



EBook Gratis

APRENDIZAJE neo4j

Free unaffiliated eBook created from
Stack Overflow contributors.

#neo4j

Tabla de contenido

Acerca de.....	1
Capítulo 1: Empezando con neo4j.....	2
Observaciones.....	2
Examples.....	2
Instalación o configuración.....	2
Instalación e inicio de un servidor Neo4j.....	2
Inicie Neo4j desde la consola (sin cabeza, sin servidor web).....	3
Iniciar servidor web Neo4j.....	3
Iniciar servidor web Neo4j.....	3
Eliminar una de las bases de datos.....	3
Cypher Query Language.....	4
RDBMS Vs Graph Database.....	5
Capítulo 2: Cifrar.....	6
Introducción.....	6
Examples.....	6
Creación.....	6
Crear un nodo.....	6
Crear una relacion.....	6
Plantillas de consulta.....	6
Crear un borde.....	6
Supresión.....	6
Borrar todos los nodos.....	6
Eliminar todos los nodos de una etiqueta específica.....	7
Coincidir (capturar grupo) y enlazar nodos emparejados.....	7
Actualizar un nodo.....	7
Eliminar todos los nodos huérfanos.....	7
Capítulo 3: Pitón.....	8
Examples.....	8
Instalar neo4jrestclient.....	8

Conectarse a neo4j.....	8
Crear algunos nodos con etiquetas.....	8
Puede asociar una etiqueta con muchos nodos de una sola vez.....	8
Crear relaciones.....	8
Relaciones bidireccionales.....	8
Partido usando neo4jrestclient.....	8
"db" como se define arriba.....	9
Imprimir resultados.....	9
Salida:.....	9
Creditos.....	10

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [neo4j](#)

It is an unofficial and free neo4j ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official neo4j.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con neo4j

Observaciones

Esta sección proporciona una descripción general de qué es neo4j y por qué un desarrollador puede querer usarlo.

También debe mencionar cualquier tema importante dentro de neo4j, y vincular a los temas relacionados. Dado que la Documentación para neo4j es nueva, es posible que deba crear versiones iniciales de los temas relacionados.

Examples

Instalación o configuración

Vaya a [Instalar Neo4j](#), que debería detectar la plataforma del sistema operativo a través de su navegador web, descargue y siga las instrucciones de instalación habituales para su sistema operativo.

Neo4j fue creado con Java, por lo tanto, se ejecutará en cualquier plataforma con Java instalada, sin embargo, el equipo de Neo4j ha simplificado la instalación al proporcionar paquetes de instalación fáciles para la plataforma popular (por ejemplo, un .dmg para Mac, un .deb para Debian y Ubuntu, un .exe Para plataformas Windows de 64 y 32 bits ...).

Para revisar otras versiones y plataformas disponibles, consulte la [página de otras versiones de Neo4j](#)

Configura Neo4j como un contenedor Docker:

```
## Required : Docker machine, docker cli

# Pull neo4j image from the docker hub
docker pull neo4j

# create the docker container
docker run \
  --publish=7474:7474 --publish=7687:7687 \
  --volume=$HOME/neo4j/data:/data \
  neo4j

# If you are running docker directly on the host (e.g ubuntu, RHEL, CentOS etc)
#   Access the neo4j console at http://localhost:7474
# If you are on OSX/ Windows
#   Access the neo4j console at http://<docker-machine-ip>:7474
```

Instalación e inicio de un servidor Neo4j

Prerrequisitos pasos:

- Instala Java en tu máquina
- Visite el [sitio web de neo4j](#) y haga clic en el enlace "Descargar Community Edition" o visite directamente el [enlace de descarga](#) .
- Descomprima el archivo descargado .tar en su directorio personal

Inicie Neo4j desde la consola (sin cabeza, sin servidor web)

- Visite el subdirectorio `/bin` de la carpeta extraída y ejecute en la terminal `./neo4j console`
- Ahora puedes ejecutar consultas neo4j en el terminal.

Iniciar servidor web Neo4j

- Visite el subdirectorio `/bin` de la carpeta extraída y ejecute en la terminal `./neo4j start`
- Visite <http://localhost:7474/>
- Solo la primera vez, tendrá que iniciar sesión con la cuenta predeterminada y cambiar la contraseña predeterminada. A partir de la versión 3.0.3 de la comunidad, el nombre de usuario y la contraseña predeterminados son neo4j y neo4j.
- Ahora puede insertar consultas Neo4j en la consola provista en su navegador web e investigar visualmente los resultados de cada consulta.

Iniciar servidor web Neo4j

Cada servidor Neo4j actualmente (en la edición comunitaria) puede alojar una sola base de datos Neo4j, así que para configurar una nueva base de datos:

- Visite el subdirectorio `/bin` y ejecute `./neo4j stop` para detener el servidor
- Visite el subdirectorio `/conf` y edite el archivo `neo4j.conf` , cambiando el valor del parámetro `dbms.active_database` al nombre de la nueva base de datos que desea crear.
- Visite nuevamente el subdirectorio `/bin` y ejecute `./neo4j start`
- El servidor web ha comenzado de nuevo con la nueva base de datos vacía. Puede visitar nuevamente <http://localhost:7474/> para trabajar con la nueva base de datos.
- La base de datos creada se encuentra en el subdirectorio `/data/databases` , en una carpeta con el nombre especificado en el parámetro `dbms.active_database` .

Eliminar una de las bases de datos.

- Asegúrese de que el servidor Neo4j no se está ejecutando; `./neo4j status` al subdirectorio `/bin` y ejecute el `./neo4j status` . Si el mensaje de salida muestra que el servidor se está ejecutando, también ejecute `./neo4j stop` .

- Luego vaya al subdirectorio / datos / bases de datos y elimine la carpeta de la base de datos que desea eliminar.

Cypher Query Language

Este es el lenguaje de consulta de Cypher, Neo4j. En muchos sentidos, Cypher es similar a SQL si está familiarizado con él, excepto que SQL se refiere a elementos almacenados en una tabla, mientras que Cypher se refiere a elementos almacenados en un gráfico.

Primero, debemos comenzar por aprender cómo crear un gráfico y agregar relaciones, ya que eso es esencialmente de lo que se trata Neo4j.

```
CREATE (ab:Object { age: 30, destination: "England", weight: 99 })
```

- **Usas CREATE** para crear datos
- Para indicar un nodo, utiliza paréntesis: ()
- La parte ab: Object se puede desglosar de la siguiente manera: una variable 'ab' y la etiqueta 'Object' para el nuevo nodo. Tenga en cuenta que la variable puede ser cualquier cosa, pero debe ser coherente en una línea de Cypher Query
- Para agregar propiedades al nodo, use paréntesis: {} corchetes

A continuación, aprenderemos sobre cómo encontrar PARTIDOS.

```
MATCH (abc:Object) WHERE abc.destination = "England" RETURN abc;
```

MATCH especifica que desea buscar un patrón de relación / nodo determinado (abc: Object) se refiere a un patrón de nodo (con objeto de etiqueta) que almacena las coincidencias en la variable abc. Puedes pensar en esta línea entera como la siguiente

```
abc = find the matches that is an Object WHERE the destination is England.
```

En este caso, DONDE agrega una restricción que es que el destino debe ser Inglaterra. Debe incluir una devolución al final de todas las consultas de MATCH (neo4j no aceptará solo una coincidencia ... su consulta siempre debe devolver algún valor [esto también depende del tipo de consulta que esté escribiendo ... hablaremos más sobre esto más adelante a medida que introducimos los otros tipos de consultas que puede hacer].

La siguiente línea se explicará en el futuro, después de que veamos algunos elementos más del lenguaje de consulta Cypher. ¡Esto es para darle una idea de lo que podemos hacer con este idioma! A continuación, encontrará un ejemplo que muestra el reparto de películas cuyo título comienza con 'T'

```
MATCH (actor:Person)-[:ACTED_IN]->(movie:Movie)
WHERE movie.title STARTS WITH "T"
RETURN movie.title AS title, collect(actor.name) AS cast
ORDER BY title ASC LIMIT 10;
```

Puede encontrar una lista completa de los comandos y su sintaxis en la [Tarjeta de referencia de](#)

cifrado oficial de [Neo4j](#) [aquí](#) .

RDBMS Vs Graph Database

RDBMS	Base de datos gráfica
Mesas	Graficas
Filas	Nodos
Columnas y datos	Propiedades y sus valores.
Restricciones	Relaciones
Se une	Travesía

Lea Empezando con neo4j en línea: <https://riptutorial.com/es/neo4j/topic/945/empezando-con-neo4j>

Capítulo 2: Cifrar

Introducción

Cypher es el lenguaje de consulta utilizado por Neo4j. Utiliza Cypher para realizar tareas y coincidencias con un gráfico Neo4j.

Cypher está "inspirado en SQL" y está diseñado para ser intuitivo en la forma en que describe las relaciones, es decir, típicamente el dibujo del patrón se verá similar a la representación Cypher del patrón.

Examples

Creación

Crear un nodo

```
CREATE (neo:Company) //create node with label 'Company'

CREATE (neo:Company {name: 'Neo4j', hq: 'San Mateo'}) //create node with properties
```

Crear una relacion

```
CREATE (beginning_node)-[:edge_name{Attribute:1, Attribute:'two'}]->(ending_node)
```

Plantillas de consulta

Al ejecutar neo4j localmente, en la GUI del navegador (predeterminado: <http://localhost:7474/browser/>), puede ejecutar el siguiente comando para obtener una paleta de consultas.

```
:play query template
```

Esto le ayuda a comenzar a crear y fusionar nodos y relaciones escribiendo consultas.

Crear un borde

```
CREATE (beginning_node)-[:edge_name{Attribute:1, Attribute:'two'}]->(ending_node)
```

Supresión

Borrar todos los nodos

```
MATCH (n)
DETACH DELETE n
```

DETACH no funciona en versiones anteriores (menos de 2.3), para versiones anteriores.

```
MATCH (n)
OPTIONAL MATCH (n)-[r]-()
DELETE n, r
```

Eliminar todos los nodos de una etiqueta específica

```
MATCH (n:Book)
DELETE n
```

Coincidir (capturar grupo) y enlazar nodos emparejados

```
Match (node_name:node_type {}), (node_name_two:node_type_two {})
CREATE (node_name)-[:edge_name{}]->(node_name_two)
```

Actualizar un nodo

```
MATCH (n)
WHERE n.some_attribute = "some identifier"
SET n.other_attribute = "a new value"
```

Eliminar todos los nodos huérfanos

Los nodos / vértices huérfanos son aquellos que carecen de todas las relaciones / bordes.

```
MATCH (n)
WHERE NOT (n)--()
DELETE n
```

Lea Cifrar en línea: <https://riptutorial.com/es/neo4j/topic/3669/cifrar>

Capítulo 3: Pitón

Examples

Instalar neo4jrestclient

```
pip install neo4jrestclient
```

Conectarse a neo4j

```
from neo4jrestclient.client import GraphDatabase
db = GraphDatabase("http://localhost:7474", username="neo4j", password="mypass")
```

Crear algunos nodos con etiquetas.

```
user = db.labels.create("User")
u1 = db.nodes.create(name="user1")
user.add(u1)
u2 = db.nodes.create(name="user2")
user.add(u2)
```

Puede asociar una etiqueta con muchos nodos de una sola vez

```
Language = db.labels.create("Language")
b1 = db.nodes.create(name="C++")
b2 = db.nodes.create(name="Python")
beer.add(b1, b2)
```

Crear relaciones

```
u1.relationships.create("likes", b1)
u1.relationships.create("likes", b2)
u2.relationships.create("likes", b1)
```

Relaciones bidireccionales

```
u1.relationships.create("friends", u2)
```

Partido usando neo4jrestclient

```
from neo4jrestclient import client

q = 'MATCH (u:User)-[r:likes]->(m:language) WHERE u.name="Marco" RETURN u, type(r), m'
```

"db" como se define arriba

```
results = db.query(q, returns=(client.Node, str, client.Node))
```

Imprimir resultados

```
for r in results:
    print("(%s)-[%s]->(%s)" % (r[0]["name"], r[1], r[2]["name"]))
```

Salida:

```
(Marco)-[likes]->(C++)
(Marco)-[likes]->(Python)
```

Lea Pitón en línea: <https://riptutorial.com/es/neo4j/topic/5990/piton>

Creditos

S. No	Capítulos	Contributors
1	Empezando con neo4j	Andrew Lank , Community , cs_user2017 , Dimos , Frank Pavageau , Prosen Ghosh , TinkerBotFoo , wintersolider
2	Cifrar	frant.hartm , Govind Singh , JOG , Liam , Nicole White , Renato Dinhani , SerialDev , TJ Walker
3	Pitón	SerialDev