DBT ASSIGNMENT 4

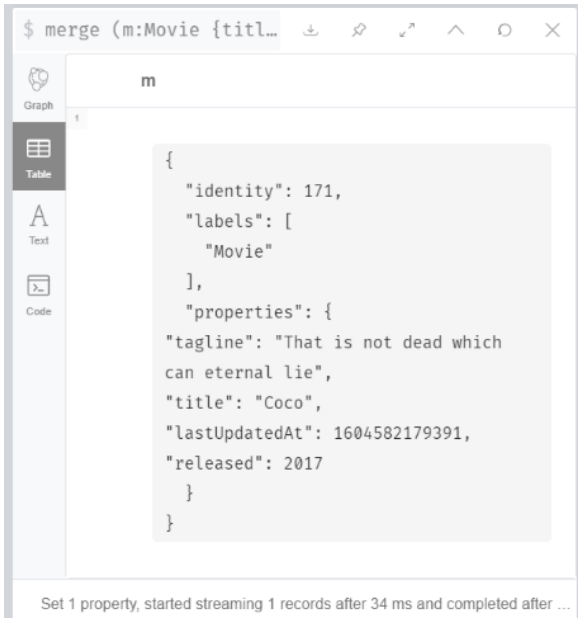
Sneha Hegde

PES1201801157

3. Neo4j-

- Establishing a new Movie node (Coco) and setting timestamp value-
  create (m:Movie
  {title:"Coco",tagline:"That is not dead which can eternal lie",released:2017})
  return m

  set timestamp value-
  merge (m:Movie {title:"Coco"})
  on match set m.lastUpdatedAt = timestamp()
  on create set m.lastUpdatedAt = timestamp()
  return m



- Establishing new Person nodes-

  create (p:Person
  {name:"Anthony Gonzalez"})
  return p

  set timestamp value-
  merge (p:Person {name:"Anthony Gonzalez"})
  on match set p.lastUpdatedAt = timestamp()
  on create set p.lastUpdatedAt = timestamp()
  return p

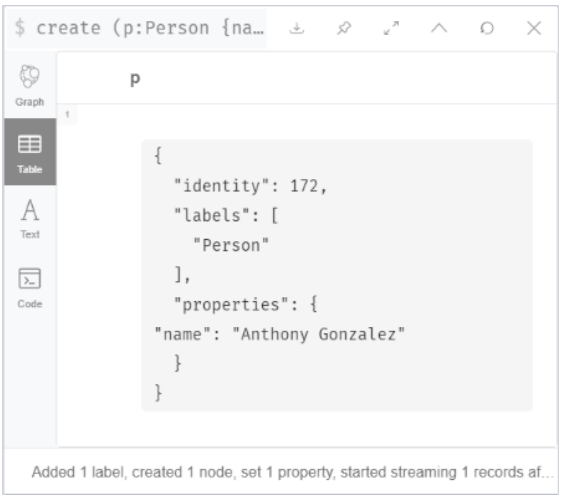
  create (p:Person
  {name:"Benjamin Bratt"})
  return p

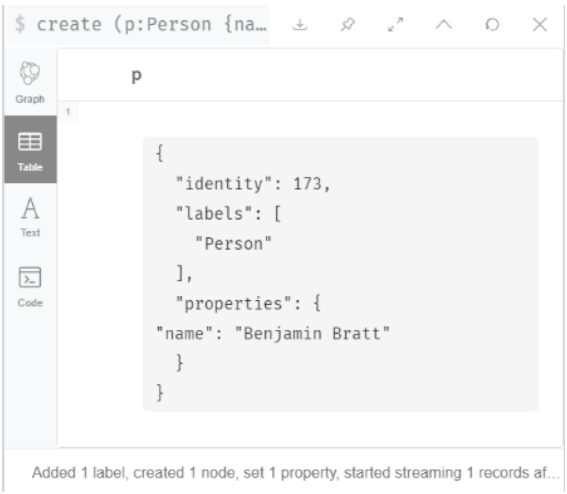
  set timestamp value-
  merge (p:Person {name:"Benjamin Bratt"})
  on match set p.lastUpdatedAt = timestamp()
  on create set p.lastUpdatedAt = timestamp()
  return p

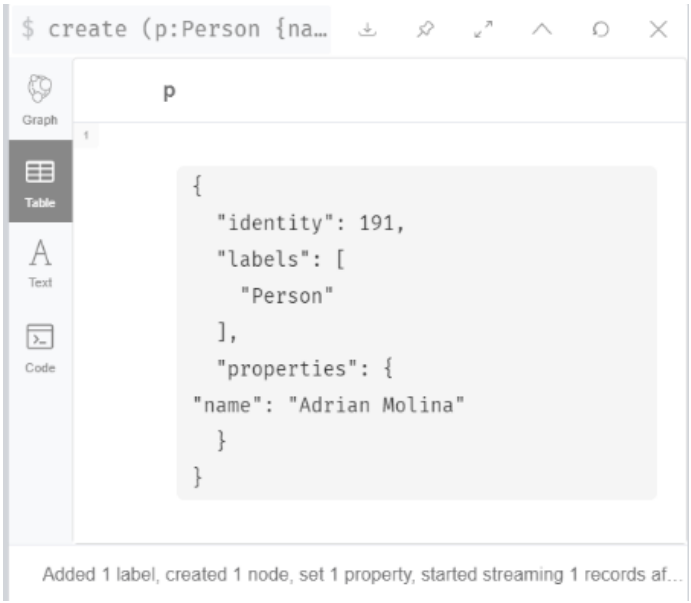
  create (p:Person
  {name:"Adrian Molina"})

return p

set timestamp value-
merge (p:Person {name:"Adrian Molina"})
on match set p.lastUpdatedAt = timestamp()
on create set p.lastUpdatedAt = timestamp()
return p







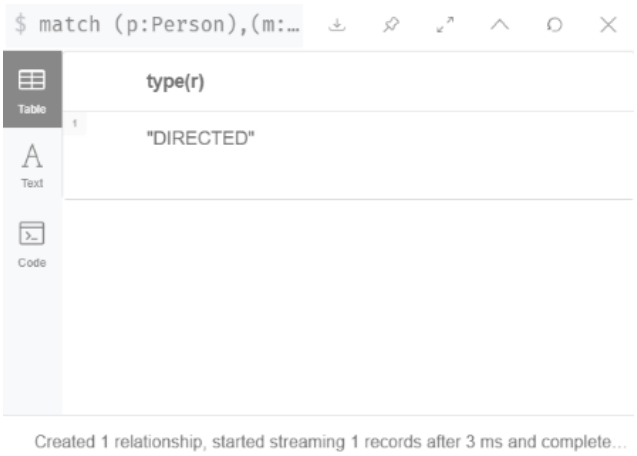
- Associating these two new Person nodes (Anthony Gonzalez, Benjamin Bratt) with the Movie node (Coco) through the relationship ACTED_IN-

  ```
  match (p:Person),(m:Movie)
  where p.name="Anthony Gonzalez" and m.title="Coco"
  create (p)-[r:ACTED_IN]->(m)
  return type(r)

  match (p:Person),(m:Movie)
  where p.name="Benjamin Bratt" and m.title="Coco"
  create (p)-[r:ACTED_IN]->(m)
  return type(r)
  ```

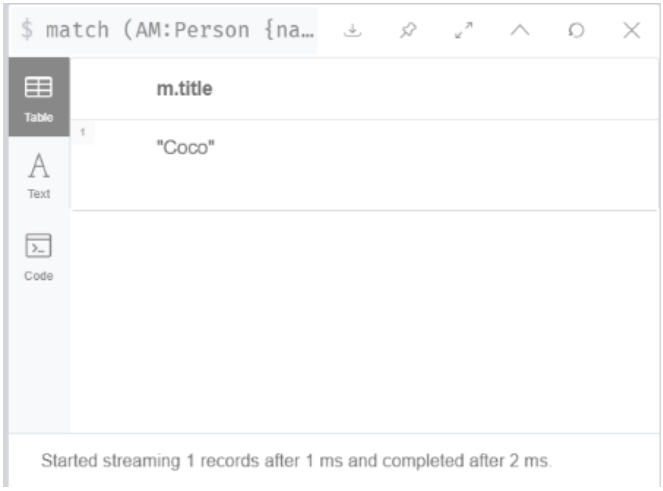
- Associating the third Person node (Adrian Molina) with the Movie node (Coco) through the relationship DIRECTED-

  ```
  match (p:Person),(m:Movie)
  where p.name="Adrian Molina" and m.title="Coco"
  create (p)-[r:DIRECTED]->(m)
  return type(r)
  ```

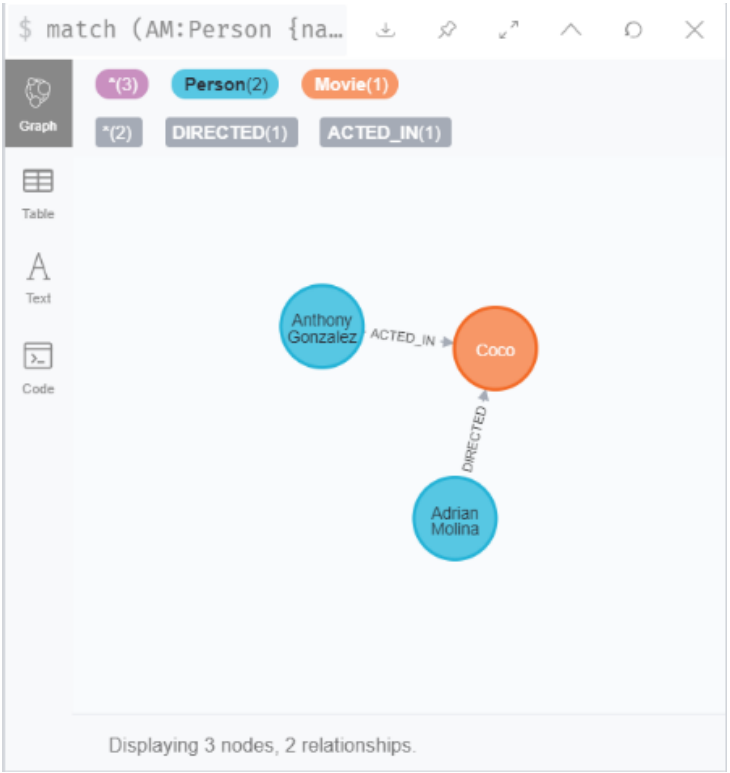


- Return the name of the movie which Adrian Molina directed and Anthony Gonzalez acted in (It should give Coco)-

  match (AM:Person {name:"Adrian Molina"})-[d:DIRECTED]->(m:Movie)<-[a:ACTED_IN]-(p:Person {name:"Anthony Gonzalez"}) return m.title

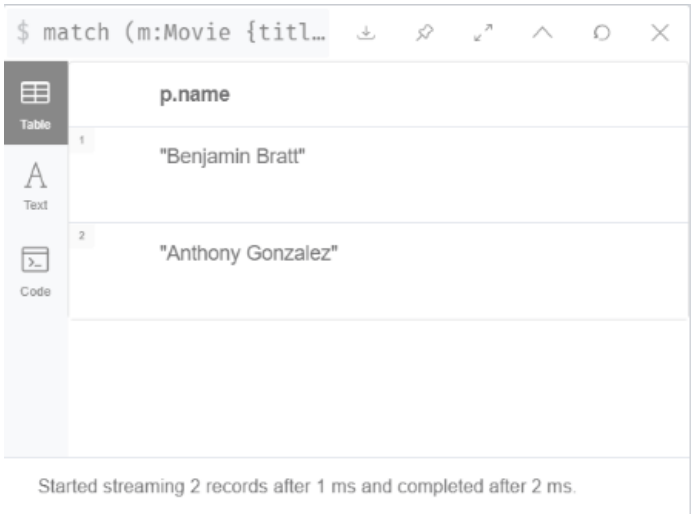


  match (AM:Person {name:"Adrian Molina"})-[d:DIRECTED]->(m:Movie)<-[a:ACTED_IN]-(p:Person {name:"Anthony Gonzalez"}) return AM,d,m,a,p



- Return the names of the people who acted in Coco (based on what we've entered)-

  match (m:Movie {title:"Coco"})<-[a:ACTED_IN]-(p:Person)
  return p.name



- Return the number of people who acted in Coco (based on what we've entered)-

  match (m:Movie {title:"Coco"})<-[a:ACTED_IN]-(p:Person)
  return count(*)