

ER-diagram övningar med lösningsförslag

Övning 1:

A)

Ett bibliotekssystem där det finns böcker, författare och biblioteksmedlemmar. Varje bok har en titel, ISBN och kan ha flera författare. Varje författare har ett namn och en unik identifierare. Varje biblioteksmedlem har ett namn och en unik identifierare. Böcker kan lånas av biblioteksmedlemmar.

B) extra

Skapa DB med tabeller och dess relationer. Fyll alla tabeller med passande information. Gör en SQL-fråga som joinar alla tabeller och hämtar relevant info. Ex hämta alla böcker från författare 1, tillsammans med medlemmarna som för tillfället lånar dessa böcker.

Lösning A:

Entiteter:

- Bok (Titel, ISBN)
- Författare (FörfattarID, Namn)
- Biblioteksmedlem (MedlemsID, Namn)

Relationer:

- Relationstabell mellan Författare och Bok (ID, FörfattarID, ISBN)
- Relationstabell mellan medlem och Bok (ID, MedlemID, ISBN)

Lösning B:

```
SELECT * FROM authors
JOIN authors_books ON authors.id = authors_books.author_id
JOIN books ON authors_books.isbn = books.isbn
JOIN books_members ON books.isbn = books_members.isbn
JOIN library_members ON books_members.library_member_id = library_members.id
WHERE authors.id = 1
```

Övning 2:

A)

En databas för ett universitetssystem ska designas. Universitetet har flera institutioner, var och en med en unik kod och ett namn. Professorer arbetar för en institution och har en unik identifierare, namn och lön. Kurser erbjuds av institutioner och har en kod, namn och antal poäng. En professor kan undervisa i flera kurser. En kurs kan enbart undervisas av en professor

B) extra

Skapa DB med tabeller och dess relationer. Fyll alla tabeller med passande information. Gör en SQL-fråga som joinar alla tabeller och hämtar relevant info. Ex hämta alla kurser som hålls av professor 1, tillsammans med vilken institution som håller kursen.

Lösning A:

Entiteter:

- Institution (Institutionskod, Namn)
- Professor (ProfID, Namn, Lön)
- Kurs (Kurskod, Namn, Poäng)

Relationer:

- Professor Arbetar för Institution (ProfID, Institutionskod)
- Kurs Erbjuds av Institution (Kurskod, Institutionskod)
- Relationstabell mellan kurser och professorer

Lösning B:

// Visa alla kurser med tillhörande professorer, i institution 2

```
SELECT institutions.name, courses.name, professors.name FROM courses
JOIN institutions ON courses.institution_id = institutions.id
JOIN courses_professors ON courses.id = courses_professors.course_id
JOIN professors ON courses_professors.professor_id = professors.id
WHERE institutions.id = 2
```

// Visa alla professorer, i institution 2

```
SELECT COUNT(*) AS "Number of professors in Inst 2" FROM institutions
JOIN professors ON institutions.id = professors.institution_id
WHERE institutions.id = 2
```

Övning 3:

Designa ett ER-diagram för ett sjukhus ledningssystem. Sjukhuset har flera läkare, sjuksköterskor och patienter. Varje läkare och sjuksköterska har en unik identifierare, namn och specialitet. Varje patient har en unik identifierare, namn och rumsnummer. Läkare behandlar patienter och sjuksköterskor tar hand om patienter.

Lösning:

Entiteter:

- Läkare (LäkarID, Namn, Specialitet)
- Sjuksköterska (SjuksköterskeID, Namn, Specialitet)
- Patient (PatientID, Namn, Rumnummer)

Relationer:

- Relationstabell mellan Läkare och Patient
- Relationstabell mellan Sjuksköterska och Patient

Övning 4:

Konstruera ett ER-diagram för ett bilförsäkringsbolag vars kunder äger en eller flera bilar var. Varje bil kan associeras med valfritt antal registrerade olyckor.

Lösning:

Entiteter

- Kund (KundID, Namn, Adress, Telefonnummer)
- Bil (Regnummer, Märke, Modell)
- Olycka (OlycksID, Datum, Plats, SkadaBeskrivning)

Relationer:

- Ägarskap (KundID, Regnummer)
- Skada (OlycksID, Regnummer, KundID)

I denna design:

- Varje kund kan ha flera bilar.
- Varje bil kan ha noll eller flera skador kopplade till den.
- Varje skada registreras mot en specifik bil.

Genom att använda denna ER-modell kan försäkringsbolaget effektivt spåra kunder, deras fordon och eventuella skador som är relaterade till varje fordon.