

Projecte Neo4j

Exercici 1

Per tal d'importar correctament les dades a les base de dades de Neo4j s'ha creat el següent script de Cypher:

Primer de tot, es creant els respectius constraints per tal de prevenir la duplicació de d'identificadors dels habitatges i de les persones.

```
// Creació de constraints
```

```
CREATE CONSTRAINT UniqueLlarConstraint IF NOT EXISTS FOR (h:Habitatge)
```

```
REQUIRE h.Id_Llar IS UNIQUE;
```

```
CREATE CONSTRAINT UniqueIndividualIdConstraint IF NOT EXISTS FOR (i:Individual)
```

```
REQUIRE i.Id IS UNIQUE;
```

Seguidament es creen 2 índexs per tal de millorar en rendiment en les consultes. El primer per consultes que busquen individus pel nom i cognoms. I el segon per búsquedes d'habitatges pel municipi. S'han escollit aquests crear aquests 2 índexs basant-nos en les consultes de l'exercici 2.

```
// Creació d'índexs
```

```
CREATE INDEX IndividualNameSurname IF NOT EXISTS FOR (i:Individual) ON (i.name, i.surname, i.second_surname);
```

```
CREATE INDEX HabitatgeMunicipi IF NOT EXISTS FOR (h: Habitatge) ON (h.municipi);
```

Finalment es carreguen les dades. S'ha utilitzat el merge per així si s'executa més d'una vegada no hi hagi duplicitat de dades. Per obtenir les dades es fan les conversions corresponents per cada camp i les files les quals tenen els camps Id de municipi, llar o individu amb valor nul no s'afegeixen.

```
// Càrrega de dades de HABITATGES
```

```
LOAD CSV WITH HEADERS FROM 'file:///HABITATGES.csv' AS row
```

```
WITH row WHERE NOT row.Id_Llar IS NULL AND NOT row.Municipi IS NULL
```

```
MERGE (h:Habitatge {Id_Llar: row.Id_Llar})
```

```
SET h.Municipi = row.Municipi,
```

```
h.Any_Padro = toInteger(row.Any_Padro),
```

```
h.Carrer = row.Carrer,
```

```
h.Numero = toInteger(row.Numero);
```

```
// Càrrega de dades de INDIVIDUAL
```

```
LOAD CSV WITH HEADERS FROM 'file:///INDIVIDUAL.csv' AS row
```

```
WITH row WHERE NOT row.Id IS NULL
```

```
MERGE (i:Individual {Id: row.Id})
```

```
SET i.Year = toInteger(row.Year),
```

```
i.name = row.name,
```

```
i.surname = row.surname,
```

```
i.second_surname = row.second_surname;
```

```
// Càrrega de dades de VIU
```

```
LOAD CSV WITH HEADERS FROM 'file:///VIU.csv' AS row
```

```
WITH row WHERE NOT row.IND IS NULL AND NOT row.HOUSE_ID IS NULL
```

```
MATCH (i:Individual {Id: row.IND})
```

```
MATCH (h:Habitatge {Id_Llar: row.HOUSE_ID})
```

```
MERGE (i)-[:VIU {Location: row.Location, Year: toInteger(row.Year)}]->(h);
```

```
// Càrrega de dades de SAME_AS
```

```
LOAD CSV WITH HEADERS FROM 'file:///SAME_AS.csv' AS row
```

```
MATCH (a:Individual {Id: row.Id_A})
```

```
MATCH (b:Individual {Id: row.Id_B})
```

```
MERGE (a)-[:SAME_AS]->(b);
```

```
// Càrrega de dades de FAMILIA
```

```
LOAD CSV WITH HEADERS FROM 'file:///FAMILIA.csv' AS row
```

```
MATCH (a:Individual)
```

Pol Termes 1671849, Neil Pradas 1671897, Arnau Garriga 1676160, Sahel Caro 1674373

MATCH (b:Individual)

WHERE a.Id = row.ID_1 AND b.Id = row.ID_2

MERGE (a)-[:FAMILIA {Relacio: row.Relacio, Relacio_Harmonitzada: row.Relacio_Harmonitzada}]->(b);

Exercici 2

- a) Per a cada padró (any) de Sant Feliu de Llobregat (SFLL), retorna l'any de padró, el número d'habitants, i la llista de cognoms. Elimina duplicats i "nan".

MATCH (h:Habitatge {Municipi: 'SFLL'})<-[:VIU]-(i:Individual)

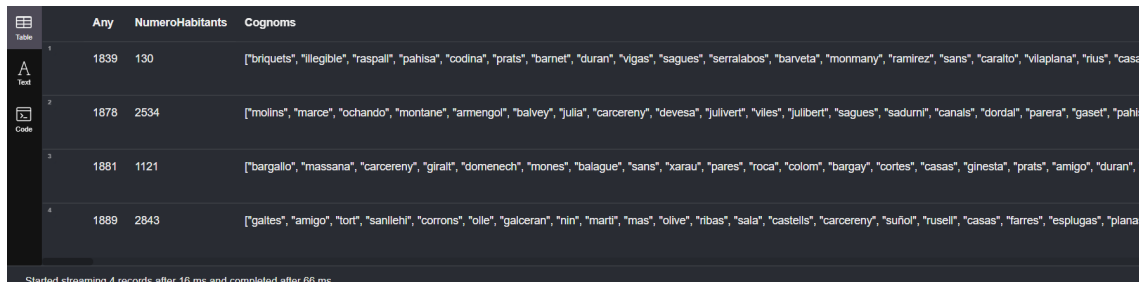
WHERE NOT i.surname IS NULL AND i.surname <> 'nan'

WITH h.Any_Padro AS Any, collect(distinct i.surname) AS Cognoms, count(distinct i) AS NumeroHabitants

RETURN Any, NumeroHabitants, Cognoms

ORDER BY Any;

Resultat:



	Any	NumeroHabitants	Cognoms
1	1839	130	["briquets", "illegible", "raspall", "pahisa", "codina", "prats", "barret", "duran", "vigas", "sagues", "serralabos", "barveta", "monmany", "ramirez", "sans", "caralto", "vilaplana", "rius", "casa
2	1878	2534	["molins", "marce", "ochando", "montane", "armengol", "balvey", "julia", "carcereny", "devesa", "julvert", "viles", "julibert", "sagues", "sadurni", "canals", "dordal", "parera", "gaset", "pahis
3	1881	1121	["bargallo", "massana", "carcereny", "giralt", "domenech", "mones", "balague", "sans", "xarau", "pares", "roca", "colom", "bargay", "cortes", "casas", "ginesta", "prats", "amigo", "duran", "
4	1889	2843	["galtes", "amigo", "tort", "sanlehi", "corrons", "olle", "galceran", "nin", "marti", "mas", "olive", "ribas", "sala", "castells", "carcereny", "suñol", "rusell", "casas", "farres", "esplugas", "planas

- b) Retorna totes les aparicions de "miguel estape bofill". Fes servir la relació SAME_AS per poder retornar totes les instàncies, independentment de si hi ha variacions lèxiques (ex. diferents formes d'escriure el seu nom/cognoms). Mostra la informació en forma de taula: el nom, la llista de cognoms i la llista de segon cognom (elimina duplicats).

MATCH (i:Individual {name: 'miguel', surname: 'estape', second_surname: 'bofill'})-[:SAME_AS]->(miguel:Individual)*

RETURN miguel.name AS Nom,

collect(distinct miguel.surname) AS PrimerCognom,

Pol Termes 1671849, Neil Pradas 1671897, Arnau Garriga 1676160, Sahel Caro 1674373

collect(distinct miguel.second_surname) AS SegonCognom;

Resultat:

	Nom	PrimerCognom	SegonCognom
1	"miguel"	["estape"]	["bofill"]

c) Mostra els fills o filles (només) de "benito julivert". Mostra la informació en forma de taula: el nom, cognom1, cognom2, i tipus de relació. Ordena els resultats alfabèticament per nom.

MATCH (pare:Individual {name: 'benito', surname: 'julivert'})

MATCH (pare)-[r:FAMILIA]->(fill:Individual)

WHERE r.Relacio_Harmonitzada IN ['fill', 'filla']

RETURN fill.name AS Nom,

fill.surname AS PrimerCognom,

fill.second_surname AS SegonCognom,

r.Relacio_Harmonitzada AS Tipus_Relacio

ORDER BY Nom;

Resultat:

	Nom	PrimerCognom	SegonCognom	Tipus_Relacio
1	"dolores"	"julibert"	"julia"	"filla"
2	"joaquina"	"julibert"	"julia"	"filla"
3	"jose"	"julibert"	"julia"	"fill"
4	"juan"	"julibert"	"julia"	"fill"
5	"magdalena"	"julibert"	"julia"	"filla"
6	"martin"	"julibert"	"julia"	"fill"

Started streaming 6 records after 9 ms and completed after 19 ms.

- d) Mostreu les famílies de Castellví de Rosanes amb més de 3 fills. Mostreu el nom i cognoms del cap de família i el nombre de fills. Ordeneu-les pel nombre de fills fins a un límit de 20, de més a menys.

```
MATCH (h:Habitatge)-[:VIU {Location: 'CR'}]-(p:Individual)

MATCH (p)-[:FAMILIA {Relacio_Harmonitzada: 'fill'}]->(c:Individual)

WITH p, COUNT(c) AS Numero_Fills

WHERE Numero_Fills > 3

RETURN p.name AS Nom_Cap_Familia,

       p.surname AS PrimerCognom,

       p.second_surname AS SegonCognom,

       Numero_Fills

ORDER BY Numero_Fills DESC

LIMIT 20;
```

Nom_Cap_Familia	PrimerCognom	SegonCognom	Numero_Fills
"jaime"	"jarrey"	"ilegible"	4
"jose"	"canals"	"mila"	4

- e) Per cada padró/any de Sant Feliu de Llobregat, mostra el carrer amb menys habitants i el nombre d'habitants en aquell carrer. Fes servir la funció min() i CALL per obtenir el nombre mínim d'habitants. Ordena els resultats per any de forma ascendent.

```
MATCH (i:Individual)-[:VIU]->(h:Habitatge {Municipi: "SFL"})

WITH h.Carrer AS Carrer, i.Year AS Any_Padro, COUNT(i) AS NumHabitants

ORDER BY Any_Padro

WITH Any_Padro, COLLECT({Carrer: Carrer, NumHabitants: NumHabitants}) AS Carrers,

       min(NumHabitants) AS MinNumHabitants
```

```
UNWIND Carrers AS CarrerData

WITH Any_Padro, CarrerData, MinNumHabitants

WHERE CarrerData.NumHabitants = MinNumHabitants

RETURN Any_Padro, CarrerData.Carrer AS Carrer, CarrerData.NumHabitants AS
NumHabitants

ORDER BY Any_Padro ASC
```

Any_Padro	Carrer	NumHabitants
1833	"carrretera"	1
1833	"calle de la carretera"	1
1838	"masover nou"	55
1839	"falguer"	2
1866	"caretera"	1
1878	"caretera"	2
1878	"carrretera"	2
1881	"falguer"	2
1881	"grane"	2
1889	"s n antonio"	1

Exercici 3

- a) Estudi de les components connexes (cc) i de l'estructura de les components en funció de la seva mida. Feu servir el mode stream. Un cop calculades les components connexes (nodes individu, habitatge i relació VIU), feu dues consultes per explorar les dades. Per exemple (podeu fer-ne d'altres):
- Mostra, en forma de taula, les 10 components connexes més grans (ids i mida).
 - Per cada municipi i any el nombre de parelles del tipus: (Individu)— (Habitatge).
 - Quantes components connexes no estan connectades a cap node de tipus 'Habitatge', és a dir, els individus sense casa.

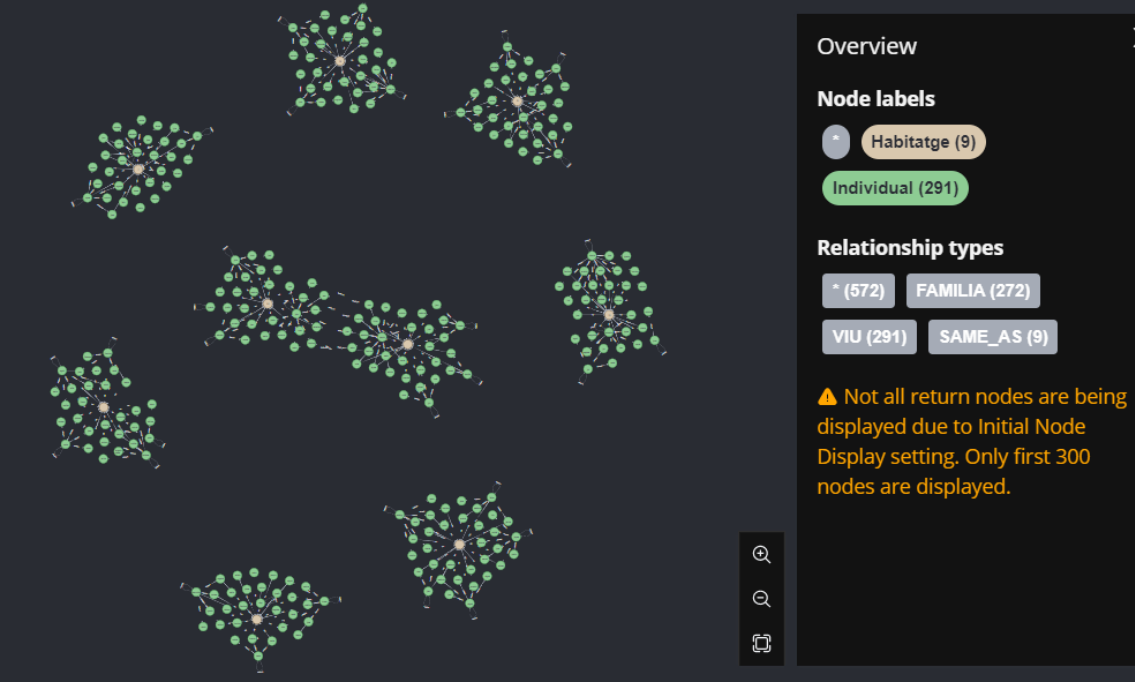
Primer de tot, s'ha de carregar un graf a memòria amb els nodes *Individual* i *Habitatge* i la relació *VIU*:

```
CALL gds.graph.project(  
  'IndividualHabitatgeGraph',  
  ['Individual', 'Habitatge'],  
  {  
    VIU: {  
      type: 'VIU'  
    }  
  }  
);
```

Per tal de visualitzar les 10 components connexes febles més grans s'ha dut a terme aquesta consulta:

```
CALL gds.wcc.stream('IndividualHabitatgeGraph') YIELD nodeId, componentId AS community  
WITH gds.util.asNode(nodeId) AS node, community  
WITH collect(node) AS allNodes, community  
RETURN community, allNodes AS nodes, size(allNodes) AS size  
ORDER BY size DESC  
LIMIT 10;
```

Resultat:



Si es vol veure en format files:

```
CALL gds.wcc.stream('IndividualHabitatgeGraph') YIELD nodeId, componentId AS community
WITH gds.util.asNode(nodeId).name AS nodeName, community
WITH collect(nodeName) AS allNodeNames, community
RETURN community, allNodeNames, size(allNodeNames) AS size
ORDER BY size DESC
LIMIT 10;
```

Resultat:

	community	allNodeNames
1	31	["mariangela", "lorenzo", "paula", "pedro", "vicente", "maria", "catalina", "francisca", "maria", "tomas", "juan", "madrone", "franca", "rosa", "miguel", "amalia"]
2	244	["jph", "rosa", "miguel", "juaquima", "antonia", "rosa", "baudilio", "jose", "teresa", "mercedes", "joaquin", "maria", "bruno", "rosa", "pedro", "francisco", "ana"]
3	609	["jph", "margarida", "maria", "jph", "agusti", "jose", "antonia", "sebastian", "magdalena", "maria", "juan", "jaye", "vicenta", "ramon", "lorenzo", "rosa", "anto"]
4	9	["pablo", "francisca", "pedro", "maria", "ramon", "teresa", "francisco", "rosa", "carmen", "eulalia", "jacinto", "rosa", "jph", "llorens", "miguel", "jph", "josep", "m"]
5	422	["catalina", "ramon", "feliu", "ignasi", "franca", "franco", "franca", "jph", "baldiri", "franco", "franca", "jose", "joaquina", "antonia", "dolores", "margarita", "anto"]
6	614	["jph", "jpha", "pere", "antonia", "francisco", "teresa", "juan", "salvador", "eulalia", "mercedes", "lorenzo", "josefa", "salvador", "maria", "jaye", "paula", "jua"]

ted streaming 10 records after 23 ms and completed after 139 ms.

Per saber quantes components connexes no estan connectades a cap node de tipus 'habitatge' hem fet la següent consulta:

```
CALL gds.wcc.stream('IndividualHabitatgeGraph')
YIELD nodeId, componentId AS community
WITH gds.util.asNode(nodeId) AS node, community
OPTIONAL MATCH (node)-[:VIU]->(h:Habitatge)
WITH community, collect(h) AS habitatges
WHERE all(h in habitatges WHERE h IS NULL)
RETURN count(DISTINCT community) AS NumComponentsSenseHabitatge;
```

OUTPUT:

NumComponentsSenseHabitatge
4741

b) Semblança entre els nodes. Ens interessa saber quins nodes són semblants com a pas previ a identificar els individus que són el mateix (i unirem amb una aresta de tipus SAME_AS). Abans de fer aquest anàlisi:

- Determineu els habitatges que són els mateixos al llarg dels anys. Afegiu una aresta amb nom "MATEIX_HAB" entre aquests habitatges. Per evitar arestes duplicades feu que la aresta apunti al habitatge amb any de padró més petit.
 - Creeu un graf en memòria que inclogui els nodes Individu i Habitatge i les relacions VIU, FAMILIA, MATEIX_HAB que acabeu de crear.
 - Calculeu la similaritat entre els nodes del graf que acabeu de crear, escriviu el resultat de nou a la base de dades i interpreteu els resultats obtinguts.
1. Primer de tot busquem tots els parells possibles de nodes Habitatge a la base de dades. Seguidament creem una relació MATEIX_HAB entre els dos nodes Habitatge seleccionats (h1 i h2) que indica que aquests dos nodes representen el mateix habitatge però en anys diferents. La relació es crea de manera que apunti del node amb l'any de padró més petit (h1) cap al node amb l'any de padró més gran (h2), evitant així duplicats.

```
MATCH (h1:Habitatge), (h2:Habitatge)
```

```
WHERE h1.Carrer = h2.Carrer AND h1.Numero = h2.Numero AND h1.Municipi =  
h2.Municipi AND h1.Any_Padro < h2.Any_Padro
```

```
MERGE (h1)-[:MATEIX_HAB]->(h2)
```

2. Després d'afegir les arestes, creem el graf que inclogui els nodes Individu i Habitatge i les relacions VIU, FAMILIA, MATEIX_HAB

```
CALL gds.graph.project(
  'graf_individu_habitatge',
  ['Individual', 'Habitatge'],
  {
    VIU: { type: 'VIU' },
    FAMILIA: { type: 'FAMILIA' },
    MATEIX_HAB: { type: 'MATEIX_HAB' }
  }
)
```

- 3.

Primer, busquem els habitatges que estan connectats per la relació MATEIX_HAB per trobar els comuns (intersecció) entre cada parella de nodes Habitatge. Després, calculem la unió de veïns per a cada node Habitatge. Finalment, calculem la similitud de Jaccard entre aquests nodes i la desmem a la base de dades com una nova relació SIMILAR_TO amb la propietat de similitud.

```
MATCH(h1:Habitatge)-[:MATEIX_HAB]-(common:Habitatge)-[:MATEIX_HAB]-(h2:Habitatge)
WHERE id(h1) < id(h2)
WITH h1, h2, COUNT(common) AS intersection
MATCH (h1)-[:MATEIX_HAB]-(n:Habitatge)
WITH h1, h2, intersection, COUNT(DISTINCT n) AS union1
MATCH (h2)-[:MATEIX_HAB]-(m:Habitatge)
WITH h1, h2, intersection, union1, COUNT(DISTINCT m) AS union2
WITH h1, h2, intersection, (union1 + union2 - intersection) AS union_count
WITH h1, h2, (1.0 * intersection / union_count) AS jaccard_similarity
MERGE (h1)-[r:JACCARD_SIMILARITY]->(h2)
SET r.similarity = jaccard_similarity
RETURN h1, h2, r.similarity
```

h1	h2	r.similarity
{:Habitatge {Numero: 15,Any_Padro: 1878,Carrer: "serra",Municipi: "SFL",Id_Llar: "447"}}	{:Habitatge {Numero: 15,Any_Padro: 1878,Carrer: "serra",Municipi: "SFL",Id_Llar: "446"}}	1.0
{:Habitatge {Numero: 4,Any_Padro: 1878,Carrer: "serra",Municipi: "SFL",Id_Llar: "433"}}	{:Habitatge {Numero: 4,Any_Padro: 1878,Carrer: "serra",Municipi: "SFL",Id_Llar: "434"}}	1.0
{:Habitatge {Numero: 10,Any_Padro: 1878,Carrer: "serra",Municipi: "SFL",Id_Llar: "440"}}	{:Habitatge {Numero: 10,Any_Padro: 1878,Carrer: "serra",Municipi: "SFL",Id_Llar: "441"}}	1.0
{:Habitatge {Numero: 20,Any_Padro: 1881,Carrer: "abajo",Municipi: "SFL",Id_Llar: "496"}}	{:Habitatge {Numero: 20,Any_Padro: 1889,Carrer: "abajo",Municipi: "SFL",Id_Llar: "473"}}	0.25
{:Habitatge {Numero: 20,Any_Padro: 1878,Carrer: "abajo",Municipi: "SFL",Id_Llar: "584"}}	{:Habitatge {Numero: 20,Any_Padro: 1889,Carrer: "abajo",Municipi: "SFL",Id_Llar: "473"}}	0.25
{:Habitatge {Numero: 20,Any_Padro: 1878,Carrer: "abajo",Municipi: "SFL",Id_Llar: "584"}}	{:Habitatge {Numero: 20,Any_Padro: 1881,Carrer: "abajo",Municipi: "SFL",Id_Llar: "496"}}	0.5