

## Jump for fun - Obl. DB opgave

**Fitnesskæden "Jump for fun"** har specialiseret sig i en særlig træningsform, som tager udgangspunkt i, at man kan hoppe sig i form.

Kæden har indtil videre etableret sig i flere sjællandske byer og har planer om at ekspandere til resten af landet.

Fitnesskæden har brug for et nyt databasesystem, og her kommer du ind i billedet – du skal nemlig designe systemet med tilhørende implementering.

Når et nyt medlem skal oprettes, sker det ved at registrere mobilnummer, e-mailadresse, for- og efternavn, fødselsdato, adresse (gade/vej, nr., postnummer og by) samt datoen for oprettelse. Disse felter må ikke være tomme.

Et medlem registreres med et unikt fortløbende medlemsnummer på tværs af alle centre, som starter fra nummer 100000 og fremefter. E-mail og mobilnummer skal være unikke – det vil sige, at man ikke må kunne oprette flere medlemmer med samme mobilnummer eller e-mailadresse. Man skal være over 18 år for at kunne blive medlem af kæden.

Som nævnt findes træningscentre på flere forskellige sjællandske lokationer.

(Kort her: Ringsted, Køge, Roskilde, Holbæk.)

Et medlem er som udgangspunkt tilknyttet et af centrene som deres foretrukne center, men medlemmerne kan sagtens booke plads på andre træningslokationer.

Medlemmer kan booke plads på forskellige træningshold. Hvert hold har et maksimalt antal pladser (25), og der kan være forskellige typer hold (f.eks. intensiv hop, hop for begyndere, familie-hop). Et hold varer som udgangspunkt 60 minutter, medmindre andet angives. Hvert hold ledes af en træner, der også er registreret i systemet med navn, telefonnummer og en unik træner-ID. Når et hold er fuldt booket, er det ikke længere muligt at tilmelde sig, medmindre en anden deltager melder fra. Holdene afholdes på specifikke tidspunkter, og hvert hold er knyttet til et center. Holdene har en start- og sluttid samt en specifik ugedag. Trænerne kan opdatere eller aflyse hold via systemet. Når et hold aflyses, fjernes alle medlemmer, som er tilknyttet holdet.

På alle lokationer er der to træningssale, som kan benyttes samtidig. Det betyder, at der om tirsdagen f.eks. i træningssal 1 kan afholdes et hold mellem 17 og 18, og det samme kan der i træningssal 2. Der må dog ikke kunne oprettes to hold i samme tidsinterval i samme træningsrum. For at undgå fejl ønsker kæden, at du sikrer, at starttidspunktet for et hold altid ligger før sluttidspunktet.

Medlemmer kan have forskellige abonnementsniveauer (f.eks. basis, premium). Basis-abonnementet giver adgang til 10 træninger pr. måned, mens premium-abonnementet giver ubegrænset adgang. Medlemmer kan kun deltage i hold inden for deres abonnementslimit. Ved opgradering af abonnement nulstilles grænsen for antal træninger for den indeværende måned.

## Opgaven

Der er følgende krav til besvarelse af denne obligatoriske opgave:

### 1. ER-diagram

Undersøg casebeskrivelsen og udarbejd et ER-diagram på baggrund af din analyse af domænet.

Du kan udarbejde dit diagram gennem draw.io. Husk attributter (inkl. de forskellige typer), relationer, entiteter og kardinalitet.

### 2. RDS-diagram

Udarbejd et normaliseret RDS-diagram gennem brug af mappingsreglerne, og indsæt også datatyper og referentiel integritet.

### 3. Implementer din database ved at udarbejde et script

Bemærk, at dette script skal afleveres. Husk at tænke på implementering af constraints og forretningslogik.

Når vi skal udarbejde DB-scripts, er der en "best practice", man følger, og det er for at sikre, at scriptet kan køre, og for overskuelighedens skyld:

- Opret eller vælg database
- Opret tabeller
- Definer primære nøgler
- Opret fremmednøgler og relationer
- Definer andre constraints (UNIQUE, CHECK, NOT NULL)
- Opret triggers
- Indsæt startdata (hvis nødvendigt)
- Opret brugere og tildel rettigheder

Du kan se **best practice** i den vedhæftede fil

Husk at afprøve og teste din løsning løbende.

### 4. Ekstra opgave (Den behøver du ikke implementere) – Venteliste

Du kan tilføje et ventelistesystem til holdene. Hvis et hold er fuldt booket, kan medlemmerne sætte sig på venteliste, og de får automatisk en plads, hvis en anden deltager melder fra.

Opgaven kan løses ved hjælp af en trigger, se evt. mere her:

<https://www.mssqltips.com/sqlservertip/5909/sql-server-trigger-example/>

Husk at opdatere ER diagrammet, hvis du vælger denne opgave



Best practice - db script

Tekstfil

◀ Forrige

Spring til...

Næste ▶

### Intra for ansatte

[Forskning & Udvikling](#)

[Globalt udsyn](#)

[GDPR for ansatte](#)

[HR & Løn](#)

[Indkøb & Bygningsdrift](#)

[Inspiration til digitale aktiviteter](#)

[IT for ansatte](#)

[Kommunikation](#)

[Kvalitet](#)

[Lektorkvalificering](#)

[Samarbejdsudvalg og Arbejdsmiljøorganisation](#)

[Studieadministration](#)

[Økonomi](#)

### Intra studerende

[De Studerendes Råd](#)

[Dispensation og Klager](#)

[Eksamen](#)

[Ekstra støtte / Extra Support \(SPS\)](#)

[GDPR for Studerende](#)

[IT for Studerende](#)

[Kom ud i verden](#)

[Moodle for Studerende](#)

[Ordensregler på Zealand](#)

[Studies after AP degree](#)

[Studievejledning](#)

[SU & Ungdomskort](#)

[Velkommen Zealanders 2024](#)

[Zealand Shop](#)

### Intra Efteruddannelse

[Zealand Efteruddannelse](#)

[EVU Onboarding for undervisere](#)

### Andre systemer

[Biblioteksdatabaser](#)

[Evaluering](#)

[Fraværsregistrering](#)

[Innomate](#)

[Office365](#)

[Panopto-Videoportal](#)

[SDBF](#)

[Parkeringsregistrering](#)

[Skema](#)

[Studiekort](#)

[Tidsregistrering](#)

[Wiseflow](#)

[Zealand UMS](#)

[Zoom](#)

[Rumoverblik](#)