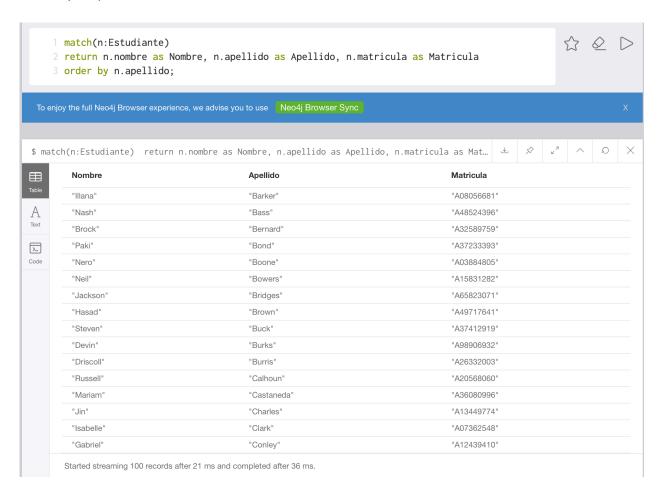
Alexandro Francisco Marcelo González A01021383 Bases de Datos Avanzadas Profesor: Dr. Vicente Cubells Nonell 29 de marzo de 2019

Tarea 4. Graph databases

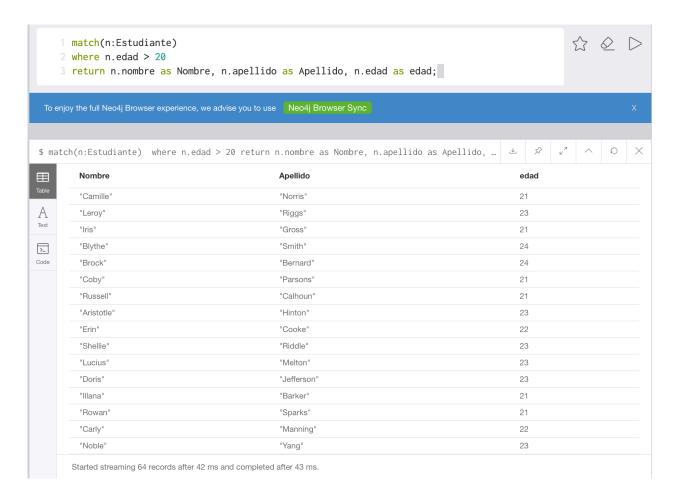
3.1. El nombre de todos los estudiantes ordenados ascendentemente por los apellidos.

match(n:Estudiante) return n.nombre as Nombre, n.apellido as Apellido, n.matricula as Matricula order by n.apellido;



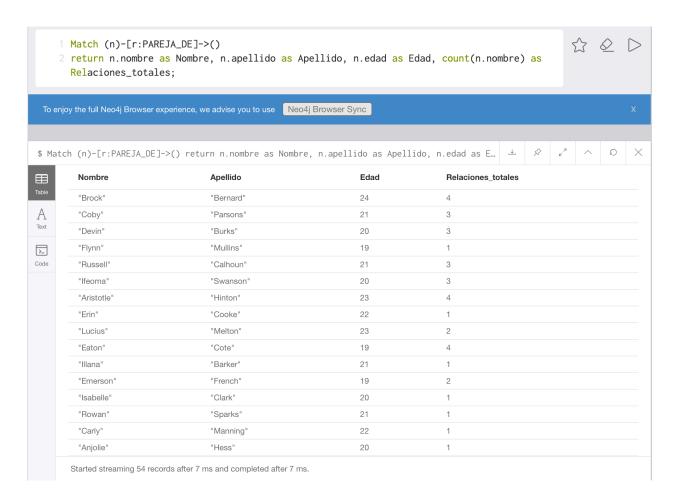
3.2. El nombre y edad de todos los estudiantes mayor de 20 años.

match(n:Estudiante) where n.edad > 20 return n.nombre as Nombre, n.apellido as Apellido, n.edad as edad;



3.3. El nombre y edad de todos los estudiantes que han tenido al menos una pareja.

Match (n)-[r:PAREJA_DE]->() return n.nombre as Nombre, n.apellido as Apellido, n.edad as Edad, count(n.nombre) as Relaciones_totales;



3.4. Dado el nombre de un estudiante, obtener quiénes han sido sus parejas y qué tiempo duró cada relación.

Match (n {nombre: 'Emerson'})-[r:PAREJA_DE]->(m)
unwind[
duration.between(date(r.inicioRelacion), date(r.finRelacion))
]as Duracion_Relacion
return m.nombre as Nombre, m.apellido as Apellido, m.edad as Edad, m.matricula as Matricula,
Duracion_Relacion;



3.5. Quién es el estudiante que ha tenido más relaciones de amistad.

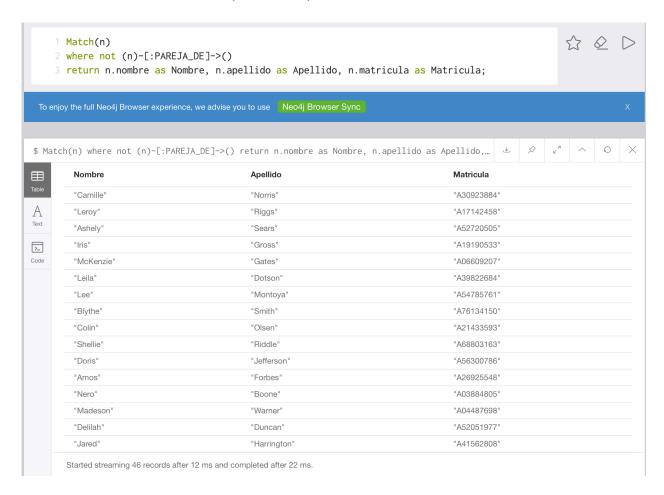
Match (n)-[r:AMIGO_DE]->()

return n.nombre as Nombre, n.apellido as Apellido, count(n.matricula) as total_amigos order by(total_amigos) desc limit 1;



3.6. Quiénes son los estudiantes que no han tenido relaciones de pareja.

match(n)
where not (n)-[:PAREJA_DE]->()
return n.nombre as Nombre, n.apellido as Apellido, n.matricula as Matricula;



3.7. Quiénes son los estudiantes que han tenido mas de 3 relaciones de pareja.

Match (n)-[r:PAREJA_DE]->()
with n, count(n.nombre) as Total_parejas
where Total_parejas > 3
return n.nombre as Nombre, n.apellido as Apellido, n.matricula as Matricula, Total_parejas;



3.8. Dado un estudiante, determinar el número de estudiantes que él conoce.

Match (n {nombre: 'Lucy'})-[r:CONOCE_A]->() return n.nombre as Nombre, n.apellido as Apellido, n.matricula as Matricula, count(n) as Conoce_personas;



3.9. Dado un estudiante, determinar quiénes lo conocen a él.

Match (n:Estudiante)-[r:CONOCE_A]->(m:Estudiante {nombre: 'Moana'}) return n.nombre as Nombre, n.apellido as Apellido, n.matricula as Matricula;

