

TDT4145 Datamodellering og databasesystemer

Mappingsoppgave 1 – løsningskisse

1) Karuselløp

Karusell(Årstall, Beskrivelse)

Løp(LøpNr, Dato, Sted, KarusellID)

- KarusellID er fremmednøkkel mot Karusell

LøpKlasse(Løpnr, KlasseID, Alder, Kjønn, Distanse)

- LøpNr er fremmednøkkel mot Løp

Plassering(LøpNr, KlasseID, Plass, Tid, DeltakerNr)

- LøpNr, KlasseID er fremmednøkkel mot LøpKlasse
- DeltakerNr er fremmednøkkel mot Deltaker

Deltaker(Nr, Navn, FAar, Kjønn, E-post)

Påmelding(LøpNr, KlasseID, DeltakerNr, StartNr)

- LøpNr, KlasseID er fremmednøkkel mot LøpKlasse
- DeltakerNr er fremmednøkkel mot Deltaker

Betaling(Årstall, DeltakerNr, Beløp)

- Årstall er fremmednøkkel mot Karusell
- DeltakerNr er fremmednøkkel mot Deltaker

2) Skiutøvere

a) Beholder kun superklassen

Når vi har en slik løsning må superklassen ha alle attributter som finnes i superklassen eller i en eller flere subklasser. Ut fra modellen kan en utøver være med i null til mange subklasser. Deltakelse i subklasser registreres derfor med attributtene Hopp, Kombinert, Langrenn og Skiskyting som er logiske attributter som settes lik *true* hvis utøveren driver med øvelsen.

Skiutøver(ID, Navn, FAar, Nasjon, Høyde, Vekt, MaksPuls, StåTreffProsent, LiggTreffProsent, Hopp, Kombinert, Langrenn, Skiskyting)

Fordelen med denne løsningen er at alt samles i en tabell. Ulempen er NULL-verdier for attributter som ikke er relevante for en utøver).

b) Beholder superklassen og subklasser

I følge modellen kan superklassen inneholde utøvere som ikke er med i noen subklasse. Vi må derfor ha en tabell som representerer superklassen. Vi synes ikke subklassen *LangrennsUtøver* tilfører nok nytte til å beholde den og velger å sløyfe denne i relasjonsskjemaet:

Skiutøver(ID, Navn, FAar, Nasjon)

Hopper(ID, Høyde, Vekt)

- ID er fremmednøkkel mot Skiutøver

KombinertUtøver(ID, MaksPuls)

- ID er fremmednøkkel mot Hopper

Langrenner(ID, MaksPuls)

- ID er fremmednøkkel mot Skiutøver

Skiskytter(ID, MaksPuls, StåTreffProsent, LiggTreffProsent)

- ID er fremmednøkkel mot Skiutøver

Fordelen med denne løsningen er at vi unngår nullverdier, til gjengjeld må vi forene (sette sammen med join) data fra flere tabeller for å finne alle opplysninger om en utøver. Legg merke til at data om en kombinertutøver vil være spredt over tre tabeller.

3) Eksamenssensur

Student(StudentNr, Navn, E-post)

Eksamen(EksamenNr, Dato, EmneKode)

- EmneKode er fremmednøkkel mot Emne

Emne(EmneKode, Tittel, Beskrivelse)

EksamensResultat(SensurNr, Karakter, Type, EksamenNr, StudentNr, SensorID)

- EksamenNr er fremmednøkkel mot Eksamen
- StudentNr er fremmednøkkel mot Student
- SensorID er fremmednøkkel mot Sensor

Sensor(SensorID, Type)

Ansatt(AnsattNr, Navn, E-post, SensorID)

- SensorID er fremmednøkkel mot Sensor

OppnevntSensor(OppnevntSensorID, Navn, E-post, SensorID)

- SensorID er fremmednøkkel mot Sensor

Klagenemnd(NemndID, StartDato, InternAnsattNr, OppnevntSensorID, SensorID)

- InternAnsattNr er fremmednøkkel mot Ansatt
- OppnevntSensorID er fremmednøkkel mot OppnevntSensor
- SensorID er fremmednøkkel mot Sensor

For å håndtere kategorien *Sensor* har vi laget en sensortabell med surrogatnøkkelen *SensorID* og et type-attributt som forteller om sensor er en ansatt sensor, en oppnevnt (ekstern) sensor eller en klagenemnd. *SensorID* blir også en fremmednøkkel i tabellene for superklassene *Ansatt*, *OppnevntSensor* og *Klagenemnd*, og fremmednøkkel i *EksamensResultat* for å ivareta relasjonen mellom *EksamensResultat* og *Sensor*.

Roger Midtstraum, 21.01.2012