Çisem Helvacı 19.05.2025

Hafta 4 - Ödev

**Ödev 3**

CYPHER KODLARI (LOAD CSV)

**1. Constraints**

CREATE CONSTRAINT author\_name IF NOT EXISTS FOR (a:Author) REQUIRE a.name IS UNIQUE;

CREATE CONSTRAINT book\_title IF NOT EXISTS FOR (b:Book) REQUIRE b.title IS UNIQUE;

CREATE CONSTRAINT genre\_name IF NOT EXISTS FOR (g:Genre) REQUIRE g.name IS UNIQUE;

CREATE CONSTRAINT country\_name IF NOT EXISTS FOR (c:Country) REQUIRE c.name IS UNIQUE;

**2. Nodes**

**authors.csv**

LOAD CSV WITH HEADERS FROM 'file:///authors.csv' AS row

MERGE (a:Author {name: row.name})

SET a.birthYear = toInteger(row.birthYear),

a.nationality = row.nationality;

**books.csv**

LOAD CSV WITH HEADERS FROM 'file:///books.csv' AS row

MERGE (b:Book {title: row.title})

SET b.publishYear = toInteger(row.publishYear),

b.genre = row.genre;

**genre.csv**

LOAD CSV WITH HEADERS FROM 'file:///genres.csv' AS row

MERGE (g:Genre {name: row.name});

**country.csv**

LOAD CSV WITH HEADERS FROM 'file:///countries.csv' AS row

MERGE (c:Country {name: row.name});

**3. Relationshis**

**WROTE (Yazar → Kitap)**

LOAD CSV WITH HEADERS FROM 'file:///wrote.csv' AS row

MATCH (a:Author {name: row.author})

MATCH (b:Book {title: row.book})

MERGE (a)-[:WROTE {year: toInteger(row.year)}]->(b);

**BELONGS\_TO (Kitap → Tür)**

LOAD CSV WITH HEADERS FROM 'file:///belongs\_to.csv' AS row

MATCH (b:Book {title: row.book})

MATCH (g:Genre {name: row.genre})

MERGE (b)-[:BELONGS\_TO {assignedDate: date(row.assignedDate)}]->(g);

**FROM (Yazar → Ülke)**

LOAD CSV WITH HEADERS FROM 'file:///from.csv' AS row

MATCH (a:Author {name: row.author})

MATCH (c:Country {name: row.country})

MERGE (a)-[:FROM {since: toInteger(row.since)}]->(c);

**4. Exploratory Sorgular (Veri Keşfi)**

**1. En üretken yazarlar**

MATCH (a:Author)-[:WROTE]->(b:Book)

RETURN a.name AS Author, COUNT(b) AS NumberOfBooks

ORDER BY NumberOfBooks DESC;

**2. Yıllara göre kitap sayısı**

MATCH (b:Book)

RETURN b.publishYear AS Year, COUNT(\*) AS BookCount

ORDER BY Year;

**3. Türlere göre kitap sayısı**

MATCH (b:Book)-[:BELONGS\_TO]->(g:Genre)

RETURN g.name AS Genre, COUNT(b) AS BookCount

ORDER BY BookCount DESC;

**4. Ülkelere göre yazar sayısı**

MATCH (a:Author)-[:FROM]->(c:Country)

RETURN c.name AS Country, COUNT(a) AS NumberOfAuthors

ORDER BY NumberOfAuthors DESC;

**5. Aynı türde kitap yazmış yazar çiftleri**

MATCH (a1:Author)-[:WROTE]->(:Book)-[:BELONGS\_TO]->(g:Genre)<-[:BELONGS\_TO]-(:Book)<-[:WROTE]-(a2:Author)

WHERE a1.name < a2.name

RETURN DISTINCT a1.name AS Author1, a2.name AS Author2, g.name AS SharedGenre;

**6. Yazarların kitapları ve ülkesi**

MATCH (a:Author)-[:WROTE]->(b:Book), (a)-[:FROM]->(c:Country)

RETURN a.name AS Author, c.name AS Country, collect(b.title) AS Books

ORDER BY Country;

**7. Her tür için yayın yıllarına göre dağılım**

MATCH (b:Book)-[:BELONGS\_TO]->(g:Genre)

RETURN g.name AS Genre, b.publishYear AS Year, COUNT(\*) AS Count

ORDER BY Genre, Year;

**8. Yalnızca bir kitap yazmış yazarlar**

MATCH (a:Author)-[:WROTE]->(b:Book)

WITH a, COUNT(b) AS bookCount

WHERE bookCount = 1

RETURN a.name AS Author;

**9. Aynı ülke ve aynı türde kitap yazan yazar eşleşmeleri**

MATCH (a1:Author)-[:FROM]->(c:Country)<-[:FROM]-(a2:Author),

(a1)-[:WROTE]->(:Book)-[:BELONGS\_TO]->(g:Genre)<-[:BELONGS\_TO]-(:Book)<-[:WROTE]-(a2)

WHERE a1.name < a2.name

RETURN DISTINCT a1.name AS Author1, a2.name AS Author2, c.name AS Country, g.name AS SharedGenre;

**10. Farklı türlerde kitap yazmış yazarlar**

MATCH (a:Author)-[:WROTE]->(b:Book)-[:BELONGS\_TO]->(g:Genre)

WITH a.name AS Author, COLLECT(DISTINCT g.name) AS Genres

WHERE SIZE(Genres) > 1

RETURN Author, Genres;