

**Memoria Trabajo Práctico 2 (TP2)**

Bases de Datos Orientadas a Grafos: Neo4j

***Alumno****: Francisco Javier Piqueras Martínez*

***Asignatura****: Gestión y Almacenamiento de Información no Estructurada*

***Fecha******de******entrega****: 13 de mayo de 2020*

Índice

[1. Descripción del documento 4](#_Toc40207326)

[2. Ejercicios 4](#_Toc40207327)

[2.1. Ejercicio 1: Creación de la base de datos. 4](#_Toc40207328)

[2.2. Ejercicio 2: Consultas sobre la base de datos. 6](#_Toc40207329)

[2.3. Ejercicio 3: Creación de una base de datos propia. 7](#_Toc40207330)

Ilustraciones

[Ilustración 2‑1. Acceso al contenedor Neo4j 5](#_Toc40207334)

[Ilustración 2‑2. Confirmación carga Base de Datos Neo4j 5](#_Toc40207335)

[Ilustración 2‑3. Información de la base de datos tras la creación. 6](#_Toc40207336)

[Ilustración 2‑4. Esquema de la base de datos. 6](#_Toc40207337)

# Descripción del documento

Este documento consiste en la memoria del Trabajo Práctico 3, correspondiente al Módulo 5 de la asignatura de “Gestión y Almacenamiento de Información no Estructurada” del “Máster en Ingeniería y Ciencia de Datos” de la UNED.

El trabajo consiste en la realización de tres ejercicios con el objetivo de familiarizarse con todos los aspectos de la utilización de bases de datos orientadas a grafos, mediante el uso de la herramienta Neo4j.

La realización de estos ejercicios se entregará en unos ficheros de texto adjuntos en la entrega llamados ejercicio1.txt, ejercicio2.txt y ejercicio3.txt.

Para realizar esta práctica se va a utilizar un sistema basado en virtualización ligera con Docker. Para desplegar dicho sistema, deberemos tener Docker instalado en nuestro ordenador. En este caso, se ha realizado con el sistema operativo MAC OS.

Nota: Nótese que los enunciados de los ejercicios destacan por emplear letra en formato *cursiva*.

# Ejercicios

## Ejercicio 1: Creación de la base de datos.

*Crea una base de datos con los datos que se adjuntan en el archivo grafo.txt disponible en el curso virtual. El procedimiento de despliegue del sistema neo4j se detalla en el apartado 3 y la base de datos se explica en el apartado 4.*

En primer lugar, se ha realizado la ejecución del siguiente comando:

|  |
| --- |
| $ docker run –-publish 7474:7474 –-publish 7687:7687 –-volume=$HOME/neo4j/data:/data neo4j |

Tras esta ejecución, se ha descargado la imagen oficial de Neo4j, y se ha creado un contenedor accesible a través de la dirección http://localhost:7474. Además, crea un volumen para que la base de datos persista fuera del contenedor. A continuación, se muestran las trazas del sistema:

|  |
| --- |
| (base) MacBook-Pro-de-Javier:TP3 - Neo4j javierpiquerasmartinez$ **docker run --publish=7474:7474 --publish=7687:7687 --volume=$HOME/neo4j/data:/data neo4j**  Unable to find image 'neo4j:latest' locally  latest: Pulling from library/neo4j  54fec2fa59d0: Pull complete  b7dd01647a92: Pull complete  793cbc6f8a59: Pull complete  9a08b87df712: Pull complete  27fd00dc5a07: Pull complete  769f827a6cd8: Pull complete  1ea3d537b7db: Pull complete  199ed086dc4a: Pull complete  Digest: sha256:772647cac3fe0e4dbd92ac0d08f2c50dd3be617c04948428b6d828807847fc39  Status: Downloaded newer image for neo4j:latest  Warning: Folder mounted to "/data" is not writable from inside container. Changing folder owner to neo4j.  Directories in use:  home: /var/lib/neo4j  config: /var/lib/neo4j/conf  logs: /logs  plugins: /var/lib/neo4j/plugins  import: /var/lib/neo4j/import  data: /var/lib/neo4j/data  certificates: /var/lib/neo4j/certificates  run: /var/lib/neo4j/run  Starting Neo4j.  2020-05-12 17:36:04.010+0000 INFO ======== Neo4j 4.0.4 ========  2020-05-12 17:36:04.016+0000 INFO Starting...  2020-05-12 17:36:39.193+0000 INFO Bolt enabled on 0.0.0.0:7687.  2020-05-12 17:36:39.193+0000 INFO Started.  2020-05-12 17:36:40.123+0000 INFO Remote interface available at http://localhost:7474/ |

El acceso a dicho contenedor se realizará usando la clase neo4j y la contraseña neo4j como se puede observar en la siguiente Ilustración.

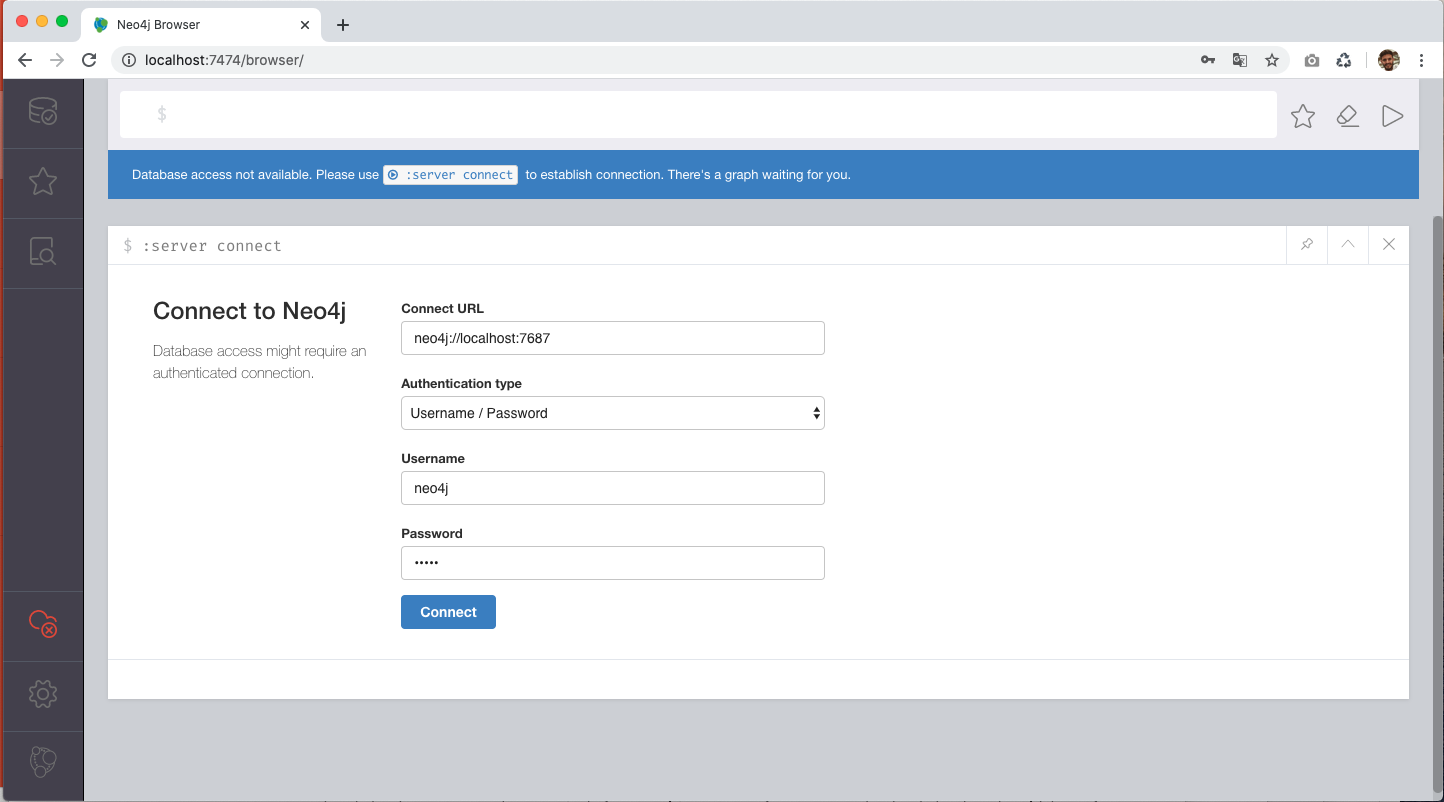


Ilustración 2‑1. Acceso al contenedor Neo4j

Para la creación de la base de datos con los datos del fichero “grafo.txt”, se ejecutan los comandos que este archivo contiene en Neo4j Browser. Se obtiene la confirmación de la ejecución como se muestra a continuación:

Captura de pantalla de un celular

Descripción generada automáticamente

Ilustración 2‑2. Confirmación carga Base de Datos Neo4j

Se comprueba que se ha creado correctamente y se observa la estructura de ésta en el panel izquierdo.

Captura de pantalla de un celular

Descripción generada automáticamente

Ilustración 2‑3. Información de la base de datos tras la creación.

Además, también se puede ver esta estructura con la ejecución del siguiente comando:

|  |
| --- |
| CALL db.schema.visualization() |

El cual proporciona el resultado mostrado en la siguiente Ilustración.

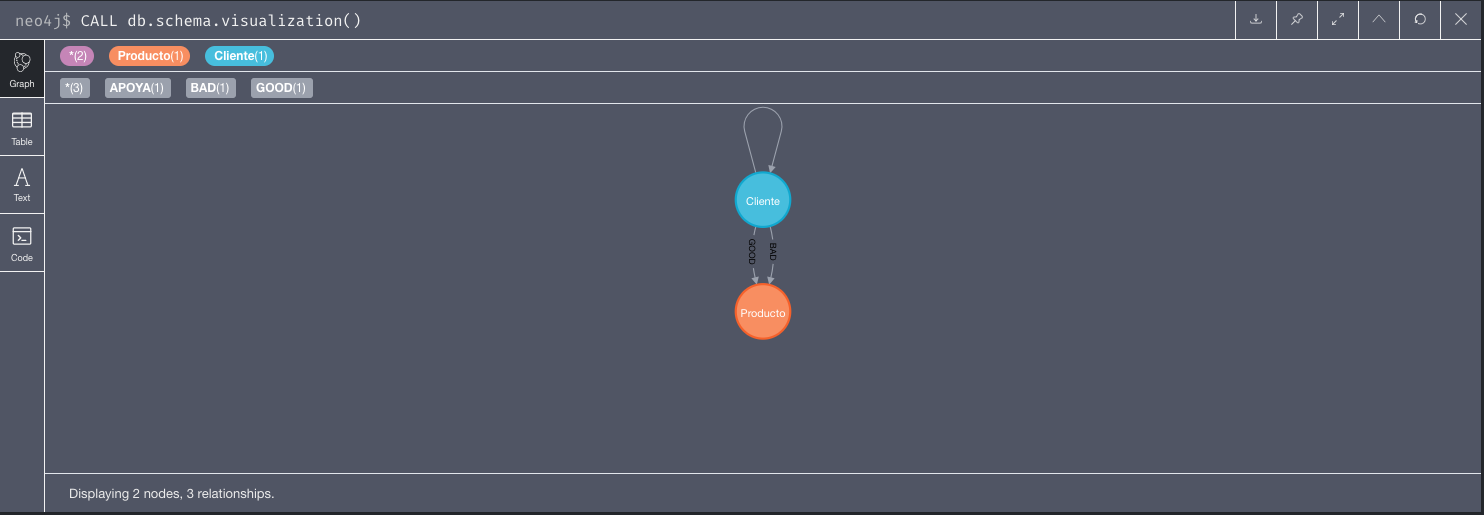


Ilustración 2‑4. Esquema de la base de datos.

## Ejercicio 2: Consultas sobre la base de datos.

*Sobre la base de datos, implementa consultas para conseguir la siguiente información:*

* *Queremos conocer nodos tipo Cliente que hayan emitido una opinión tipo BAD del producto con id=2 (hay 3 clientes con estas características).*
* *Queremos buscar clientes que formen un “circuito” de relaciones de tipo “APOYA”, es decir una secuencia de relaciones, todas en la misma dirección, que acaba y empieza en el mismo Cliente.*
* *Demetrio ha decidido mostrar su apoyo a Genara. Añadir la relación APOYA entre los nodos correspondientes. No se debe suponer que el nodo existe (ni que no existe), solo se debe asegurar que tras la instrucción ambos existen (sin duplicarlos) y con la relación APOYA indicada.*

## Ejercicio 3: Creación de una base de datos propia.

*Crea tu propia base de datos sobre Neo4j, e implementa al menos 2 consultas sobre dichos datos. Se valorará la originalidad y la complejidad de la base de datos y de las consultas creadas.*