

## Innleveringsoppgave 4

Vi skal her bygge videre på databasen fra innleveringsoppgave 3. (Modellen fra innleveringsoppgave 3 er gjengitt på siste side.)

Universitetet i Uqbar er veldig opptatt av effektivisering og vil gjerne få timelister, både for overtid og for timeansatte, på elektronisk form. En måte å gjøre dette på, er å utvide databasen fra forrige innlevering. Siden vi nå skal se på dette fra arbeidsgiverens side, trenger vi å skille mellom forskjellige ansatte. Dessuten må vi vite om den enkelte er ansatt på timebasis (og skal ha vanlig timelønn) eller fast (og bare skal ha ekstra betalt når timetallet overstiger visse grenser).

Fast ansatte kan være ansatt full tid, noe som i Ruritania er 30 timer pr. uke, eller de kan være deltidsansatt. I hvert fall for alle fast deltidsansatte må systemet vite hvor mange timer pr. uke den ansatte skal jobbe. (Overstiger timetallet dette, skal det utbetales timelønn frem til 30 timer overstiges; deretter skal det utbetales overtid.)

Ansatte har et lønnstrinn, og for hvert lønnstrinn finnes en månedslønn for fulltidsansatte, en vanlig timelønn og en timelønn for overtid. (Myntenheten i Ruritania heter ruro, og en ruro er hundre ur.)

Videre jobber de ansatte ofte, men ikke alltid, på prosjekter. Prosjekter har en prosjektkode, dessuten har de en budsjettramme for personalkostnader (som angis i antall timer, ikke i ruro og ur). De har en prosjektleder som alltid er fast ansatt, og som skal godkjenne timebruken. Når en ansatt blir satt på et prosjekt, bestemmes det også en øvre grense for hvor mange timer vedkommende skal jobbe på prosjektet.

*Oppgave 1.* Bruk ORM til å lage en informasjonsmodell som passer med virkelighetsbeskrivelsen (og forretningsreglene) over.

Hint:

1. Prøv å identifisere deler som kan modelleres hver for seg.
2. Hvis en del virker komplisert å modellere, så forsøk å modellere en forenklet versjon først. (Det er bedre å ha en enkel modell der noen aspekter kanskje ikke er helt ivaretatt, enn å enten ha en komplisert modell som er feil, eller å ikke få til noe i det hele tatt...)
3. Husk at ORM-modeller bare inneholder statiske skranker. Derfor kan det generelt være at en beskrivelse inneholder noen «dynamiske» opplysninger som ikke er relevante for ORM-modelleringen. Det er også ofte slik at noen av de «statiske» opplysningene heller ikke modelleres, enten fordi de gir en unødig komplisert modell, eller fordi de simpelthen ikke er relevante.

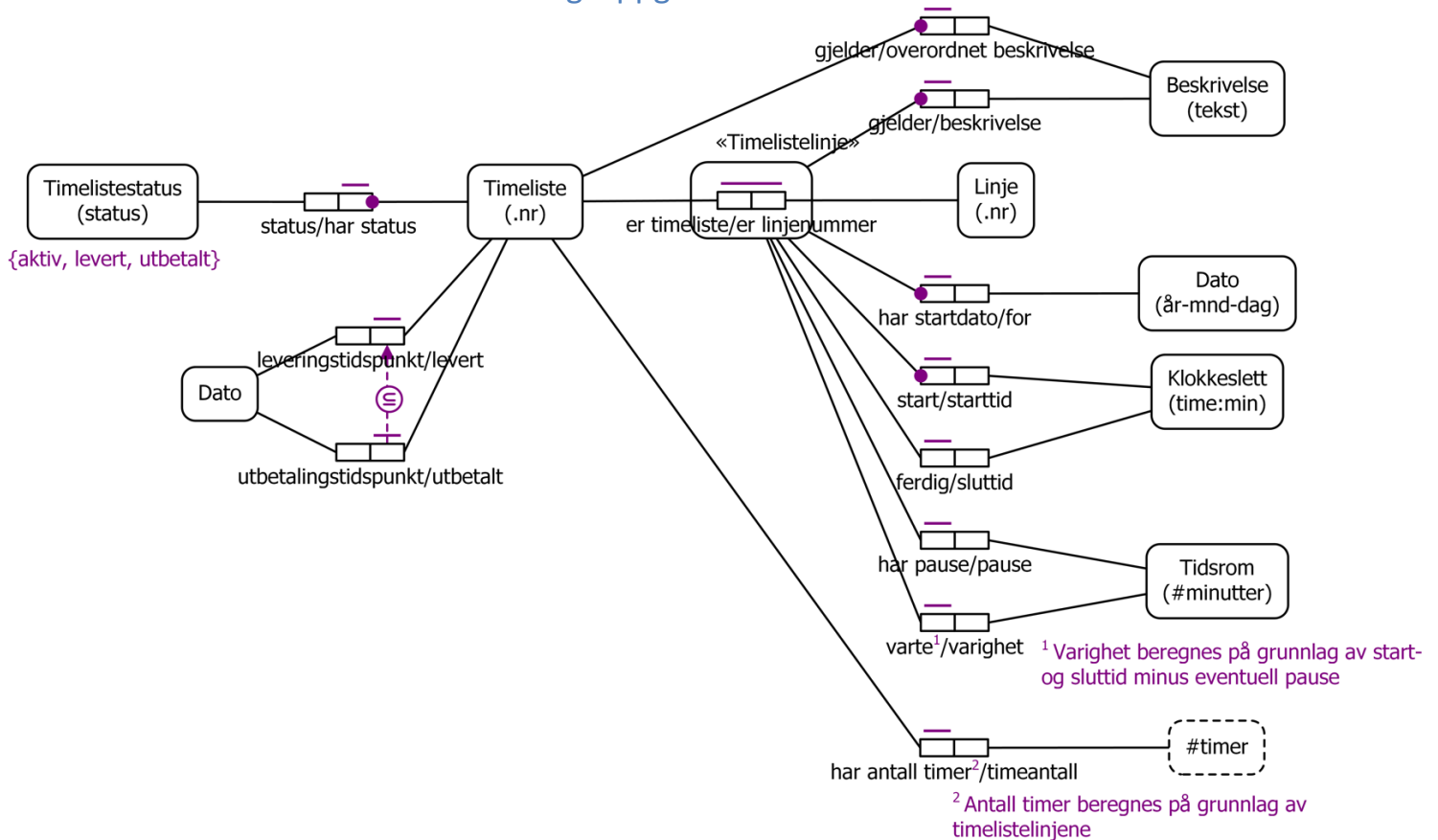
*Oppgave 2.* Realiser ORM-modellen som et relasjonsskjema.

Dere skal ikke levere SQL-kode for dette, men presentere realiseringen på samme form som relasjonsskjemaet for den opprinnelige databasen; dette finner dere på siste side. Det kan være lurt å ta en kikk på dette skjemaet først. Les også de tilhørende kommentarene.

*Oppgave 3.* Bruk SQL til å finne ut

1. om det er noen som er prosjektledere på flere enn ett prosjekt
2. om det er noen prosjektledere som også jobber på prosjekter der de ikke er prosjektledere
3. hvilken av de timeansatte som har høyest lønnstrinn
4. gjennomsnittslønnen til de fast ansatte fordelt på stillingsbrøk, dvs.
  - (i) gjennomsnittslønnen til heltidsansatte (dvs. med 30 timer i uken),
  - (ii) gjennomsnittslønnen til deltidsansatte med 29 timer i uken,
  - (iii) gjennomsnittslønnen til deltidsansatte med 28 timer i uken, osv.Sorter listen etter avtagende stillingsbrøk.

## ORM-modellen fra innleveringsoppgave 3:



### Relasjonsskjema:

Hvis vi bruker realiseringsalgoritmen rett frem uten å ta hensyn til kommentarene merket 1 og 2 i modellen over<sup>1</sup>, får vi:

Timeliste(timelistenr, status, [levert], [utbetalt], [timeantall], beskrivelse)

Timelistelinje(timelistenr, linjenr, startdato, starttid, [sluttid], [pause], [varighet], beskrivelse)

Følgende tabeller er undertrykt, dvs. sløyfet fordi det er lite poeng i å ha egne tabeller for disse opplysningene<sup>2</sup>:

TimelisteStatus(status)

Dato(aammdd)

Klokkeslett(hhmm)

Tidsrom(antminutter)

Beskrivelse(tekst)

Linje(linjenr)

Fremmednøkler:

Timelistelinje(timelistenr) → Timeliste(timelistenr)

<sup>1</sup> I reelle modeller er det ikke uvanlig å ha slike tekstlige skranker. Dette er skranker som ikke kan uttrykkes i ORM. For å gjøre det enklere, skal dere generelt (med mindre vi eksplisitt sier noe annet) ikke ta hensyn til tekstlige skranker under realiseringen.

<sup>2</sup> Dere skal vanligvis ikke gjengi undertrykte tabeller; de er tatt med her for å illustrere hva slags tabeller som typisk undertrykkes. Dette blir forelest skikkelig senere, så dere får ikke trekk om dere tar med samtlige tabeller i besvarelsen på akkurat denne innleveringsoppgaven.