

**ANO**  
**2024**



# **UNINTER**

**CADERNO DE RESPOSTAS DA  
ATIVIDADE PRÁTICA DE:**

**NoSQL**

**ALUNO: CARLOS HENRIQUE MONNERAT  
QUINTANILHA**

**RU: 4328237**

**Caderno de Resposta Elaborado por:  
Prof. MSc. Guilherme Ditzel Patriota**

## Prática 01 – JSON COM NEO4J.

### Questão 01– DESCOBERTA DA HASHTAG PRINCIPAL.

ENUNCIADO: Veja o Roteiro da Atividade Prática para mais detalhes.

I. Apresentação dos comandos (códigos e queries) usados (não esquecer do identificador pessoal):

```
1 call apoc.load.directory('*.json') yield value
2 with value as arquivo
3 call apoc.load.json(arquivo) yield value
4 unwind value.data as tweet
5 unwind tweet.entities.hashtags as hashtag
6 with tweet, hashtag
7 merge (t:Tweet{tweet_id:tweet.id})
8 merge (h:Hashtah{tag:apoc.text.replace(apoc.text.clean(hashtag.tag),'^[a-zA-Z0-9]', '')})
9 merge (a:Aluno {ru:'4328237'})
10 merge (t)-[:contem_hashtag]->(h)
11 with h, count(distinct t) as tweetCount
12 order by tweetCount desc
13 limit 1
14 with h
15 match p=()-[:contem_hashtag]->(h) return p limit 50
16 union
17 match p=()-[:interesse_em]->(h) return p limit 1;
```

Figura 1: (Comando para encontrar a hashtag principal, utilizando 50 arquivos como base

## II. Apresentação das Imagens/Print do resultado (não esquecer do identificador):

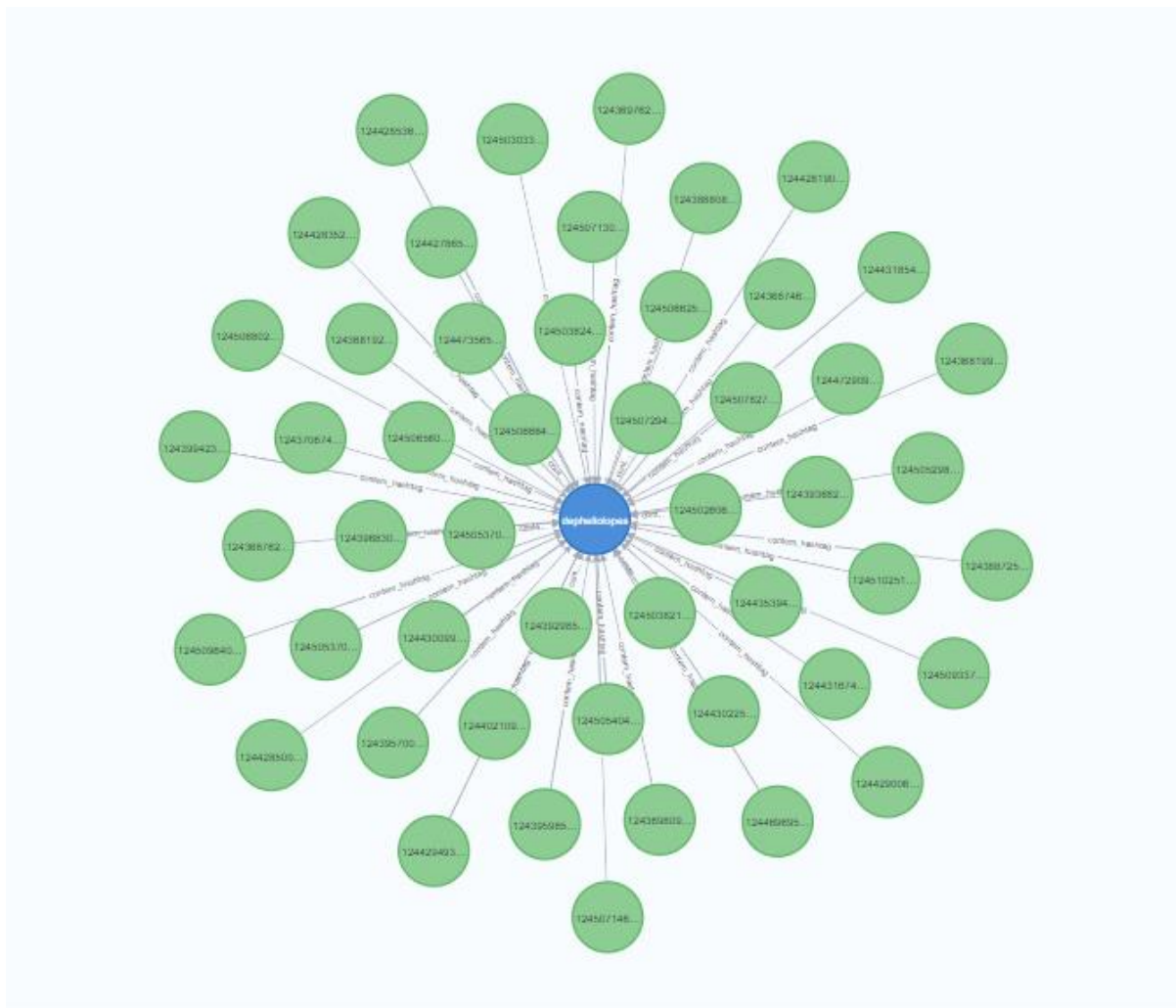


Figura 2: Grafo gerado pelo comando para encontrar a hashtag principal

## III. Resposta à pergunta: Qual foi a hashtag usada como filtro para coleta dos dados analisados?

Resposta: A hashtag foi “#depheliolopes”

## Prática 01 – JSON COM NEO4J.

### Questão 02 – ANÁLISE DOS DADOS SEGUNDO VIÉS A SUA ESCOLHA.

**ENUNCIADO:** Veja o Roteiro da Atividade Prática para mais detalhes.

- I. Apresentação dos comandos (códigos e queries) usados (não esquecer do identificador pessoal):

```
1 call apoc.load.json('tweets_coletados_177.json') yield value
2 unwind value.data as tweet
3 merge (a:Aluno {aluno_id: 4328237})
4 with tweet
5 where datetime(tweet.created_at) ≥ datetime('2006-01-01T00:00:00.000Z')
6 | and datetime(tweet.created_at) < datetime('2022-12-31T00:00:00.000Z')
7 unwind tweet.entities.mentions as mention
8 merge (t:Tweet {tweet_id: tweet.id})
9 merge (u:User {user_id: mention.id})
10 merge (t)-[:mentioned]→(u)
11 merge (a)-[:mentioned]→(u)
12 with u, count(*) as mentions
13 where mentions ≥ 10
14 return u.user_id as user_id, mentions
15 order by mentions desc
16 limit 1
```

```
1 match (u:User)
2 where size((:Tweet)-[:mentioned]→(u)) ≥ 10
3 with u
4 limit 1
5 match(a:Aluno)
6 match (t:Tweet)-[:mentioned]→(u)
7 return t, u, a
8 limit 10
```

Figura 1: Comandos para identificar qual usuário recebeu mais menções

## II. Apresentação das Imagens/Print do resultado (não esquecer do identificador):

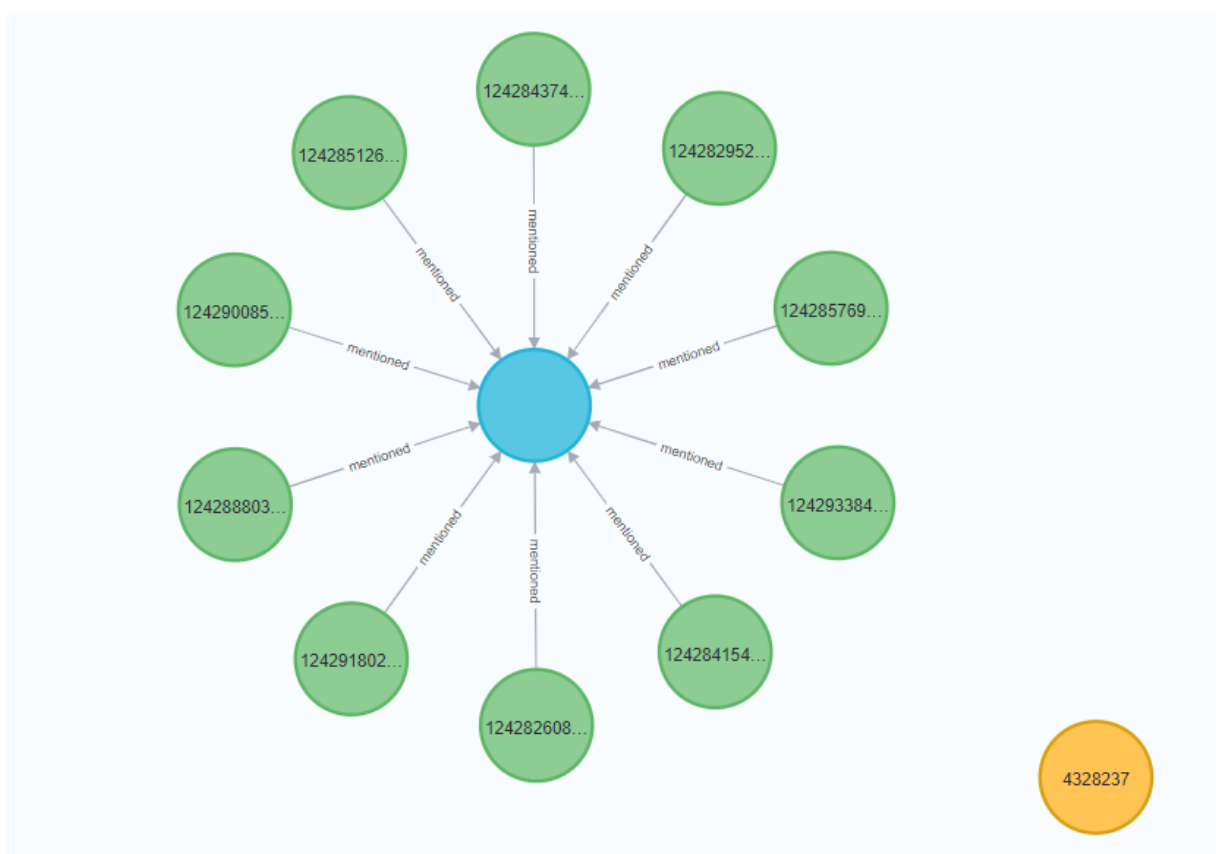


Figura 2: Grafo que mostra qual usuario recebeu mais menções

## III. Responda à pergunta: Qual foi o comando usado por você para dar entrada dos dados em JSON no seu banco de dados Neo4j?

Resposta: `CALL apoc.load.json('tweets_coletados_177.json') YIELD value`