Grau en Matemàtica Computacional i Analítica de Dades

Pràctica Neo4j

Bases de Dades no Relacionals

Judit Yebra Valencia (1603614)
Pau Fuentes Hernández (1600158)
Andrea González Aguilera (1603921)
Xavier Seminario Monllaó (1603853)

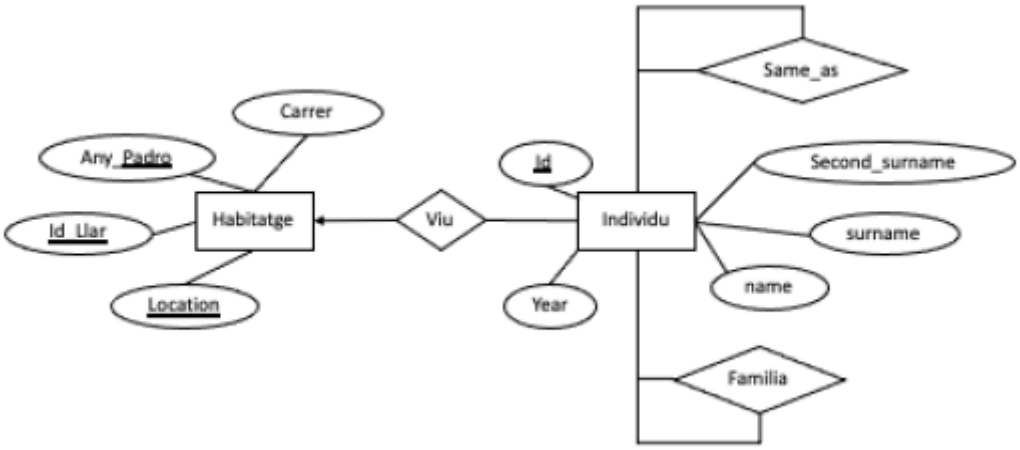
Índex

1 Distribució de les tasques

- csv ok, modificació de noms, eliminació files amb nulls (no tenien sentit), ... : tots -

2 Introducció

L'objectiu principal d'aquest projecte és explorar com utilitzar Neo4J. Per dur-ho a terme es farà servir una base de dades sobre uns padrons els quals són els llistats d'habitants que elabora un municipi on figura la informació la seva informació com noms, cognoms, edat i altres dades personals. Aquest padró està organitzat segons el següent disseny Entitat-Relació:



Aquesta base de dades està conformada per cinc arxius els quals corresponen a les entitats i les relacions del diagrama anterior. Aquests arxius contenen la diferent informació que hi havia en el padró.

3 Link al repositori de github

Tot seguit es disposa el link, per poder accedir al repositori de github on es troba tota la informació la qual s'ha usat per la realització d'aquest treball.
link repositori grup08.

4 Exercici 1

Importa les dades en la BD de Neo4j del projecte. Genera un script en cypher que carregui totes les dades, generi tots els nodes, relacions i afegeixi les característiques allà on toqui. Consideracions:

- **Feu servir *constrains* i *indexos* quan sigui necessari.**
- **Assegureu-vos que en executar el script dues vegades no es dupliquin les dades.**
- **No carregueu files *null* del fitxer CSV (Id de municipi, llar o individu = *null*).**
- **Feu les conversions de tipus que siguin necessàries.**

Per aquest exercici es proporcionarà un document amb els passos documentats per importar dades amb èxit a un projecte per utilitzar-lo amb Neo4j.

5 Exercici 2

Resoleu les següents consultes:

1. **Del padró de 1866 de Castellví de Rosanes (CR), retorna el número d'habitants i la llista de cognoms, sense eliminar duplicats.**
   Codi:

```
MATCH (p:Individu)-[:VIU]->(h:Habitatge)
WHERE h.Any_Padro=1866 AND h.Municipi="CR"

RETURN count(p), collect(p.Cognom)
```

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

```
MATCH (i:Individu)-[:VIU]->(h:Habitatge)
WHERE h.Municipi = "SFLL" AND i.Cognom <> 'nan'
WITH h.Id AS ID, i.Any_Padro AS Any_Padro,
    count(h.Id) AS Recompte,
    collect(DISTINCT(i.Cognom)) AS Cognoms

RETURN Any_Padro, Recompte, Cognoms ORDER BY Any_Padro ASC
```

3. **Dels padrons de Sant Feliu de Llobregat (SFLL) d'entre 1800 i 1845 (no inclosos), retorna la població, l'any del padró i la llista d'identificadors dels habitatges de cada padró. Ordena els resultats per l'any de padró.**
   Codi:

```
MATCH (h:Habitatge{Municipi:"SFLL"})
WHERE 1800<h.Any_Padro<1845

RETURN h.Municipi, h.Any_Padro, collect(distinct h.Id)
ORDER BY h.Any_Padro
```

4. **Retorna el nom de les persones que vivien al mateix habitatge que "rafel marti"(no té segon cognom) segons el padró de 1838 de Sant Feliu de Llobregat (SFLL). Retorna la informació en mode graf i mode llista.**
   Codi:

```
MATCH (p:Individu{Nom:'rafel',Cognom:'marti'})-[:VIU]->
    (h:Habitatge{Municipi:"SFLL",Any_Padro:1838})<-[:VIU]-(p2:Individu)

RETURN p2.Nom
```

5. **Retorna totes les aparicions de "miguel estape bofill". Fes servir la relació SAME_AS per poder retornar totes les instancies, independentment de si hi ha variacions lèxiques (ex. diferents formes d'escriure el seu nom/cognoms). Mostra la informació en forma de subgraf.**
   Codi:

```
MATCH (p:Individu{Nom:'miguel',Cognom:'estape',Segon_Cognom:'bofill'})
    <-[:SAME_AS]->(p2:Individu)

RETURN p,p2
```

6. **De la consulta anterior, retorna la informació en forma de taula: el nom, la llista de cognoms i la llista de segon cognom (elimina duplicats).**
   Codi:

```
MATCH (p:Individu{Nom:'miguel',Cognom:'estape',Segon_Cognom:'bofill'})
    <-[:SAME_AS]->(p2:Individu)

RETURN p.Nom, collect(distinct p2.Cognom),
    collect(distinct p2.Segon_Cognom)
```

7. **Mostra totes les persones relacionades amb "benito julivert". Mostra la informació en forma de taula: el nom, cognom1, cognom2, i tipus de relació.**
   Codi:

```
MATCH (p:Individu
{Nom:'benito',Cognom:'julivert'})-[f:FAMILIA]-(otraPersona:Individu)

RETURN otraPersona.Nom AS Nom,
    otraPersona.Cognom AS Cognom1,
    otraPersona.Segon_Cognom AS Cognom2,
    f.Relacio\_Harmonitzada AS Tipus\_Relaci
```

8. **De la consulta anterior, mostra ara només els fills o filles de "benito julivert". Ordena els resultats alfabèticament per nom.**
   Codi:

```
MATCH (p:Individu {Nom:'benito',
Cognom:'julivert'})-[f:FAMILIA]-(fill:Individu)
WHERE f.Relacio_Harmonitzada IN ['fill', 'filla']

RETURN fill.Nom AS Nom,
    fill.Cognom AS Cognom1,
    fill.Segon_Cognom AS Cognom2,
    f.Relacio_Harmonitzada AS Tipus_Relaci
ORDER BY Nom ASC
```

9. **Llisteu totes les relacions familiars que hi ha.**
   Codi:

```
MATCH (:Individu)-[f:Familia]->(:Individu)

RETURN f.Relacio
```

10. **Identifiqueu els nodes que representen el mateix habitatge (carrer i numero) al llarg dels padrons de Sant Feliu del Llobregat (SFLL). Seleccioneu només els habitatges que tinguin totes dues informacions (carrer i numero). Per a cada habitatge, retorneu el carrer i número, el nombre total de padrons on**

apareix, el llistat d'anys dels padrons i el llistat de les Ids de les llars (eviteu duplicats). Ordeneu de més a menys segons el total de padrons i mostreu-ne els 15 primers.
Codi:

```
MATCH (h:Habitatge)
WHERE h.Municipi = "SFLL" AND h.Carrer <> 'null' AND h.Numero <> -1

RETURN h.Carrer AS Carrer, h.Numero AS Numero, count(h.Id) AS
Aparicions,      collect(DISTINCT(h.Any_Padro)) AS Anys,
    collect(DISTINCT(h.Id)) AS Llista_Ids
ORDER BY Aparicions DESC
LIMIT 15
```

11. **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.**
Codi:

```
MATCH (f:FAMILIA)-[:VIU]->(:HABITATGE {Municipi: 'CR'})
WITH f, cap,
    size([ f in f.Relacio_Harmonitzada IN ['fill', 'filla']]) AS nfills
WHERE nfills > 3

RETURN f.cap.Nom AS Nom,
   f.cap.Cognom AS Cognom1,
   f.cap.Segon_Cognom AS Cognom2,
   nfills AS Nombre_de_fills
ORDER BY nfills DESC
LIMIT 20
```

12. **Mitja de fills a Sant Feliu del Llobregat l'any 1881 per família. Mostreu el total de fills, el nombre d'habitatges i la mitja de fills per habitatge. Fes servir CALL per obtenir el nombre de llars.**
Codi:

```
CALL{
    MATCH (p2:Individu)-[:VIU{Any_Habitatge:1881}]->
        (h:Habitatge{Municipi:'SFLL'})<-[:VIU]-(p:Individu)

    RETURN p,p2,h
}
MATCH(p)<-[r:FAMILIA]-(p2)
WHERE r.Relacio_Harmonitzada="fill" or r.Relacio_Harmonitzada="filla"

RETURN count(distinct p.Id),count(distinct h.Id),
    toFloat(count(distinct p.Id))/toFloat(count(distinct h.Id))
```

13. **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.**
Codi:

```
CALL {
    MATCH (i:Individu)-[:VIU]->(h:Habitatge)
    WHERE h.Municipi = 'SFLL'
    WITH h.Any_Padro AS Any, h.Carrer AS Carrer, count(i) AS Habitants

    RETURN Any, Carrer, min(Habitants) AS Min_Habitants
}

RETURN Any, Carrer, Min_Habitants
ORDER BY Any ASC
```

6 Exercici 3

En aquest exercici analitzarem les dades del graf per entendre millor l'estructura de les dades. Els següents apartats tenen com objectiu orientar-vos respecte a com utilitzar algunes de les eines que ofereix Neo4J.

a) Estudi de les components connexes (cc) i de l'estructura de les component en funció de la seva mida. A continuació uns indiquem algunes consultes que podeu fer per explorar les dades:

- Taula agrupant els resultats segons la mida de la cc.
- Distribució de tipus de nodes (Individu o Habitatge) segons la mida de la cc.
- 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'.

Aquestes consultes són només una orientació del tipus d'exploració de dades que podeu fer. Tant si feu aquestes com d'altres, heu d'acompanyar-les d'una motivació (que preteneu saber amb la consulta) i d'una explicació dels resultats. Acompanyeu aquesta explicació amb el codi de la consulta i el resultat obtingut. (Indicació: utilitzeu la funció wcc en mode 'stream')

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.
- Per cada municipi i any el nombre de parelles del tipus: (Individu)—(Habitatge).
- 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.