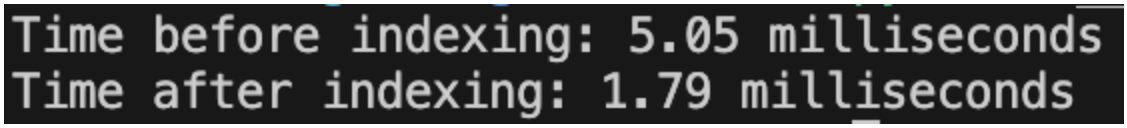
Part 3

Query Optimization and Indexing

We've recently worked on optimizing our MongoDB queries, particularly focusing on our `playlists` collection. The goal was to speed up how quickly we could access data, especially since our app often pulls playlists based on their names. Initially, the queries weren't too sluggish, but we knew there was room for improvement.



After setting up an index on the `name` field, we saw a significant change. Before indexing, fetching playlists starting with a letter (like "S") took about 5.05 milliseconds. Not bad, but we wanted to see if we could do better. Post-indexing, the same query took just 1.79 milliseconds. That's a substantial speed boost, cutting down the time by more than half!

This improvement is a big win for us. Faster queries mean a smoother experience for users, and it shows the power of effective indexing in MongoDB. It's pretty amazing to see how a relatively simple change can have such a noticeable impact on performance.