

Dokumentacja Grafowe Bazy Danych

Adam Jędrychowski

16 grudnia 2023

1 Projekt koncepcji, założenia

Projekt prezentuje wykorzystanie grafowej bazy danych przy tworzeniu aplikacji do przechowywania informacji o filmach. Jego celem jest demonstracja umiejętności korzystania z grafowej struktury bazy danych oraz języka Cypher do komunikacji z tą bazą danych.

2 Rozwiązania chmurowe

Aplikacja została umieszczona w Azurowej chmurze i jest dostępna pod adresem GraphDB.

3 Funkcjonalność

Aplikacja umożliwia

- dodawanie i usuwanie filmów
- wylistowanie ekipy filmowej danego filmu
- dodawanie i usuwanie danego pracownika z bazy danych
- aktualizacja danych pracowników oraz filmów
- dodanie nowej oceny i komentarza do filmu
- wylistowanie ocen danego filmu
- przypisanie pracownika do danego filmu
- wylistowanie filmów, w których dany aktor brał udział
- filtrowanie bazy danych

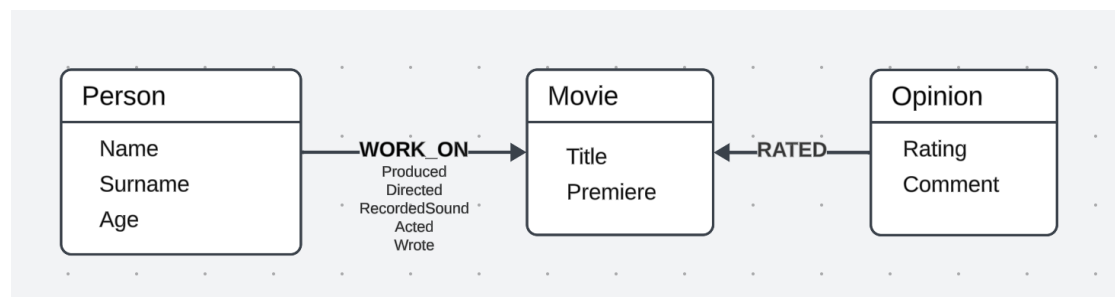
Filtracji można dokonać według:

- roli pracownika w filmie
- wieku pracowników
- imionach pracowników
- nazwiskach pracowników
- tytule filmu
- roku premiery filmu

Wyniki, które zostały odfiltrowane, można uporządkować według różnych kategorii, takich jak imiona, nazwiska, tytuły filmów oraz oceny. Dodatkowo, istnieje możliwość określenia liczby wyników, które zostaną zwrócone z bazy danych.

4 Schemat bazy danych

Głównym węzłem w bazie danych jest węzeł opisujący film. Do tego węzła możemy przypisać kolejne opinie oraz informacje o osobie z ekipy filmowej. Węzły te łączymy za pomocą jednej z relacji: aktor, producent, reżyser, scenarzysta lub dźwiękowiec.



Rysunek 1: Struktura grafowej bazy danych

5 Zapytania w języku Cypher

Utworzenie węzła pracownik i wypełnienie odpowiednimi wartościami. Funkcja zwraca unikalne id węzła.

```
CREATE (e:Employee {name:$name, surname:$surname, age:$age})
RETURN elementId(e)
```

Utworzenie węzła film i wypełnienie odpowiednimi wartościami. Funkcja zwraca unikalne id węzła.

```
CREATE (m:Movie {title:$title, premiere:$premiere})
RETURN elementId(m)
```

Wyszukanie i zwrócenie wszystkich ocen filmu o określonym id.

```
MATCH (m:Movie)<-[r:RATED]-(o:Opinion) WHERE elementId(m)=$id
RETURN o
```

Utworzenie nowej opinii określonego filmu.

```
MATCH (m:Movie) WHERE elementId(m)=$movieID CREATE
(m)<-[r:RATED]-(o:Opinion {rating:$rating, comment:$comment})
```

Wyszukanie i zwrócenie wszystkich węzłów pracownik.

```
MATCH (e:Employee) RETURN e
```

Wyszukanie wszystkich filmów oraz opcjonalnie opinii, jeśli dany film posiada je. Zwracamy filmy oraz odpowiadającą mu średnią wartość ocen.

```
MATCH (m:Movie) OPTIONAL MATCH (m)<-[r:RATED]-(o:Opinion)
RETURN m, avg(o.rating)
```

Połącz relacją określony film z danym pracownikiem.

```
MATCH (e:Employee), (m:Movie) WHERE elementId(e)=$employeeID
AND elementId(m)=$movieID CREATE
(e)-[w:WORKON {role:$role}]->(m)
```

Wyszukaj wszystkich pracowników ekipy danego filmu.

```
MATCH (e:Employee)-[w:WORKON]->(m:Movie)
WHERE elementId(m)=$movieID RETURN w.role, e
```

Wyszukaj wszystkie filmy, nad którymi dana osoba pracowała.

```
MATCH (e:Employee)-[w:WORKON]->(m:Movie)
WHERE elementId(e)=$employeeID RETURN w.role, m
```

Odfiltrowanie bazy danych

```
MATCH (e:Employee)-[w:WORKON]->(m:Movie) WHERE w.role=$role
AND e.age>$lowAge AND e.age<=$highAge AND e.name=$name AND
e.surname=$surname AND m.title=$title AND
m.premiere.year=$premiere OPTIONAL MATCH
(m)<-[r:RATED]-(o:Opinion) RETURN e, w.role, m, avg(o.rating)
AS rating ORDER BY e.name LIMIT 45
```

Usunięcie pracownika z bazy, o konkretnym ID.

```
MATCH (e:Employee) WHERE elementId(e)=$employeeID
DETACH DELETE e
```

Wyszukanie pracownika w bazie, o konkretnym ID.

```
MATCH (e:Employee) WHERE elementId(e)=$employeeID RETURN e
```

Aktualizacja danych określonego pracownika.

```
MATCH (e:Employee) WHERE elementId(e)=$employeeID
SET e.name=$name, e.surname=$surname, e.age=$age
```

Usuń konkretny film wraz z relacjami podłączonymi do niego.

```
MATCH (m:Movie) WHERE elementId(m)=$movieID DETACH DELETE m
```

Wyszukaj film o konkretnym id.

```
MATCH (m:Movie) WHERE elementId(m)=$movieID RETURN m
```

Aktualizacja danych określonego filmu.

```
MATCH (m:Movie) WHERE elementId(m)=$movieID  
SET m.title=$title , m.premiere=$premiere
```