

DBT Assignment-4

ASHWIN KRISHNA P

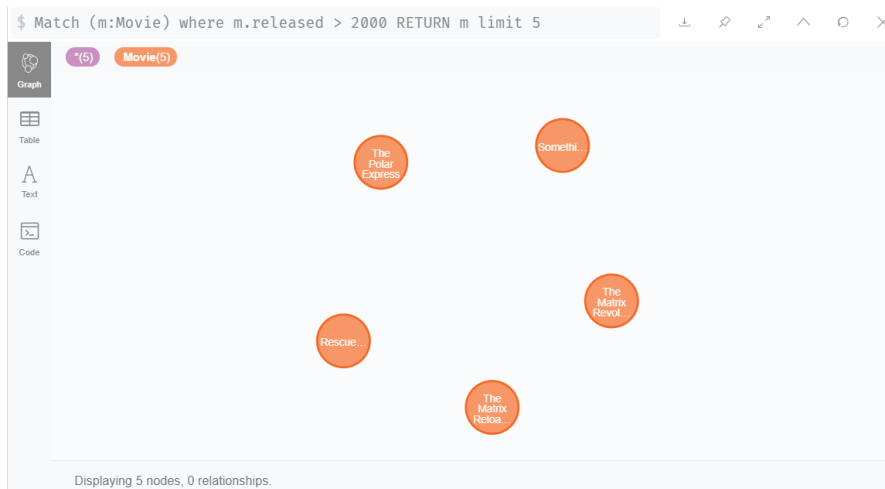
PES1201801465

SEM: 5 , SECTION: F

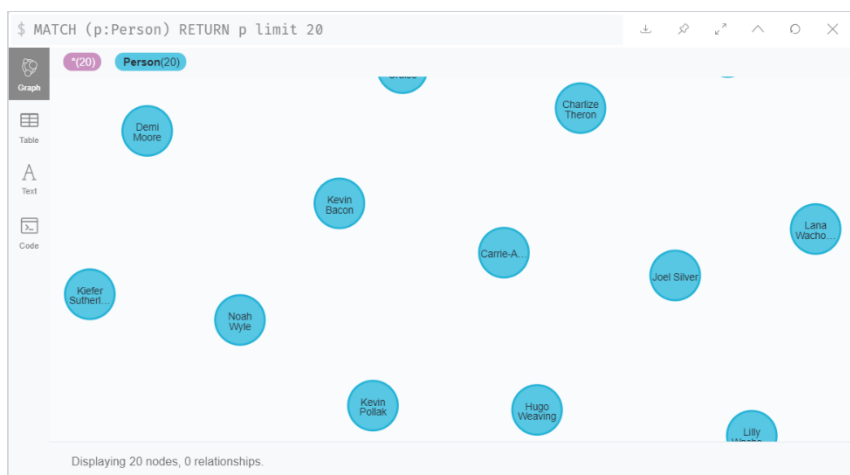
Neo4j (Graph Database)

All the operations are done on Movies database in Neo4j sandbox.

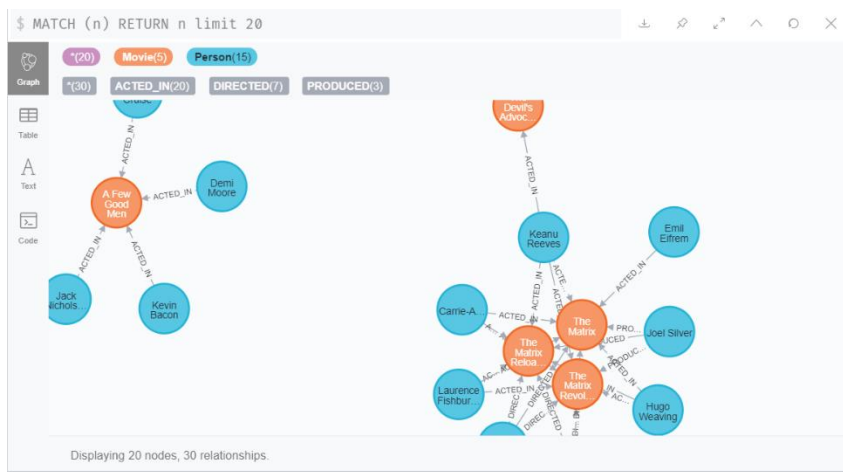
Query to return all the movies that were released after the year 2000 limiting the result to 5 items:



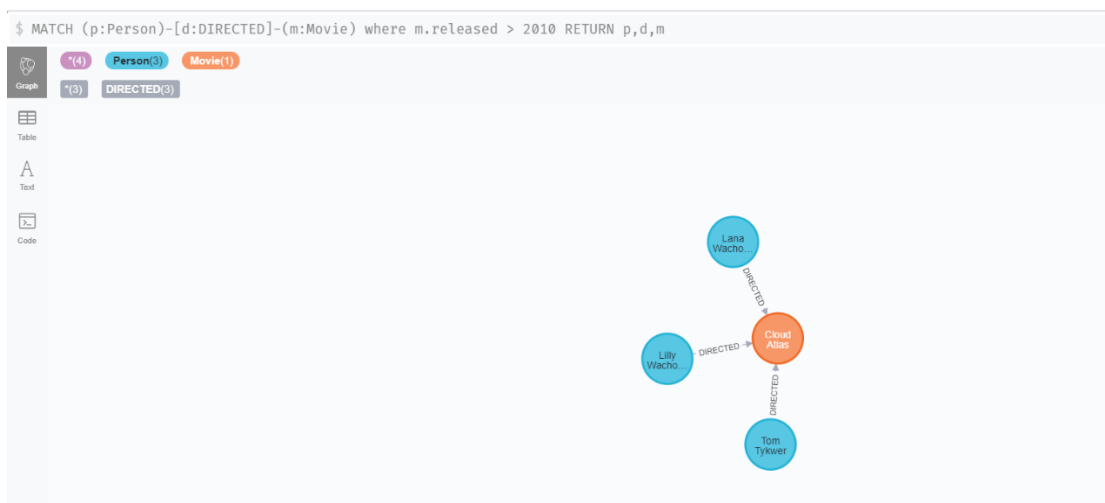
Query to return only person nodes , limiting nodes to 20:



Query to return all nodes , limiting nodes to 20:



To return all Person nodes who directed a movie that was released after 2010:



To return Movie nodes but with only the title and released properties:

Cypher query: `$ MATCH (m:Movie) return m.title,m.released`

	m.title	m.released
1	"The Matrix"	1999
2	"The Matrix Reloaded"	2003
3	"The Matrix Revolutions"	2003
4	"The Devil's Advocate"	1997
5	"A Few Good Men"	1992
6	"Top Gun"	1986
7	"Lenny Henrie"	2000

Started streaming 38 records after 2 ms and completed after 6 ms.

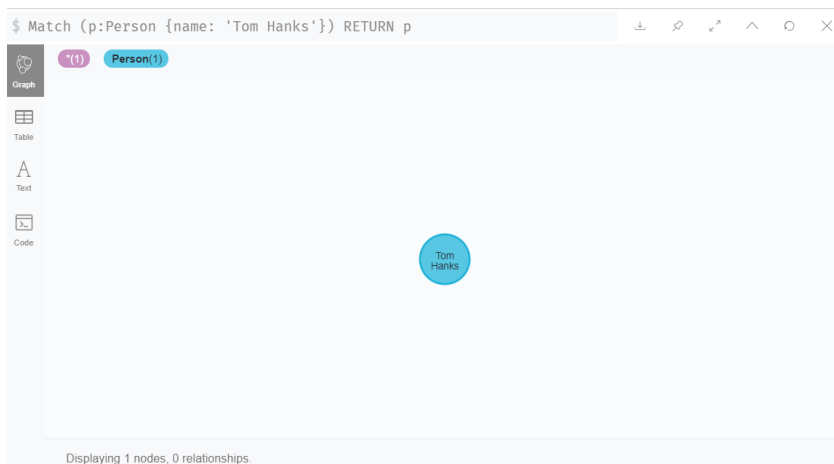
The command to create a new person node with property name having value 'John Doe':

Create (p:Person {name: 'John Doe'}) RETURN p

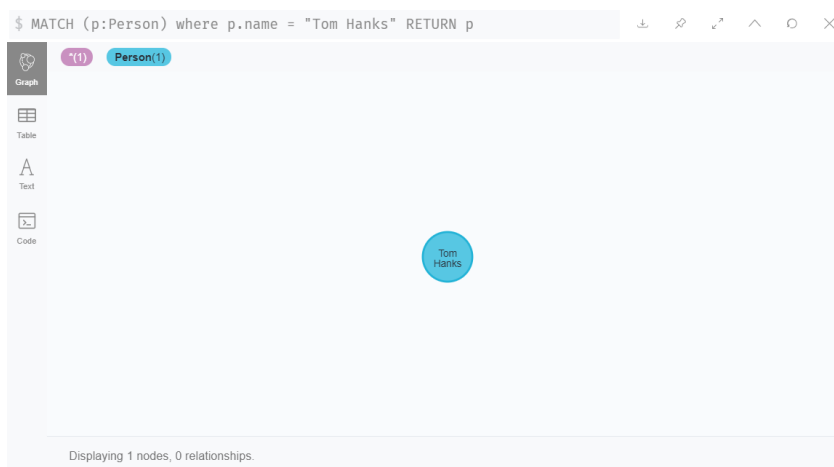
Then we check if that node exist:



Let us check another node with name 'Tom hanks':



The above same query can also be written as:



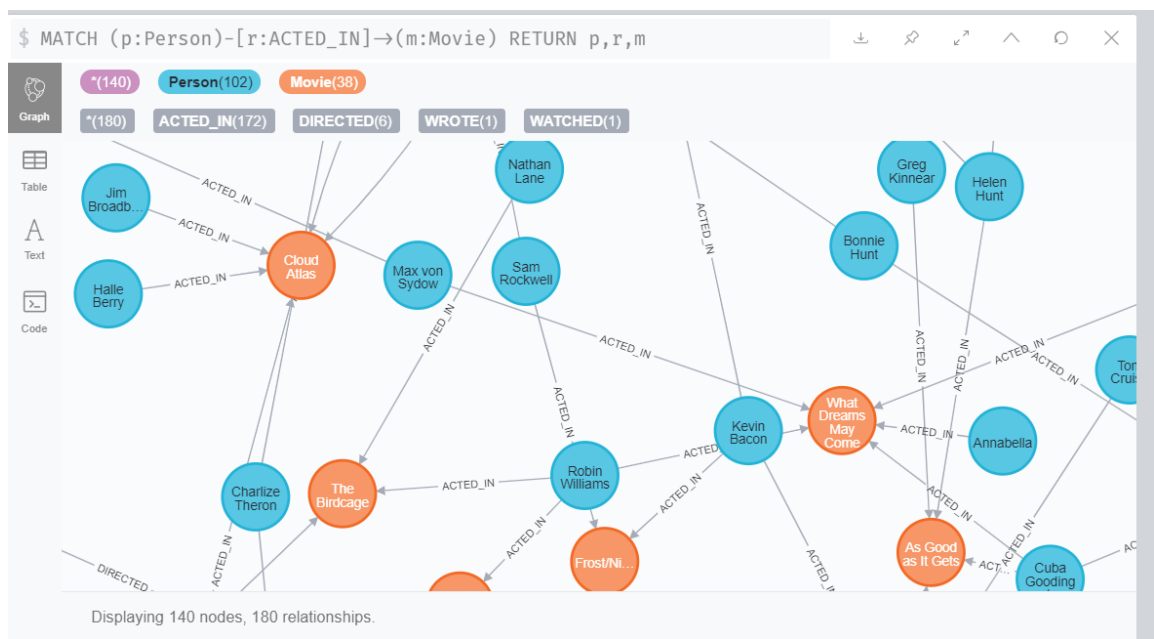
Statement to **create the Person node if it does not exist**. If the node already exists, then it will set the property lastLoggedInAt to the current timestamp. If node did not exist and was newly created instead, then it will set the createdAt property to the current timestamp:



Statement to **create a relationship :WATCHED between the existing Person and Movie nodes** and return the type of relationship (i.e WATCHED):



To show all the ACTED_IN relationships between nodes Person and Movie:



Finding who directed ‘Cloud Atlas’ movie:

\$ MATCH (m:Movie {title: 'Cloud Atlas'})←[d:DIRECTED]-(p:Person) return p.name

Table

Text

Code

p.name
"Tom Tykwer"
"Lana Wachowski"
"Lilly Wachowski"

Finding all people who have co-acted with ‘Tom Hanks’ in any movie:

\$ MATCH (tom:Person {name: "Tom Hanks"})-[:ACTED_IN]→(:Movie)←[:ACTED_IN]-(p:Person) return p.name

Table

Text

Code

p.name
"Ed Harris"
"Gary Sinise"
"Kevin Bacon"
"Bill Paxton"
"Parker Posey"
"Greg Kinnear"
"Meg Ryan"
"Steve Zahn"
"Dave Chappelle"

Started streaming 39 records after 1 ms and completed after 4 ms.

Finding all people related to the movie ‘Cloud Atlas’ in any way :

\$ MATCH (p:Person)-[relatedTo]-(m:Movie {title: "Cloud Atlas"}) return p.name, type(relatedTo)

Table

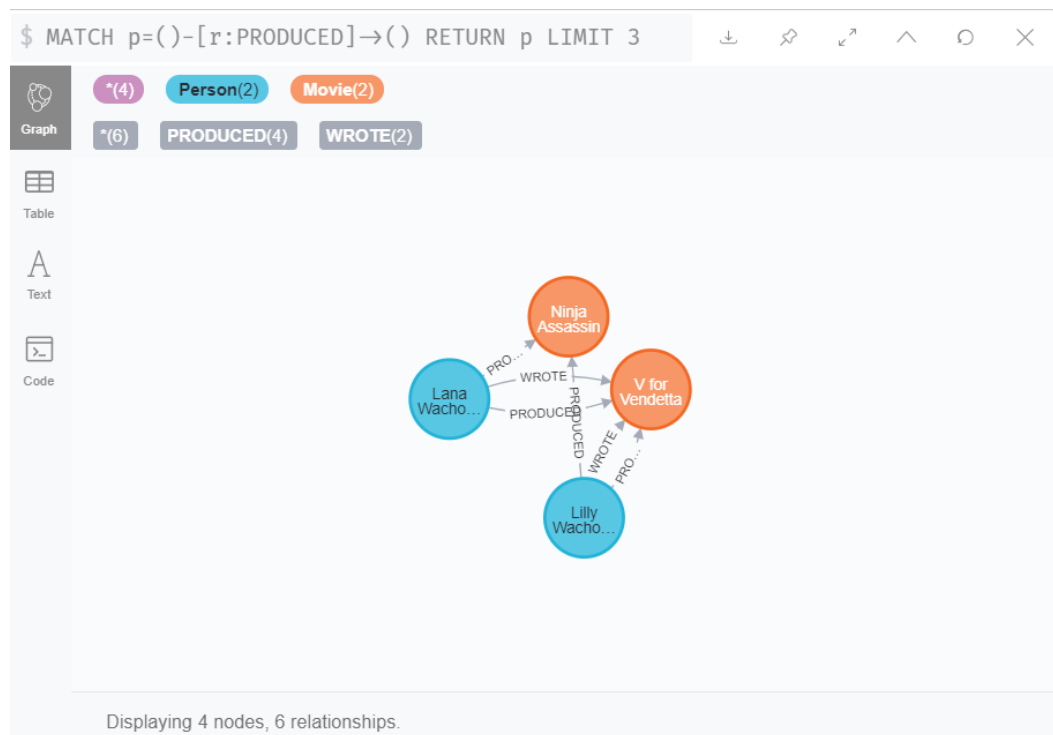
Text

Code

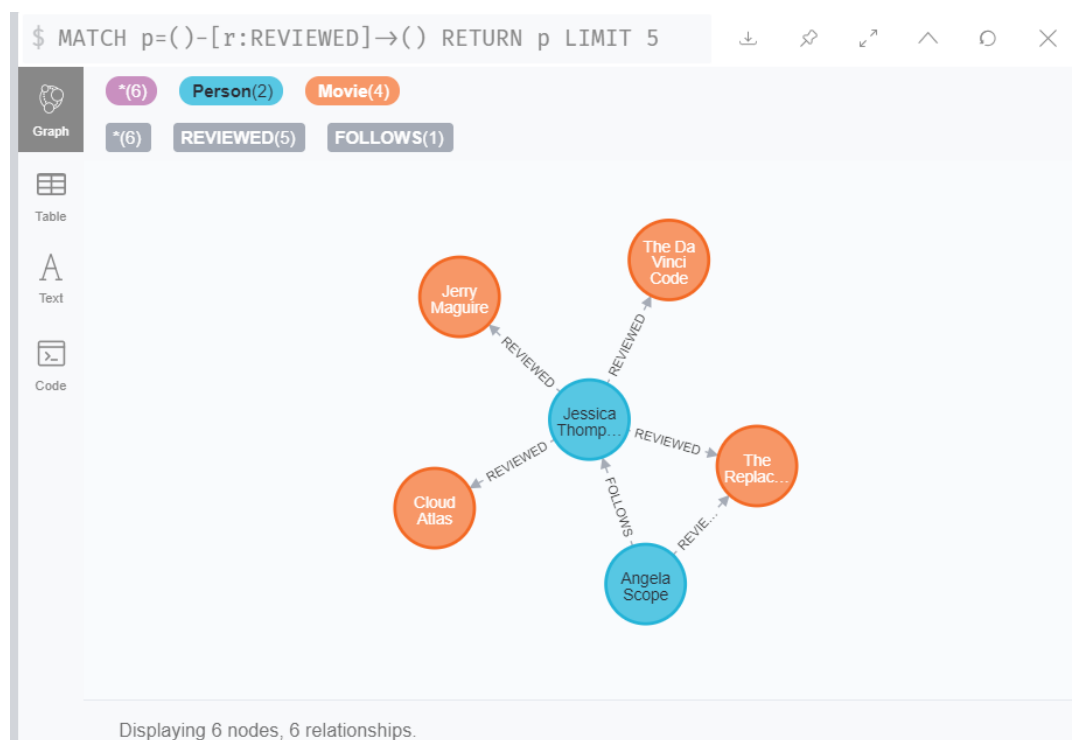
p.name	type(relatedTo)
"Tom Hanks"	"WATCHED"
"Tom Hanks"	"ACTED_IN"
"Jim Broadbent"	"ACTED_IN"
"David Mitchell"	"WROTE"
"Tom Tykwer"	"DIRECTED"
"Lana Wachowski"	"DIRECTED"
"Stefan Arndt"	"PRODUCED"
"Jessica Thompson"	"REVIEWED"
"Halle Berry"	"ACTED_IN"

Started streaming 11 records after 1 ms and completed after 2 ms.

Query to return Nodes with PRODUCED relationship, limiting it to 3:



Query to return nodes related with REVIEWED relationship, limiting it to 5:



Thank you