**İLERİ SEVİYE NEO4J CYPHER SORGULARI İLE USE CASELERİ**

Bu ödevimde daha önce kullanmış olduğum 2nci ödev örnek verileriyle ileri seviye cypher sorguları kullanarak bunların use caselerini de incelemeye çalışacağım.

1. **En Fazla Davası Bulunan Şüpheliler**

MATCH (p:Person {occupation: "Şüpheli"})-[:INVOLVED\_IN]->(c:Case)

RETURN p.name AS Şüpheli, COUNT(c) AS DavaSayısı,

COLLECT(c.caseId) AS DavaListesi

ORDER BY DavaSayısı DESC

LIMIT 5

1. **Mesleklere Göre Dava Dağılımları**

MATCH (p:Person)-[r]->(c:Case)

WHERE TYPE(r) IN ["REPRESENTS", "PRESIDES\_OVER", "PROSECUTES", "INVOLVED\_IN", "WITNESSED"]

RETURN p.occupation AS Meslek, COUNT(DISTINCT c) AS DavaSayısı,

COUNT(DISTINCT p) AS KişiSayısı,

AVG(p.age) AS OrtalamaYaş

ORDER BY DavaSayısı DESC

1. **Aynı Davada Hem Tanık Hem Mağdur Olan Kişiler**

MATCH (p:Person)-[:WITNESSED]->(c:Case)<-[:INVOLVED\_IN {role: "Mağdur"}]-(p)

RETURN p.name AS İsim, c.caseId AS DavaNo, c.title AS DavaBaşlığı

1. **İkiden Fazla Davada Avukatlık Yapan Kişiler**

MATCH (p:Person)-[r:REPRESENTS]->(c:Case)

WITH p, COUNT(DISTINCT c) AS dava\_sayisi

WHERE dava\_sayisi > 2

RETURN p.name AS Avukat, dava\_sayisi AS DavaSayısı,

[(p)-[r:REPRESENTS]->(c:Case) | c.caseId] AS DavaListesi

ORDER BY dava\_sayisi DESC

1. **Dava Sürelerinin Ay Olarak Hesaplanması**

MATCH (c:Case)

WHERE c.status = 'Sonuçlandı'

WITH c, date(c.date) AS baslangic, date(datetime({epochmillis: timestamp()})) AS bugun

MATCH (c)-[:HEARD\_IN]->(court:Court)

RETURN c.caseId AS DavaNo,

duration.between(baslangic, bugun).months AS SureAy,

court.name AS Mahkeme

ORDER BY SureAy DESC

LIMIT 10

1. **Aylık Olarak Dava İstatistikleri**

WITH range(1,12) AS aylar

UNWIND aylar AS ay

MATCH (c:Case)

WHERE date(c.date).month = ay AND date(c.date).year = 2023

OPTIONAL MATCH (p)-[:INVOLVED\_IN]->(c)

WITH ay, c, COUNT(p) AS tarafSayisi

RETURN ay AS Ay,

COUNT(c) AS ToplamDava,

SUM(CASE WHEN c.status = 'Sonuçlandı' THEN 1 ELSE 0 END) AS SonuclananDava,

AVG(tarafSayisi) AS OrtalamaTaraflar

ORDER BY ay

1. **Belirli Bir Davada Bulunan Tüm Tarafları Görüntüleme**

MATCH (p:Person)-[r]->(c:Case {caseId: "2023/123"})

RETURN p.name AS İsim, p.occupation AS Meslek, type(r) AS İlişkiTürü, r.role AS Rol, r.since AS BaşlangıçTarihi

1. **Aynı Davada Görev Yapan Avukat ve Savcıları Eşleştirme**

MATCH (avukat:Person {occupation: "Avukat"})-[:REPRESENTS]->(c:Case)<-[:REPRESENTS]-(savcı:Person {occupation: "Savcı"})

RETURN DISTINCT c.caseId AS DavaNo, avukat.name AS Avukat, savcı.name AS Savcı

1. **Yıllara Göre Dava Sayısı**

MATCH (c:Case)

WITH date(c.date).year AS Yıl, COUNT(c) AS DavaSayısı

RETURN Yıl, DavaSayısı

ORDER BY Yıl DESC

Vermiş olduğum bu örnek sorgularla ilişkiler analizi, istatistik analiz sorguları, dosyalar arasında örüntüler sağlamaya çalıştım. Bu tip sorgular ile, örnek veritabanındaki karmaşık ilişkileri istenilen formatta karşımıza getirmekte ve sonrasında yapacağım chatbot entegrasyonu sayesinde kullanıcıların bu verilere doğal bir yoldan ulaşmasını hedeflemekteyim.