**Final Project Part-2**

**Vinayaka S Gadag**

**CYPHER SCRIPT**

**// Load file into database**

LOAD CSV WITH HEADERS FROM 'file:///Users/vinayakgadag/IU/FirstSemester/SQL-NOSQL/neo4j-project/out.csv' AS row // load csv

WITH row // accessible object

**// Create nodes, set properties, and merge**

MERGE(grandprix:GrapndPrix{name:row.GPName,date:row.GPDate,year:row.Year})

MERGE(driver:Driver{name:row.Driver,notionality:row.DriverNationality,dob:row.DriverDOB,driverWin:row.DriverWin})

MERGE(circuit:CircuitName{name:row.CircuitName,location:row.CircuitLocation,country:row.CircuitCountry})

MERGE(result:Result{id:row.resultId})

MERGE(points:Points{driverPoints:row.points,constructorPoints:row.ConstructorPoints})

MERGE(rank:Rank{driverRank:row.rank,constructorRank:row.ConstructorRank})

MERGE(constructor:Constructor{name:row.Constructor,country:row.ConstructorNationality})

**// Create relationships between nodes**

MERGE (driver)-[:DRIVES]->(constructor)

MERGE (driver)-[:PARTICIPATES\_IN]->(grandprix)

MERGE (grandprix)-[:HAS\_CIRCUITS]->(circuit)

MERGE (grandprix)-[:RESULT]->(result)

MERGE (driver)-[:HAS\_POINTS]->(points)

MERGE (driver)-[:RANKED\_AT]->(rank)

MERGE (driver)-[:WON{resultid:row.resultId}]->(result)

MERGE (constructor)-[:CONSTRUCTOR\_RANKED\_AT]->(rank)

MERGE (constructor)-[:CONSTRUCTOR\_POINTS]->(points)

MERGE (driver)-[:PART\_OF]->(circuit)

**// Create Indexes**

CREATE INDEX driver For (d: Driver) ON (d.driver);

CREATE INDEX country For (c:Country) ON (c.country);

CREATE INDEX grandprix For (gp:GrapndPrix) ON (gp.grandprix);

CREATE INDEX circuit For (cir:CircuitLocation) ON (cir.circuit);

CREATE INDEX resultid For (rd:resultId) ON (rd.resultid);

CREATE INDEX circuitname For (cn:CircuitName) ON (cn.circuitname);

**//Create unique constraint**

CREATE CONSTRAINT ON (rd:resultId) ASSERT rd.resultId IS UNIQUE;

**Details and Screenshots of the database:**

The database has a total of seven nodes and ten unique relationships. Namely,

Nodes: GrandPrix, Driver, CircuitName, Result, Points, Rank, Constructor

Relationships: DRIVES, PARTICIPATES\_IN, HAS\_CIRCUITS, RESULT, HAS\_POINTS, RANKED\_AT, CONSTRUCTOR\_RANKED\_AT, CONSTRUCTOR\_POINTS, PART\_OF, WON.

The data has been reduced to 3500 records and 17 columns or attributes for the simplicity.

Each node has its properties. In the below screenshots, the graph represents the drivers participating in the GrandPrix tournament.

Chart

Description automatically generated with medium confidence

Diagram

Description automatically generated with low confidence

**REFERENCES**

* Week-12 course Workshop
* Neo4j Cypher manual: <https://neo4j.com/docs/cypher-manual/current/>