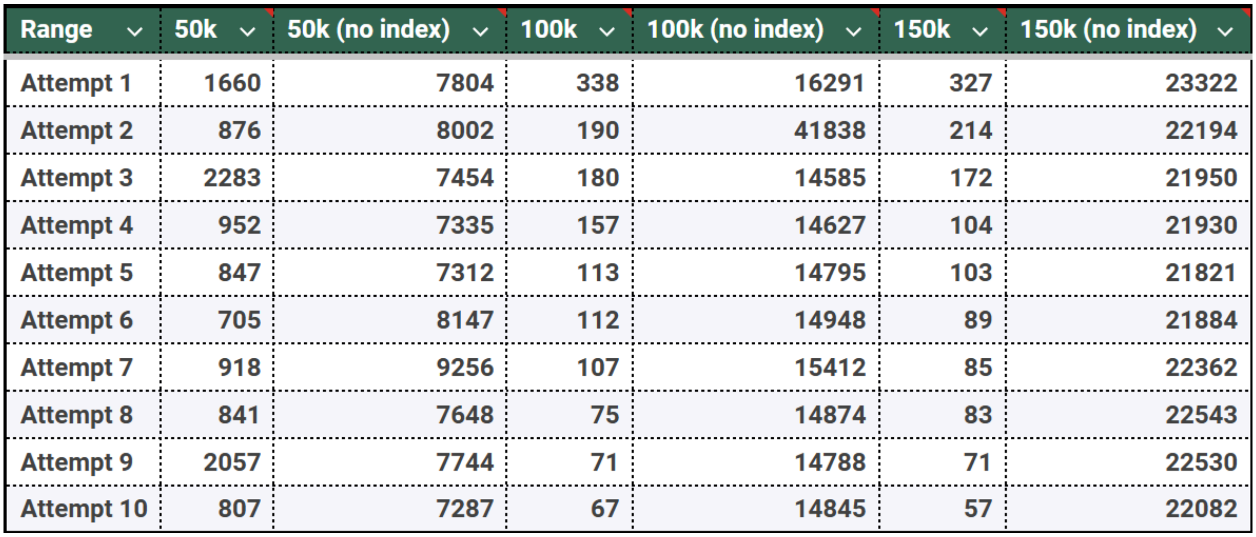
Below is the time (in microseconds) for our point queries, we did queries with and without indexes (the indexed queries being the ones that just have plain text while no-indexed queries are the ones with the parentheses stating so). The queries we ran for the Point queries were:

SELECT count(*) FROM accounts FORCE INDEX (idx_branch_name, idx_balance) WHERE branch_name = 'Downtown' AND balance = 50000;



Point	# 50k	50k (no index)	100k	100k (no index)	150k	150k (no index)
Attempt 1	895	7621	1000	14555	1089	21757
Attempt 2	829	7340	1058	14614	117	21564
Attempt 3	1020	7368	919	14633	148	21479
Attempt 4	956	7852	939	14812	230	21382
Attempt 5	808	8367	1074	14804	184	21892
Attempt 6	799	7334	992	14415	236	22661
Attempt 7	852	7525	933	14502	634	24817
Attempt 8	1216	7262	870	14554	246	22541
Attempt 9	893	7435	1005	14418	192	21269
Attempt 10	2714	10021	1653	15040	278	21518

Below is our Range queries, same thing as above, we ran them 10 times and these are our results. *T*he query we used for this was: *WHERE branch_name = 'Downtown' AND balance BETWEEN 10000 AND 5000;*



Range	50k	50k (no index)	100k	100k (no index)	150k	150k (no index)
Attempt 1	1660	7804	338	16291	327	23322
Attempt 2	876	8002	190	41838	214	22194
Attempt 3	2283	7454	180	14585	172	21950
Attempt 4	952	7335	157	14627	104	21930
Attempt 5	847	7312	113	14795	103	21821
Attempt 6	705	8147	112	14948	89	21884
Attempt 7	918	9256	107	15412	85	22362
Attempt 8	841	7648	75	14874	83	22543
Attempt 9	2057	7744	71	14788	71	22530
Attempt 10	807	7287	67	14845	57	22082

Report

Results of the point query were as expected, low with indexing and high w/out indexing, and it scaled as expected with the higher volume of items in the lists. The unique twist on all this was the Range queries, I genuinely expected this to take longer, seeing as there was another parameter, but with indexing, the overall average of range queries was faster than point queries. Interestingly enough as well, the average speed of range queries w/out indexing was SLOWER than point queries, so it seems there are benefits and disadvantages to each.

So what I have gathered from this; indexing saves a lot of time compared to not using indexing at all, but more importantly, if you are NOT indexing, range queries will be much slower than just doing a standard point query.