

Utilice la aplicación Neo4J para probar los siguientes ejemplos mostrados en el temario. Realice una captura de pantalla que muestre los resultados obtenidos. En la captura de pantalla deberá aparecer algún identificador que muestre la auditoría de cada alumno. Para la realización de esta actividad genere una base de datos en Neo4J con versiones 4.x.x

Para poder llevar a cabo estos ejemplos será necesario que la base de datos tenga una versión 4.x.x y que se instalen los siguientes plugins: APOC y GDS (Graph Data Science Library)

Ejemplo 1. Creación de grafos → importación de datos a partir de archivos en la nube. Ciudades

Listado 4.3: Importación de nodos

```
1 WITH "https://github.com/neo4j-graph-analytics/book/raw/master/data/transport-nodes.csv"
  AS uri
2 LOAD CSV WITH HEADERS FROM uri AS row
3 MERGE (place:Place {id:row.id})
4 SET place.latitude = toFloat(row.latitude),
5 place.longitude = toFloat(row.longitude),
6 place.population = toInteger(row.population)
```

Listado 4.4: Importación de aristas

```
1 WITH "https://github.com/neo4j-graph-analytics/book/raw/master/data/transport-relationships.csv" AS uri
2 LOAD CSV WITH HEADERS FROM uri AS row
3 MATCH (origin:Place {id: row.src})
4 MATCH (destination:Place {id: row.dst})
5 MERGE (origin)-[:EROAD {distance: toInteger(row.cost)}]->(destination)
```

Ejemplo 2. Creación de grafos → creación de grafo a partir de 0. Coocurrencia de hashtag

Listado 4.5: Creación grafo de co-ocurrencia de hashtags

```
1 CREATE
2     (JS:Hashtag {name: 'JoaquinSabina'}),
3     (RS:Hashtag {name: 'Rusia2018'}),
4     (AG:Hashtag {name: 'Argentina'}),
5     (FD:Hashtag {name: 'Feliz Domingo'}),
6     (MS:Hashtag {name: 'Messi'}),
7
8     (JS)-[:C00C {ntweet: 52}]->(FD),
9     (FD)-[:C00C {ntweet: 52}]->(JS),
10    (RS)-[:C00C {ntweet: 183}]->(FD),
11    (FD)-[:C00C {ntweet: 183}]->(RS),
12    (RS)-[:C00C {ntweet: 73}]->(AG),
13    (AG)-[:C00C {ntweet: 73}]->(RS),
14    (AG)-[:C00C {ntweet: 112}]->(MS),
15    (MS)-[:C00C {ntweet: 112}]->(AG),
16    (FD)-[:C00C {ntweet: 81}]->(MS),
17    (MS)-[:C00C {ntweet: 81}]->(FD)
```

Ejemplo 3. Importación de datos. Grafos

Importación de nodos

```
WITH "https://github.com/neo4j-graph-analytics/book/raw/master/data/" AS base
WITH base + "social-nodes.csv" AS uri
LOAD CSV WITH HEADERS FROM uri AS row
MERGE (:User {id: row.id})
```

Importación de grafo (aristas)

```
WITH "https://github.com/neo4j-graph-analytics/book/raw/master/data/" AS base
WITH base + "social-relationships.csv" AS uri
LOAD CSV WITH HEADERS FROM uri AS row
MATCH (source:User {id: row.src})
MATCH (destination:User {id: row.dst})
MERGE (source)-[:FOLLOWS]->(destination)
```

Realizar captura de pantalla

Añadir nueva conexión

```
MATCH (src {id: "David"})
MATCH (dst {id: "Mark"})
CREATE (src)-[:FOLLOWS]->(dst)
```

Realizar captura de pantalla

```
MATCH (src {id: "David"})-[:FOLLOWS]->(dst {id: "Mark"})
DELETE rel
```

Añadir nueva conexión

```
MATCH (src {id: "David"})
MATCH (dst {id: "Mark"})
CREATE (src)-[:FOLLOWS]->(dst)
```

Añadir nuevo nodo

```
CREATE (:User {id: "Kevin"})
```

Realizar captura de pantalla

Ejemplo 4. Recorridos sobre grafos. BFS recorrido en anchura

```
CALL gds.graph.create('myGraph', 'City', 'EROAD', {relationshipProperties: 'cost'})
//creación del grafo llamado "myGraph"
YIELD graphName, nodeCount, relationshipCount, createMillis; // asignamos los valores a las variables

MATCH (a:City {id: 'Doncaster'}) WITH id(a) AS startNode
CALL gds.alpha.bfs.stream('myGraph', { startNode: startNode }) // realizamos el recorrido BFS -> nº de caminos
YIELD path UNWIND [n in nodes(path) | n.id] AS tags // asignamos resultados
RETURN tags; // mostramos datos
```

Resultado:

```
"Doncaster"
"London"
"Colchester"
"Ipswich"
"Felixstowe"
"Hoek van Holland"
"Den Haag"
"Rotterdam"
"Gouda"
"Utrecht"
```

Ejemplo 5. Recorridos sobre grafos. Recorrido en profundidad (DFS)

```
CALL gds.graph.create('myGraph', 'City', 'EROAD', {relationshipProperties: 'cost'})
YIELD graphName, nodeCount, relationshipCount, createMillis;
```

	graphName	nodeCount	relationshipCount	createMillis	
1	"myGraph6"	12	15	6	

```
MATCH (a:City {id: 'Doncaster'})
WITH id(a) AS startNode
CALL gds.alpha.dfs.stream('myGraph', {startNode: startNode})
YIELD path
UNWIND [n in nodes(path) | n.id] AS tags
RETURN tags;
```

Resultado:

```
"Doncaster"
"London"
"Colchester"
"Ipswich"
"Felixstowe"
"Hoek van Holland"
```

"Rotterdam"

"Den Haag"

"Gouda"

"Utrecht"

Ejemplo 6.