链滴

Neo4j 常用语句

作者：Gakkiyomi2019
原文链接：https://ld246.com/article/1599925874551
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

Neo4j 常用语句,不一定全但都是自己使用过的，会长期更新。

给节点的某个属性添加唯一性约束

CREATE CONSTRAINT ON (n :Gakkiyomi) ASSERT n.name IS UNIQUE;

根据两个已有节点创建一条关系并返回

MATCH (a),(b) WHERE a.name="fangc" AND b.name="ssg229" CREATE (a) -[x:Layer3 {vlan: 13 }]-> (b) return x;

节点添加新属性

MATCH (n:Gakkiyomi) WHERE n.id = "a22" SET n.sex = "男" RETURN n

删除指定关系

MATCH (p1:Gakkiyomi)-[r:Friend {id:"349f8fa3-65b7-4c53-b047-1d6c3aa5ec8c"}]-(p2:Firewall)

DELETE r

修改节点label

match(n:oldlabel) set n:newlabel remove n:oldlabel

修改关系label

MATCH p=(n:User)-[r:subord]->(e:department) create(n)-[r2:subord_new]->(e) set r2 = r with

r delete r

根据默认生成id删除节点

MATCH (r),(b) WHERE id(r) = 10 AND id(b) = 9 Delete r,b

查询一条关系(返回 节点-关系-节点)

MATCH p=()-[r:Friend {id:"2ec0ddde-0eb1-4f6b-a9e4-094cfbdfc694"}]->() RETURN p

查询一条关系(只返回关系)

MATCH p=()-[r:Friend]->() RETURN r

查询一条关系

MATCH p=(a)-[r:Friend]->(b) with p as ps, labels(a) as x,labels(b) as y return ps,x,y

查询label名

MATCH (r:Firewall) RETURN distinct labels(r)

查询两点之间的最短路径 (3 为在路径深度为3以内中查找)

match(n{name:"哈士奇"}),(m{name:"fangc"})with n,m match p=shortestpath((n)-[*]->(m)) retur
p;

match(n{name:"哈士奇"}),(m{name:"ssg229"})with n,m match p=shortestpath((n)-[*..3]-(m)) ret
rn p;

shortestpath查询一条

allshortestpath查询所有

变长路径检索

变长路径的表示方式是：[*N...M]，N和M表示路径长度的最小值和最大值。

(a)-[*2]->(b)：表示路径长度为2，起始节点是a，终止节点是b；

(a)-[*3...5]->(b)：表示路径长度的最小值是3，最大值是5，起始节点是a，终止节点是b；

(a)-[*...5]->(b)：表示路径长度的最大值是5，起始节点是a，终止节点是b；

(a)-[*3...]->(b)：表示路径长度的最小值是3，起始节点是a，终止节点是b；

(a)-[*]->(b)：表示不限制路径长度，起始节点是a，终止节点是b；

查询两点之间的所有路径

MATCH p=(a)-[*]->(b)
RETURN p

查询数组里的属性 [1,2,4,5]

```
match (n) where 5 in n.ip return n
```

修改节点属性

```
MATCH (a:Ta{names:"afaf"})
SET a.names="a"
return a
```

修改节点属性名称

```
match(n) set n.propertyNew=n.propertyOld remove n.propertyOld
```

查询多label多条件

```
match (n) where any(label in labels(n) WHERE label in ['HDSStorage','BrocadePort']) and '192.68.1.106' in n.ip or n.domain = '28' return n
```

查询所有节点的属性

```
match (n) return distinct keys(n)
```