

Övningar SQL jämfört med MongoDB

I dessa övningsuppgifter får du möjlighet att öva vidare på att skapa relationsdatabaser med SQL (MariaDB/MySQL) och dokumentdatabaser i MongoDB. Du får öva på att fatta beslut om hur man bäst strukturerar de olika typerna av databaser, utifrån olika aspekter. Du får också öva på att arbeta mot en SQL-databas respektive en databas i MongoDB genom att ställa frågor (queries) som skapar, läser, uppdaterar och tar bort data ur respektive databas.

1. Hur lagras data i relationsdatabaser? Beskriv grundstrukturen.

2. Hur lagras data i MongoDB? Beskriv grundstrukturen.

3. Skapa en förenklad databas i MariaDB (SQL). Databasen ska lagra data över filmer, skådespelare och filmgenrer. En skådespelare kan vara med i flera filmer, en film kan ha flera genrer.

Grundattribut för skådespelare: Id, firstName, lastName, birthyear, country

Grundattribut för filmer: Id, title, productionYear

Grundattribut för genre: Id, namn

- Bestäm vilka tabeller du behöver skapa, vad de ska innehålla (förutom de ovan listade grundattributen) och vilka kopplingar du behöver skapa mellan de olika tabellerna för att följa specifikationen i uppgiften.

Skriv SQL som ...

4.

- a) ... Lägger till ett antal skådespelare.
- b) ... Lägger till ett antal filmer.
- c) ... Lägger till ett antal genrer.
- d) ... Kopplar ihop skådespelare och filmer.
- e) ... Kopplar ihop filmer och genrer.

5. ... listar alla skådespelare som är födda ett visst år och kommer från ett visst land.

6. ... listar alla skådespelare som kommer från antingen ett land eller ett annat land.

7. ...listar alla filmer som en viss skådespelare spelar i.

8. ... listar alla filmer som en viss skådespelare spelar i och som är producerade ett visst år.

9.

a)... listar alla filmer som producerades mellan två årtal. Ordna listan i stigande ordning.

b) Skriv om samma query så att listan sorteras i fallande ordning istället.

10. ... uppdaterar en viss skådespelares efternamn.

11. ... tar bort kopplingen mellan en specifik film och en specifik genre (en koppling du lade till i övning 4e).

12. ... tar bort en skådespelare ur databasen.

- Kan man ta bort en skådespelare om det fortfarande finns data över filmer där skådespelaren är med i databasen? Varför eller varför inte? Hur löser du uppgiften?

13. ... tar bort en genre ur databasen.

- Kan man ta bort en genre ur databasen om det finns filmer som hör ihop med genren? Varför eller varför inte? Hur löser du uppgiften?

14. ... uppdaterar så att en existerande film får ytterligare en genre och en skådespelare kopplade till sig.

15. ...listar alla skådespelare tillsammans med alla filmer de är med i.
16. ... tar fram data över en specifik film plus data över alla skådespelare som spelar i den.
17. ... listar alla filmer som har en specifik genre kopplad till sig.
18. Skapa nu en databas över filmer, skådespelare och genrer i MongoDB.
- a) Bestäm själv vilka Collections som du anser är bäst att skapa för att lagra den beskrivna datan. Man brukar prata om två typer av övergripande datamodellering i MongoDB: **embedded data model** och **normalized data model**, använd någon av de här som utgångspunkt för hur du väljer att skapa din databas.
- * Text som beskriver skillnaden mellan embedded och normalized data model:*
- <https://docs.mongodb.com/manual/core/data-model-design/>
- b) Vilken datamodell väljer du att använda dig av för att lagra datan (embedded data model eller normalized data model), varför passar den bra för just det beskrivna use case (med skådespelare, filmer och genrer)?
- c) Om du valde "normalized data model" använde du dig då av "manual references" eller "dbreferences"? Eller använde du kanske båda två? Motivera varför du valde ett visst upplägg.

Skriv queries för att skapa och påverka/använda data i din MongoDB-databas:

Om du inte redan har gjort det: Skapa en eller flera collections (det är ditt beslut att avgöra hur du bäst ska strukturera databasen) där du lägger in dokument med data över skådespelare, filmer och genrer.

Tänk på att du kan behöva skriva flera separata queries för att kunna lösa vissa av uppgifterna nedan.

Skriv document-queries som...

19. ... listar alla skådespelare som kommer från ett visst land.

20. ... listar alla skådespelare som är födda mellan två årtal.

21. ... listar alla filmer som är producerade ett visst årtal.

22. ... listar alla filmer som är producerade mellan två årtal.

23. ... tar bort en specifik film och dem skådespelare som är kopplade till filmen.

24. ... lägger till en till skådespelare i en collection (den som passar bäst utifrån vilka collections du har valt att skapa).

25. ... kopplar ihop en film med en genre genom en manuell referens eller en DBref.

(se t.ex. Denna sida för info om vad detta innebär:

<https://docs.mongodb.com/manual/reference/database-references/>).

26. ... tar bort en specifik film som du tidigare har lagt till (en film med ett specifikt namn som nu också har ett unikt id i databasen).

27. ... tar bort alla filmer som hör ihop med/är av en viss genre.

27 ... uppdaterar en i databasen existerande film så att den får ytterligare en genre kopplad till sig.