ADT Coding Practice

# Week 10

## Dipak Bange

1. Look at the first 5 lines

LOAD CSV WITH HEADERS FROM "file:///adt-sping23\week\_10\flights\_10k.csv" AS row

RETURN row

LIMIT 5

1. Create nodes and constraints

CREATE CONSTRAINT FOR (a:Airport)

REQUIRE a.code IS UNIQUE;

1. Import nodes and relationships

LOAD CSV WITH HEADERS FROM " file:///adt-sping23\week\_10\flights\_10k.csv " AS row

MERGE (origin:Airport {code: row.Origin})

MERGE (destination:Airport {code: row.Dest})

WITH row.UniqueCarrier + row.FlightNum + "\_" + row.Year + "-" + row.Month + "-" + row.DayofMonth + "\_" + row.Origin + "\_" + row.Dest AS flightIdentifier, row, origin, destination

MERGE (flight:Flight { id: flightIdentifier })

ON CREATE SET flight.date = row.Year + "-" + row.Month + "-" + row.DayofMonth,

flight.airline = row.UniqueCarrier, flight.number = row.FlightNum, flight.departure = row.CRSDepTime,

flight.arrival = row.CRSArrTime, flight.distance = row.Distance, flight.cancelled = row.Cancelled

MERGE (flight)-[:ORIGIN]->(origin)

MERGE (flight)-[:DESTINATION]->(destination)

1. Check your import

MATCH (:Flight)

RETURN count(\*)

1. Check schema

call db.schema.visualization()

1. Coercing values: Integers

MATCH (flight:Flight)

SET flight.distance = toInteger(flight.distance)

# Test the integer

MATCH (flight:Flight)

WHERE flight.distance > 500

RETURN flight

1. Coercing values: Boolean

MATCH (flight:Flight)

SET flight.cancelled = CASE WHEN flight.cancelled = "1" THEN true ELSE false END

# Test the Boolean

MATCH (flight:Flight)

WHERE flight.cancelled

RETURN flight

1. Test the boolean

PROFILE

MATCH path = (origin:Airport {code: "LAS"})<-[:ORIGIN]-(flight:Flight)-[:DESTINATION]->(destination:Airport {code: "MDW"})

WHERE flight.date = "2008-1-3"

RETURN path

1. Add nee relationship

# Add CONNECTED\_TO

:auto USING PERIODIC COMMIT

LOAD CSV WITH HEADERS FROM " file:///adt-sping23\week\_10\flights\_10k.csv " AS row

MERGE (origin)-[connection:CONNECTED\_TO {

airline: row.UniqueCarrier,

flightNumber: row.FlightNum,

date: toInteger(row.Year) + "-" + toInteger(row.Month) + "-" + toInteger(row.DayofMonth)}]->(destination)

ON CREATE SET connection.departure = toInteger(row.CRSDepTime), connection.arrival = toInteger(row.CRSArrTime)

# Count Connections

MATCH ()-[:CONNECTED\_TO]->()

RETURN COUNT(\*)

1. Search

MATCH path = (origin:Airport {code: "LAS"})<-[:ORIGIN]-(flight:Flight)-[:DESTINATION]->(destination:Airport {code: "MDW"})

WHERE flight.date = "2008-1-3"

RETURN path