**PART: 3 Query Processing and Optimization Techniques**

* *Ajith Kumar Panja Umasankar (1225467126)*

1. **Query Optimization:**

i) Query to list the match id, team names and the players playing for the match

**Unoptimized:**

A screenshot of a computer

Description automatically generated

**Optimized:**

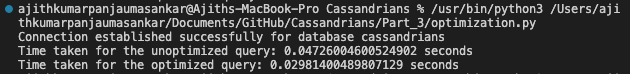
A screenshot of a computer

Description automatically generated

The Unoptimized query which uses Cartesian product took almost 65 msec, whereas the optimized query with joins took almost half of that ~39 msec to provide the same result with same number of rows.

Our Python code using psycopg2 library seems to be prove the same observance we found in PgAdmin.

Optimized code is taking lesser time than the unoptimized one.



1. **Distributed Indexing:**

Query to print the ball by ball data by joining all the tables

**Before Indexing:**

**A screenshot of a computer program

Description automatically generated**

**Index Creation:**

A screenshot of a computer

Description automatically generated

**After Indexing:**

A screenshot of a computer program

Description automatically generated

Indexing decreased the time taken by ~30 secs.

Our Python code using psycopg2 library seems to be prove the same observance we found in PgAdmin.

