

# Neo4j cypher queries

## General queries

### How many nodes are there:

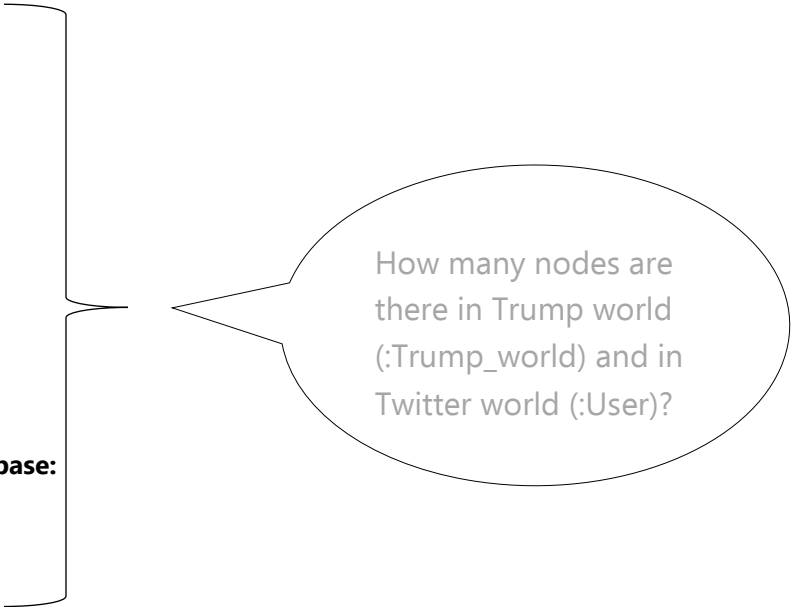
```
MATCH (n)
RETURN COUNT (n)
```

### How many relationships are there:

```
MATCH ()-[r]-()
RETURN COUNT (r)
```

### Find the unique labels that appear in the database:

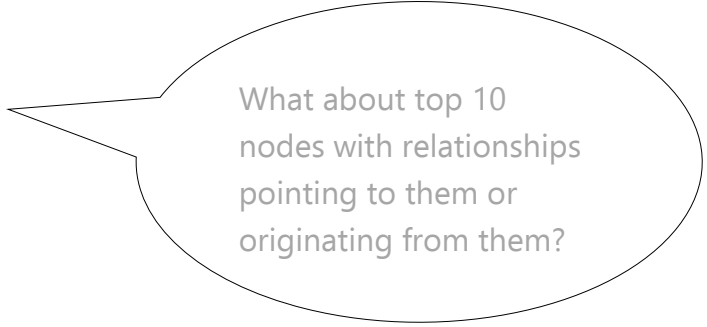
```
MATCH (n)
RETURN DISTINCT LABELS (n)
```



How many nodes are there in Trump world (:Trump\_world) and in Twitter world (:User)?

### TOP 10 nodes with the most number of relationships that they have in general:

```
MATCH (n)-[r]-()
RETURN n, COUNT(r) AS rel_count
ORDER BY rel_count DESC
LIMIT 10
```

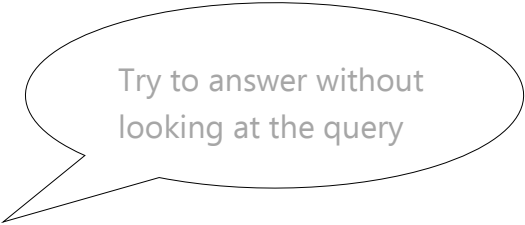


What about top 10 nodes with relationships pointing to them or originating from them?

## Twitter world

### Top 10 Users by number of relationships that are following someone:

```
MATCH (n:User)-[r:FOLLOWS]->()
RETURN n, COUNT(r) AS num
ORDER BY num DESC
LIMIT 10
```



Try to answer without looking at the query

### Top 10 Users by number of relationships that are followed by someone:

↓

```
MATCH (n:User)<-[r:FOLLOWS]-()
RETURN n, COUNT(r) AS num
ORDER BY num DESC
```

LIMIT 10

## Trump world

### Find shortest path between Putin and Trump:

```
MATCH (vp:Trump_world {name:"VLADIMIR PUTIN"}),
      (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (vp)-[*..]-(dt) )
WHERE dt.verified = True
RETURN path
```

Are there some intermediary connections which are not shown?

### Find all closest connections of Trump that are on most powerful list:

```
MATCH (u:Trump_world), (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (u)-[*..1]-(dt) )
WHERE dt.verified = True AND EXISTS(u.rank_powerful)
RETURN path
```

Try to answer without looking at the query

### Find all closest connections of Trump that are on the richest list:

```
MATCH (u:Trump_world), (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (u)-[*..1]-(dt) )
WHERE dt.verified = True AND EXISTS(u.rank)
RETURN path
```

### Find all connections that are top 20 richest and most powerful:

```
MATCH (u:Trump_world), (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (u)-[*..]-(dt) )
WHERE dt.verified = True AND u.rank <=20 AND u.rank_powerful <=20
RETURN path
```

Try to answer without looking at the query

### Find 3-level closest connections that are top 20 richest:

```
MATCH (u:Trump_world), (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (u)-[*..3]-(dt) )
WHERE dt.verified = True AND u.rank <=20
RETURN path
```

## Mix of the worlds

**Find all trump\_world members that have a twitter account in twitter world:**

```
MATCH (n)-[:HAS_TWITTER]-()  
RETURN DISTINCT n
```

**Which twitter mentioned users exist in trump world:**

```
MATCH (x)-[t:HAS_TWITTER]->(n:User)<-[r:MENTIONS]-()  
RETURN DISTINCT n, t, x  
LIMIT 10
```



Try to answer without  
looking at the query

**Which top 10 mentioned users exist in trump world:**

```
MATCH (x)-[t:HAS_TWITTER]->(n:User)<-[r:MENTIONS]-()  
RETURN DISTINCT n, t, x, COUNT(r) AS num  
ORDER BY num DESC  
LIMIT 10
```

## For those who want more

**Find the unique relationships that appear in the database:**

```
MATCH ()-[r]-()  
RETURN DISTINCT type(r)
```

**Find all trump impersonators:**

```
MATCH (u:User)  
WHERE u.name =~ "(?i)Donald J. Trump"  
RETURN u
```

**TOP 10 mentioned users:**

```
MATCH (n:User)<-[r:MENTIONS]-()  
RETURN n, COUNT(r) AS num  
ORDER BY num DESC  
LIMIT 10
```

**Top 10 users with the most connections in trump world (are there any where trump is not first connection ? NO):**

```
MATCH (n:Trump_world)-[r:IS_CONNECTED]->()
RETURN n, COUNT(r) AS num
ORDER BY num DESC
LIMIT 10
```

**Compare to top 10 users with the most connections in trump world that have twitter:**

```
MATCH (u:User)-[t:HAS_TWITTER]-(n:Trump_world)-[r:IS_CONNECTED]->()
RETURN DISTINCT n, COUNT(r) AS num
ORDER BY num DESC
LIMIT 10
```

**Find N closest connections that are top 20 most powerful:**

```
MATCH (u:Trump_world), (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (u)-[*..3]-(dt) )
WHERE dt.verified = True AND u.rank_powerful <=20
RETURN path
```

**Find all companies that are top 100 richest and shortest path to Trump:**

```
MATCH (u:Organization), (dt:User {name:"Donald J. Trump"})
MATCH path = allShortestPaths( (u)-[*..]- (dt) )
WHERE dt.verified = True AND u.rank <=100
RETURN path
```