

 eBook Gratuit

# APPRENEZ

---

## neo4j

eBook gratuit non affilié créé à partir des  
**contributeurs de Stack Overflow.**

#neo4j

# Table des matières

À propos.....	1
<b>Chapitre 1: Démarrer avec neo4j.....</b>	<b>2</b>
Remarques.....	2
Exemples.....	2
Installation ou configuration.....	2
Installation et démarrage d'un serveur Neo4j.....	2
<b>Démarrer Neo4j à partir de la console (sans tête, sans serveur Web).....</b>	<b>3</b>
<b>Démarrer le serveur Web Neo4j.....</b>	<b>3</b>
<b>Démarrer le serveur Web Neo4j.....</b>	<b>3</b>
<b>Supprimer l'une des bases de données.....</b>	<b>3</b>
Langage de requête Cypher.....	4
SGBDR Vs Graph Database.....	5
<b>Chapitre 2: Python.....</b>	<b>6</b>
Exemples.....	6
Installer neo4jrestclient.....	6
Connectez-vous à neo4j.....	6
Créer des nœuds avec des étiquettes.....	6
<b>Vous pouvez associer une étiquette avec plusieurs nœuds en une seule fois.....</b>	<b>6</b>
Créer des relations.....	6
<b>Relations bidirectionnelles.....</b>	<b>6</b>
Match en utilisant neo4jrestclient.....	6
<b>"db" comme défini ci-dessus.....</b>	<b>7</b>
<b>Imprimer les résultats.....</b>	<b>7</b>
<b>Sortie:.....</b>	<b>7</b>
<b>Chapitre 3: Zéro.....</b>	<b>8</b>
Introduction.....	8
Exemples.....	8
Création.....	8
<b>Créer un noeud.....</b>	<b>8</b>

<b>Créer une relation</b> .....	<b>8</b>
<b>Modèles de requête</b> .....	<b>8</b>
Créer un bord.....	8
Effacement.....	8
<b>Supprimer tous les nœuds</b> .....	<b>8</b>
<b>Supprimer tous les nœuds d'une étiquette spécifique</b> .....	<b>9</b>
Match (groupe de capture) et nœuds correspondants.....	9
Mettre à jour un noeud.....	9
Supprimer tous les noeuds orphelins.....	9
<b>Crédits</b> .....	<b>10</b>

---

# À propos

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](mailto:info@zzzprojects.com)

---

# Chapitre 1: Démarrer avec neo4j

## Remarques

Cette section fournit une vue d'ensemble de ce que neo4j est et pourquoi un développeur peut vouloir l'utiliser.

Il devrait également mentionner tous les grands sujets dans neo4j, et établir un lien avec les sujets connexes. La documentation de neo4j étant nouvelle, vous devrez peut-être créer des versions initiales de ces rubriques connexes.

## Exemples

### Installation ou configuration

Allez sur [Installer Neo4j](#) qui devrait détecter la plate-forme du système d'exploitation via votre navigateur Web, téléchargez et suivez les instructions d'installation habituelles pour votre système d'exploitation.

Neo4j a été créé avec Java, donc il s'exécutera sur n'importe quelle plate-forme avec Java installé. Cependant, l'équipe de Neo4j a simplifié l'installation en fournissant des paquets d'installation faciles pour une plate-forme populaire. pour les plates-formes Windows 64 et 32 bits ...).

*Pour consulter les autres versions et plates-formes disponibles, consultez la page [Autres versions de Neo4j](#)*

### Configurez Neo4j en tant que conteneur 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
```

### Installation et démarrage d'un serveur Neo4j

Étapes préalables:

- Installer Java sur votre machine
- Visitez le [site Web neo4j](#) et cliquez sur le lien "Télécharger l'édition communautaire" ou visitez directement le [lien de téléchargement](#) .
- Décompressez le fichier téléchargé .tar dans votre répertoire personnel

---

## Démarrer Neo4j à partir de la console (sans tête, sans serveur Web)

- Visitez le sous-répertoire `/bin` du dossier extrait et exécutez la `./neo4j console terminal`  
`./neo4j console`
- Vous pouvez maintenant exécuter des requêtes neo4j dans le terminal

---

## Démarrer le serveur Web Neo4j

- Visitez le sous-répertoire `/bin` du dossier extrait et exécutez le terminal `./neo4j start`
- Visitez <http://localhost:7474/>
- Seulement la première fois, vous devrez vous connecter avec le compte par défaut et changer le mot de passe par défaut. À partir de la version 3.0.3 de la communauté, le nom d'utilisateur et le mot de passe par défaut sont neo4j et neo4j.
- Vous pouvez maintenant insérer des requêtes Neo4j dans la console fournie dans votre navigateur Web et examiner visuellement les résultats de chaque requête.

---

## Démarrer le serveur Web Neo4j

Chaque serveur Neo4j actuellement (dans l'édition communautaire) peut héberger une seule base de données Neo4j, donc pour installer une nouvelle base de données:

- Visitez le sous-répertoire `/bin` et exécutez `./neo4j stop` pour arrêter le serveur
- Rendez-vous dans le sous-répertoire `/conf` et modifiez le fichier `neo4j.conf` la valeur du paramètre `dbms.active_database` sur le nom de la nouvelle base de données que vous souhaitez créer.
- Visitez à nouveau le sous-répertoire `/bin` et exécutez `./neo4j start`
- Le serveur Web a redémarré avec la nouvelle base de données vide. Vous pouvez visiter à nouveau <http://localhost:7474/> pour travailler avec la nouvelle base de données.
- La base de données créée se trouve dans le sous-répertoire `/data/databases` , sous un dossier portant le nom spécifié dans le paramètre `dbms.active_database` .

---

## Supprimer l'une des bases de données

- Assurez-vous que le serveur Neo4j ne fonctionne pas. allez dans le sous-répertoire `/bin` et exécutez le `./neo4j status` . Si le message de sortie indique que le serveur est en cours

d'exécution, exécutez également `./neo4j stop`.

- Ensuite, allez dans le sous-répertoire `/ data / databases` et supprimez le dossier de la base de données à supprimer.

## Langage de requête Cypher

C'est le langage d'interrogation de Cypher, Neo4j. À bien des égards, Cypher est similaire à SQL si vous le connaissez, sauf que SQL fait référence à des éléments stockés dans une table alors que Cypher fait référence à des éléments stockés dans un graphique.

Tout d'abord, nous devrions commencer par apprendre à créer un graphique et à ajouter des relations, car c'est essentiellement ce que propose Neo4j.

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

- Vous utilisez **CREATE** pour créer des données
- Pour indiquer un nœud, vous utilisez une parenthèse: `()`
- La partie `ab: Object` peut être décomposée comme suit: une variable 'ab' et l'étiquette 'Object' pour le nouveau nœud. Notez que la variable peut être n'importe quoi, mais vous devez être cohérent dans une ligne de requête Cypher
- Pour ajouter des propriétés au nœud, utilisez des crochets: `{}` entre crochets

Ensuite, nous allons apprendre à trouver des **MATCH**

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

**MATCH** spécifie que vous souhaitez rechercher un certain nœud / modèle de relation (`abc: Object`) fait référence à un nœud Pattern (avec label `Object`) qui stocke les correspondances dans la variable `abc`. Vous pouvez penser à cette ligne entière comme suit

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

Dans ce cas, **WHERE** ajoute une contrainte: la destination doit être l'Angleterre. Vous devez inclure un retour à la fin de toutes les requêtes **MATCH** (neo4j n'acceptera pas juste un match ... votre requête doit toujours renvoyer une valeur [cela dépend aussi du type de requête que vous écrivez ... nous en parlerons plus ceci plus tard que nous introduisons les autres types de requêtes que vous pouvez faire]).

La ligne suivante sera expliquée dans le futur, après avoir passé en revue quelques autres éléments du langage d'interrogation Cypher. C'est pour vous donner un avant-goût de ce que nous pouvons faire avec cette langue! Ci-dessous, vous trouverez un exemple qui obtient la distribution de films dont le titre commence par '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;
```

Une liste complète des commandes et de leur syntaxe peut être trouvée sur la [carte de référence officielle de la Neo4j Cypher ici](#) .

## SGBDR Vs Graph Database

SGBDR	Base de données graphique
les tables	Des graphiques
Des rangées	Les nœuds
Colonnes et données	Propriétés et ses valeurs
Contraintes	Des relations
Joint	Traversal

Lire Démarrer avec neo4j en ligne: <https://riptutorial.com/fr/neo4j/topic/945/demarrer-avec-neo4j>

---

# Chapitre 2: Python

## Exemples

### Installer neo4jrestclient

```
pip install neo4jrestclient
```

### Connectez-vous à neo4j

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

### Créer des nœuds avec des étiquettes

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

---

## Vous pouvez associer une étiquette avec plusieurs nœuds en une seule fois

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

### Créer des relations

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

---

## Relations bidirectionnelles

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

### Match en utilisant neo4jrestclient

```
from neo4jrestclient import client

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

---

## "db" comme défini ci-dessus

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

---

## Imprimer les résultats

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

---

## Sortie:

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

Lire Python en ligne: <https://riptutorial.com/fr/neo4j/topic/5990/python>

---

# Chapitre 3: Zéro

## Introduction

Cypher est le langage de requête utilisé par Neo4j. Vous utilisez Cypher pour effectuer des tâches et des correspondances avec un graphique Neo4j.

Cypher est "inspiré par SQL" et est conçu pour être intuitif dans la manière dont vous décrivez les relations, c'est-à-dire que le dessin du motif ressemblera généralement à la représentation Cypher du motif.

## Exemples

### Création

---

## Créer un noeud

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

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

---

## Créer une relation

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

---

## Modèles de requête

En exécutant localement neo4j, dans l'interface graphique du navigateur (par défaut: [http://localhost: 7474 / browser /](http://localhost:7474/browser/)), vous pouvez exécuter la commande suivante pour obtenir une palette de requêtes.

```
:play query template
```

Cela vous aide à créer et à fusionner des nœuds et des relations en tapant des requêtes.

### Créer un bord

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

### Effacement

---

# Supprimer tous les nœuds

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

`DETACH` ne fonctionne pas dans les anciennes versions (moins de 2.3), pour les versions précédentes

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

## Supprimer tous les nœuds d'une étiquette spécifique

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

### Match (groupe de capture) et nœuds correspondants

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

### Mettre à jour un noeud

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

### Supprimer tous les noeuds orphelins

Les nœuds / sommets orphelins sont ceux qui manquent de toutes les relations / arêtes.

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

Lire Zéro en ligne: <https://riptutorial.com/fr/neo4j/topic/3669/zero>

# Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec neo4j	<a href="#">Andrew Lank</a> , <a href="#">Community</a> , <a href="#">cs_user2017</a> , <a href="#">Dimos</a> , <a href="#">Frank Pavageau</a> , <a href="#">Prosen Ghosh</a> , <a href="#">TinkerBotFoo</a> , <a href="#">wintersolider</a>
2	Python	<a href="#">SerialDev</a>
3	Zéro	<a href="#">frant.hartm</a> , <a href="#">Govind Singh</a> , <a href="#">JOG</a> , <a href="#">Liam</a> , <a href="#">Nicole White</a> , <a href="#">Renato Dinhani</a> , <a href="#">SerialDev</a> , <a href="#">TJ Walker</a>