



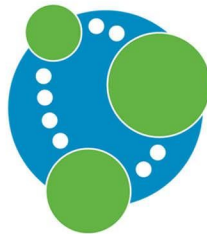
ESCUELA POLITÉCNICA  
SUPERIOR DE CÓRDOBA  
Universidad de Córdoba

EP  
SC



UNIVERSIDAD DE CÓRDOBA

# *Neo4j*



neo4j

## Comandos utilizados en todos los tutoriales



## CREATE (para nodos)

---

### **Crear Nodo simple**

```
CREATE (nodo)
```

### **Crear nodo con etiqueta Persona**

```
CREATE (Maria:Persona)
```

### **Crear nodo con etiqueta Persona y 2 propiedad (nombre,edad):**

```
CREATE (Juan:Persona{nombre:'Juan',edad:22})
```

### **Crear nodo con 2 etiquetas (Persona y Bombero) y 2 propiedades (nombre y nacionalidad)**

```
CREATE (Paco:Persona:Bombero{nombre:'Paco',nacionalidad:'española'})
```

## CREATE (para relaciones)

---

### **Forma primera de crear una relación simple entre 2 nodos**

```
CREATE (Paco:Persona{nombre:'Paco'})  
CREATE (Maria:Persona{nombre:'María'})  
CREATE (Paco)-[:FRIEND_OF]->(Maria)
```

### **Segunda forma de crear una relación simple entre 2 nodos (cada línea de comando se crea en una ejecución diferente):**

#### **1. Paso**

```
CREATE (Rosa:Persona{nombre:'Rosa'})
```

#### **2. Paso**

```
CREATE (Raquel:Persona{nombre:'Raquel'})
```

### 3. Paso

```
MATCH (Raquel:Persona{nombre:'Raquel'}) MATCH  
(Rosa:Persona{nombre:'Rosa'}) CREATE (Raquel)-[:Familia_de]->(Rosa)
```

#### **Crear una relación con una propiedad entre nodos ya existentes**

```
MATCH (Rosa:Persona{nombre:'Rosa'}) MATCH (Maria:Persona{nombre:'María'})  
CREATE (Maria)-[:FRIEND_OF{role:'Amigas_trabajo'}]->(Rosa)
```

## MATCH

---

#### **Mostrar toda la base de datos:**

```
MATCH (n) RETURN n
```

#### **Mostrar todos los nodos de etiqueta 'Árbol':**

```
MATCH (n:Árbol) RETURN n
```

#### **Mostrar los atributos 'atributo' de los nodos de etiqueta 'Árbol':**

```
MATCH (n:Árbol) RETURN n.atributo
```

#### **Mostrar los atributos 'atributo' de los nodos de etiqueta 'Árbol':**

```
MATCH (n:Árbol) RETURN n.atributo
```

#### **Mostrar los atributos 'atributo' de los nodos de etiqueta 'Árbol':**

```
MATCH (n:Árbol) RETURN n.atributo
```

#### **Mostrar los nodos de nombre 'acacia' de etiqueta 'Árbol':**

```
MATCH (n:Árbol) WHERE n.nombre='acacia' RETURN n
```

**Mostrar los nodos de fecha de plantación distinta de 20 días, de etiqueta 'Árbol':**

```
MATCH (n:Árbol) WHERE n.fecha_plantación<>20 RETURN n
```

**Mostrar los nodos de fecha de plantación de 20 o 22 días, de etiqueta 'Árbol':**

```
MATCH (n:Árbol) WHERE n.fecha_plantación IN [20,22] RETURN n
```

**Mostrar los nodos cuyo nombre empiece por 'a', de etiqueta 'Árbol':**

```
MATCH (n:Árbol) WHERE n.nombre STARTS WITH 'a' RETURN n
```

**Mostrar los nodos cuyo nombre contenga la letra 'a', de etiqueta 'Árbol':**

```
MATCH (n:Árbol) WHERE n.nombre CONTAINS 'a' RETURN n
```

**Mostrar los nodos cuyo nombre contenga la letra 'a' y cuya fecha de plantación sea de 31 días, de etiqueta 'Árbol':**

```
MATCH (n:Árbol) WHERE n.nombre CONTAINS 'a' AND n.fecha_plantación=31  
RETURN n
```

**Mostrar los nodos de etiqueta 'Árbol', ordenados por el nombre alfabéticamente de forma ascendente:**

```
MATCH (n:Árbol) ORDER BY n.nombre ASC RETURN n
```

**Mostrar los nodos de etiqueta 'Árbol', estableciendo un límite de 2 resultados:**

```
MATCH (n:Árbol) RETURN n LIMIT 2
```

**Mostrar los nodos de etiqueta 'Árbol', estableciendo que no se puedan mostrar resultados repetidos:**

```
MATCH (n:Árbol) RETURN DISTINCT n
```

**Mostrar el número de nodos de etiqueta 'Árbol':**

```
MATCH (n:Árbol) RETURN count(*)
```

**Mostrar las relaciones entre nodos de etiqueta 'Árbol', en el que el originario de la relación se un nodo de nombre 'acacia':**

```
MATCH relation=(n1:Árbol)-[r:relacionado_con]->(n2:Árbol) WHERE  
n1.nombre='acacia' RETURN relation
```

**Mostrar las relaciones entre nodos de etiqueta 'Árbol', en el que el originario de la relación se un nodo de nombre 'acacia':**

```
MATCH relation=(n1:Árbol)-[r:relacionado_con]->(n2:Árbol) WHERE  
n1.nombre='acacia' RETURN relation
```

**Mostrar las relaciones entre nodos de etiqueta 'Árbol', en el que el originario de la relación se un nodo de nombre 'acacia'. Además mostrar las que se encuentren a dos pasos:**

```
MATCH relation=(n1:Árbol)-[r:relacionado_con*2]->(n2:Árbol) WHERE  
n1.nombre='acacia' RETURN relation
```

## SET, REMOVE, DELETE

---

**Base de Datos para el tutorial**

```
CREATE (Maria:Persona{nombre:'Maria',edad:20})  
CREATE (Paco:Persona{nombre:'Paco',edad:23})  
CREATE (Rosa:Persona{nombre:'Rosa'})  
CREATE (Juan:Persona{nombre:'Juan'})  
CREATE (Rosa)-[:Family_of]->(Maria)  
CREATE (Paco)-[:Friend_of]->(Rosa)
```

- **SET**

**Añadir la propiedad apellidos al nodo de nombre María**

```
MATCH (p:Persona) WHERE p.nombre='Maria' SET p.apellidos='Jiménez'
```

**Eliminar la propiedad anterior con SET:**

```
MATCH (p:Persona) WHERE p.nombre='Maria' SET p.apellidos=NULL
```

**Otra forma de añadir la propiedad apellidos al nodo de nombre María:**

```
MERGE (p:Persona{nombre:'Maria'}) SET p.apellidos='Jiménez'
```

**Otra forma de eliminar la propiedad anterior con SET:**

```
MERGE (p:Persona{nombre:'Maria'}) SET p.apellidos=NULL
```

**Añadir propiedad a la relación de amistad entre Paco y Rosa**

```
MERGE (Paco)-[r:Friend_of]->(Rosa) SET r.tipo='Amigos_del_Trabajo'
```

**Eliminar la propiedad anterior de una relación con SET:**

```
MERGE (Paco)-[r:Friend_of]->(Rosa) SET r.tipo=NULL
```

- **REMOVE**

**Eliminar la etiqueta de Persona del nodo con nombre Juan:**

```
MATCH (p:Persona) WHERE p.nombre='Juan' REMOVE p:Persona
```

**Eliminar la propiedad edad del nodo de Persona cuya edad sea igual a 20:**

```
MATCH (p:Persona) WHERE p.edad=20 REMOVE p.edad
```

- **DELETE**

**Para borrar el nodo con nombre María pero por separado:**

1. **Borramos las relaciones que tenga. En nuestro caso la relación de familia con Rosa**

```
MATCH (Maria)-[r:Family_of]->(Rosa) DELETE r
```

2. **Eliminamos ya el nodo Maria**

```
MATCH (p:Persona) WHERE p.nombre='Maria' DELETE p
```

**Eliminar el nodo con nombre Paco borrando a la vez las relaciones y el nodo:**

```
MATCH (p:Persona) WHERE p.nombre='Paco' DETACH DELETE p
```

**Borrar toda la Base de Datos:**

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