**KONU: Cypher**

**İÇİNDEKİLER:**

Cypher Nedir?

SQL & Chyper Benzerlik/Farkları?

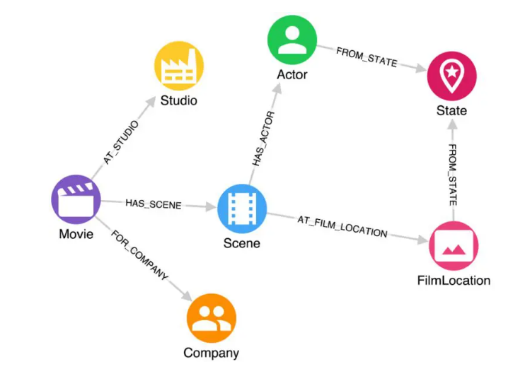
Chyper Temel Söz Dizimi?

Kaynakça

**Hazırlayan: Beyza Nur Elitok**

****

**Chyper Nedir?**

Neo4j’in kendine ait sorgulama dilidir. **Figure 2.1** SQL diline benzer ve bildirimsel ifadelerle erişimi kolaylaştırır. Neo4j veri modeli için tasarlanmıştır. Desen oluşturmak için ASCII art sözdizimini kullanır. Cypher sorgu dili, düğüm ve ilişki desenlerini tasvir eder ve bu desenleri etiketlere ve özelliklere göre filtreler. **Figure 2.0** ASCII art, Cypher’ı görsel ve okunması kolay hale getirir**.**

**Figure 2.1** X ve Y, X’ten Y’ye ilişki türü

“LotusAI” olan düğümlerdir.

(X)-[:LotusAI]->(Y)

**Figure 2.0**

**SQL & Chyper Benzerlik/Farkları?**

SQL ve Cypher, her ikisi de bildirimsel, metin tabanlı sorgu dilleri olup, anahtar sözcükler, ifadeler ve fonksiyonlar içermeleri bakımından benzerdir. Her iki dil de verileri sorgulama, filtreleme ve sonuçları döndürme amacıyla kullanılır. SQL, ilişkisel veritabanları üzerinde çalışırken, tablolar ve sütunlar arasındaki ilişkilere odaklanır. Cypher ise grafik veritabanlarıyla, düğümler ve ilişkiler arasındaki bağlantılara odaklanır. Her iki dil de veriler arasında ilişkiler kurar, ancak SQL’de bu ilişkiler tablo birleştirmeleriyle (JOIN), Cypher’da ise grafik desenleri eşleştirerek yapılır.

Cypher, grafik desenlerini eşleştirmeye odaklanır ve şemasız bir veritabanı yapısına sahip olan Neo4j'de kullanılır. Cypher'da var olmayan bir desen için sorgu hata vermez, ancak eşleşme bulamazsa herhangi bir sonuç döndürmez. SQL'de ise tablo tabanlı yapılar kullanılır ve genellikle tablolar SOL/SAĞ birleşimler ile birleştirilir. Cypher’da ilişkiler yönlüdür ve sorgular bu yönler doğrultusunda desenleri eşleştirir. Bu, Cypher’ın grafik yapısının doğasına daha uygun hale gelmesini sağlar ve özellikle karmaşık veri ilişkilerinde SQL'e kıyasla daha esnek ve sade bir sorgulama imkanı sunar.

**Chyper Temel Söz Dizimi?**

* Nodes parantezler içinde yer alır ve düğüm Etiketi ile takma adını(alias) içerir.

*“(node\_alias:NodeLabel)”*

* Relationships köşeli parantezler içinde yer alır ve ilişki Tipi ile takma adını içerir.

*“[relationship\_alias:RELATIONSHIP\_TYPE]”*

* Nodes ve Relationships, yönü belirtmek için tireler ve büyük/küçüktür sembolü ile birbirine bağlanır.

*“(person1:Person)-[rel1:HEARD\_OF]->(person2:Person)”*

* Bilgiler, Key-Value çiftleri biçiminde Nodes ve Relationships olarak kaydedilebilir. Nodes/Relationships oluşturulurken süslü parantezler ve iki nokta üst üste kullanılır.

*‘CREATE (person1:Person {property1: “a string”, property2: 123});’*

Özellikler Düğüm veya İlişki oluşturulduktan sonra nokta gösterimi kullanılarak SET

komutu ile de atanabilir:

*‘MATCH (person1:Person) SET person1.property1 = "a different string";’*

* CREAT : İki düğüm ve bunlar arasındaki yönlendirilmiş ilişkiliyi oluşturur.
  + MATCH: Belirli bir desen veya veriyi bulmak için kullanılır. SQL deki FROM komutuna benzer.
* WHERE: Sorgularda filtreleme yapmak için kullanılır. MATCH, OPTIONAL MATCH veya WITH ifadeleriyle birlikte kullanılarak belirli kriterlere uyan düğümleri veya ilişkileri seçer.
* RETURN: Sorgunun sonucunda hangi verilerin döndürüleceğini belirtmek için kullanılır.
* WITH: Sorguları aşamalara bölmek ve ara sonuçları taşımak için kullanılır.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | SQL Sorgu Dili | Cypher Sorgu Dili |  |
|  | |  |  |  |
|  | | INSERT | CREAT |  |
|  | | FROM | MATCH |  |
|  | | WHERE | WHERE |  |
|  | | SELECT | RETURN |  |
|  | CTEs (Common Table Expressions) | | WITH |  |
|  | |  |  |  |

**Figure 2.2** Cypher da yer alan methodların SQL deki karşılıkları yer almaktadır.

* Cypher üzerinde sorgu yazılırken yorumlayıcı, Nodes “()” ve Relationships “[]” gibi doğru desenleri takip etmemiz durumunda sorgunun nasıl yazıldığıyla ilgilenmez.
* Sorgu okunmasını kolaylaştırmak için topluluk tarafından Node Labels  “**P**ascal**C**ase” ve Relationship Type “MACRO\_CASE” önerilir.
* **Figure 2.3** Düğümler, ilişkiler, özellikler ve özellik değerlerinin hepsi büyük/küçük harfe duyarlıdır, yalnızca anahtar sözcük komutları duyarlı değildir.

(person\_alias:Person) ≠ (person\_alias:person)

()-[:HEARD\_OF]->() ≠ ()-[:heard\_of]->()

**Figure 2.3**

**Kaynakça:**

**ChatGPT**

<https://www.geeksforgeeks.org/neo4j-query-cypher-language/>

<https://www.graphable.ai/blog/neo4j-cypher-tutorial/>

<https://memgraph.com/blog/cypher-cheat-sheet>

<https://en.wikipedia.org/wiki/Cypher_(query_language)>

[Comparing Cypher with SQL - Getting Started (neo4j.com)](https://neo4j.com/docs/getting-started/cypher-intro/cypher-sql/)