

Oppgave 1: Begrensninger og syntaks for spesialisering

a) Vi skal nå utvide ER-modellen slik at vi kan ha entiteter som danner superklasse/subklasse-forhold. Når vi modellerer med spesialisering har vi to type begrensninger som er mulige: Total spesialisering og disjunkte subklasser. Forklar betydningen av disse to begrepene.

Total spesialisering betyr at hver entitet i en superklasse må være medlem av minst en av subklasse, og markeres med en dobbel strek. Dersom dette er frivillig kalles det delvis spesialisering.

Disjunkte subklasser betyr at en entitetsklasse kan være med i maks en underklasse, og markeres med en "D". Dersom vi ikke har noen restriksjon på dette kalles det overlappende subklasse, som markeres med "O".

b) Disse begrensningene, totalitet og disjunksjon, er uavhengige av hverandre. Det betyr at vi kan ha fire mulige situasjoner når vi modellerer med spesialisering:

i) Disjunkt og total

ii) Disjunkt og delvis

iii) Overlappende og total

iv) Overlappende og delvis

For hver av disse situasjonene, finn et eksempel på en miniverden der det er hensiktsmessig å modellere med en slik begrensning.

i) Et eksempel på en miniverden der det er hensiktsmessig å modellere med disjunkt og total begrensning er et universitet. Dersom en person i systemet ikke være fakultet og student samtidig, har vi at hver person må være registrert som student eller fakultet (total) og kan ikke være både student og fakultet (disjunkt).

ii) Et kjøretøyregistreringssystem der vi kan registrere et kjøretøy som kommersielt eller personlig kjøretøy. Her kan vi registrere et kjøretøy som kommersielt eller personlig, men ikke begge (disjunkt), men vi kan også velge å ikke registrere kjøretøyet som noen av de (delvis).

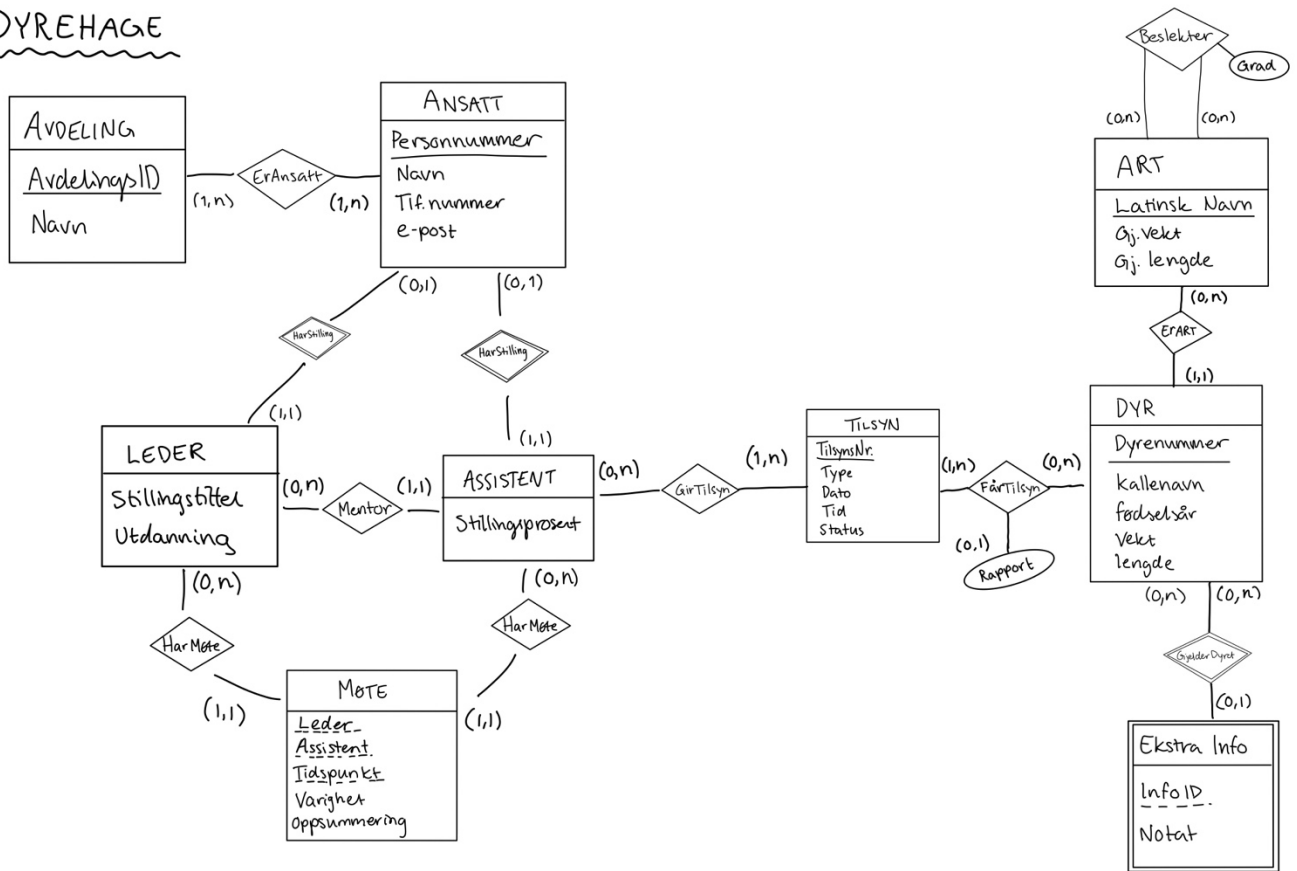
iii) Dersom vi går tilbake til universitetsmodellen kan vi også ha en total og overlappende modell. En person kan være student og eller ansatt, her endrer vi fra fakultet til ansatt og inkluderer derfor studasser som er studenter og ansatte samtidig. En person kan derfor være student, ansatt eller student og ansatt (overlappende og total).

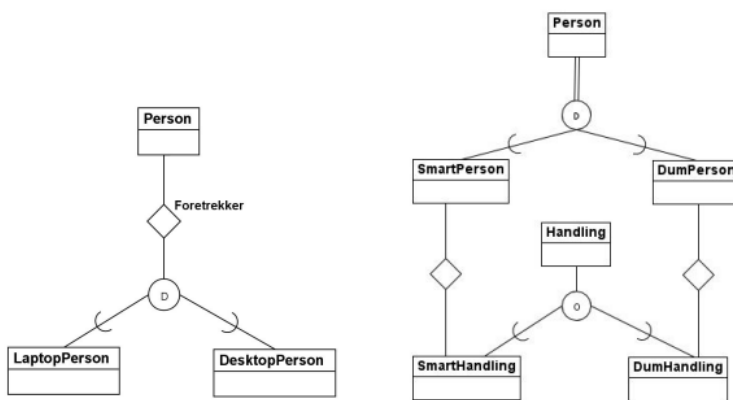
iv) Dersom vi har en nettbutikk kan en bruker være en kjøper, selger eller ikke aktiv (verken selger eller kjøper). En bruker kan være kjøper og selger samtidig (overlappende), men kan være ikke aktiv (delvis).

c) Hvilke av de følgende ER-diagrammene under (figur 1-4) er (syntaktisk) gale? Begrunn svaret ditt.

Oppgave 2: ER-modellering av dyrehage

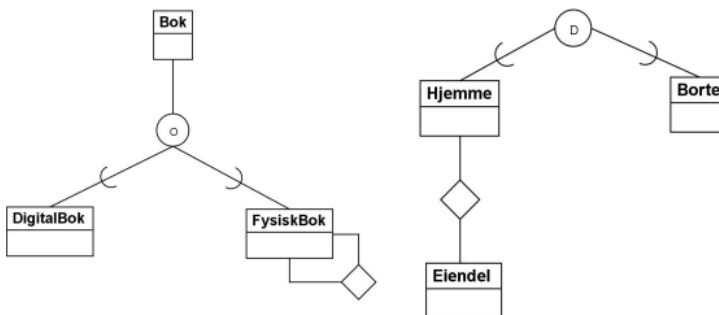
DYREHAGE





Figur 1

Figur 2



Figur 3

Figur 4

Figur 1 er ugyldig, her ser vi en superklasse-subklasse relasjon mellom superklassen Person og subklassene LaptopPerson og DesktopPerson, men vi har også en relasjons klasse Foretrekker mellom klassene som ikke gir mening å ha.

Figur 2 er gyldig, her ser vi en superklasse-subklasse relasjon mellom superklassen Person og subklassene SmartPerson og DumPerson. Vi ser også at SmartPerson og DumPerson er relatert med SmartHandling og DumHandling. Her er SmartHandling og DumHandling subclasser av Handling.

Figur 3 er gyldig, den viser en superklasse-subklasse relasjon mellom superklassen Bok og subklassen DigitalBok og FysiskBok. I tillegg er det relasjon mellom FysiskBok og FysiskBok som er lovlig.

Figur 4 er ugyldig, diagrammet viser en superklasse-subklasse relasjon, men det er ingen superklasse.

Oppgave 3: Viktige begreper i relasjonsdatabasemodellen

Forklar sammenhengen mellom primærnøkkel og entitetsintegritet, og sammenhengen mellom fremmednøkkel og referanseintegritet.

Entitetsintegritetsbegrensingen sier at en primærnøkkel ikke kan være Null. Dette er fordi primærnøkkel brukes til å identifisere individuelle tuppler i en relasjon, en Null verdi for en primærnøkkel vil derfor bety at tuplene ikke kan identifiseres.

StudentNr	Navn
-----------	------

Har eksamen:

<u>EksamenNR</u>	<u>StudentNr</u>
------------------	------------------

Eksamenslokale:

<u>RomNr</u>	Navn	Kapasitet
--------------	------	-----------

Oppsatt:

<u>EksamenNR</u>	<u>StudentNr</u>	<u>RomNr</u>	Dato	Studentplassering
------------------	------------------	--------------	------	-------------------

Bord:

<u>BordNr</u>	Type	<u>EksamenslokaleID</u>
---------------	------	-------------------------

Stol:

<u>StolNr</u>	Type	<u>EksamenslokaleID</u>
---------------	------	-------------------------

Velger å ikke bruke koblingstabell for BordPåLokale, og StolPåLokale, fordi hvis en stol finnes må den høre til et lokale, og EksamenslokaleID vil aldri ta NULL som verdi.

Ved «Oppsatt» og «HarEksamen», ville det vært mulig å se for seg at man kunne kommet i en situasjon hvor det blir NULL, dersom man brukte fremmednøkkel, framfor en koblingstabell.

b) Vi har oppgitt følgende database med tabeller:

Hotell(Hotellnr, Navn, Område)

Hotellrom(Romnr, Kvadratmeterstørrelse, Hotellnr)

Hotellnr er fremmednøkkel mot Hotell og kan ikke være NULL.

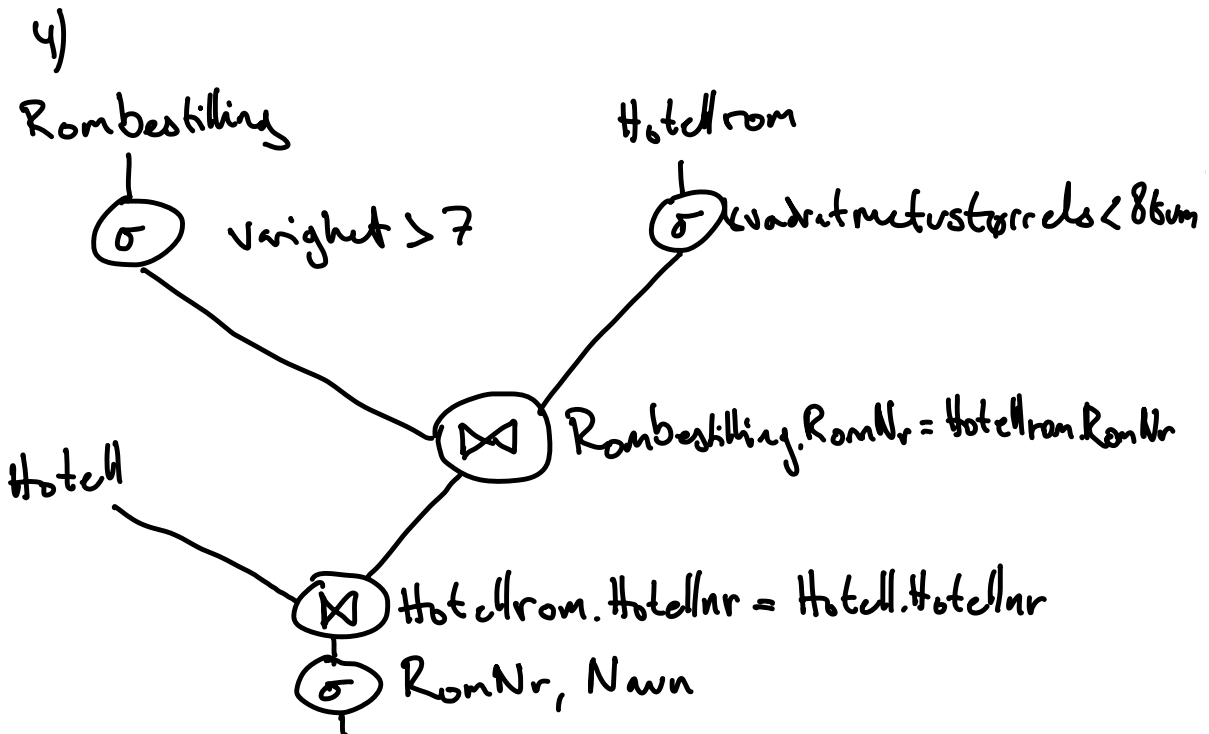
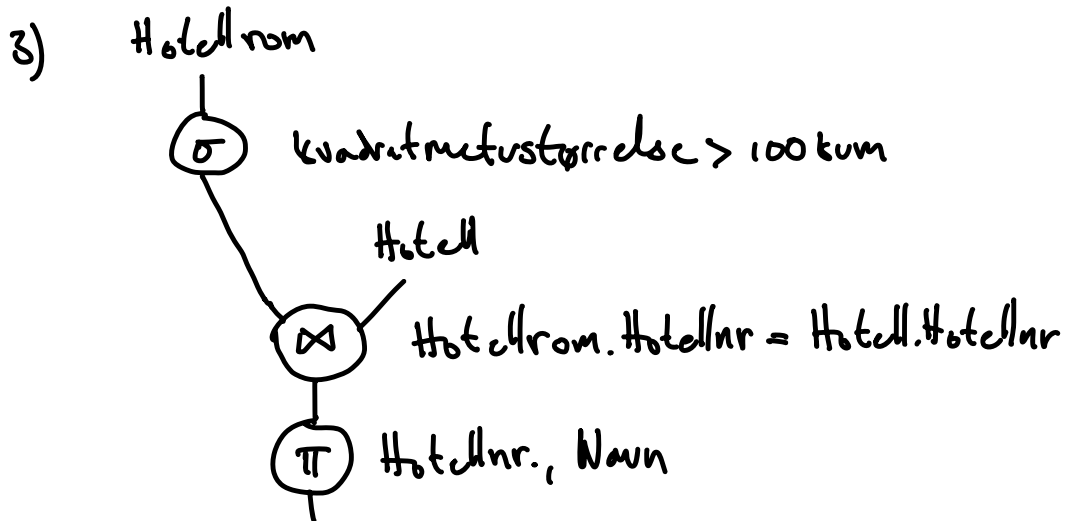
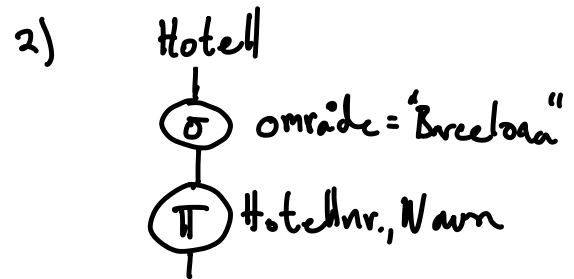
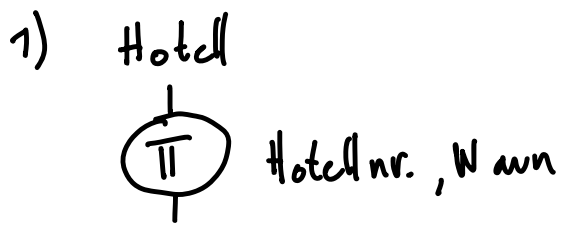
Kunde(Kundenr, Fornavn, Etternavn, Telefonnr)

Rombestilling(Bestillingsnr, Startdato, Varighet, Kundenr, RomNr)

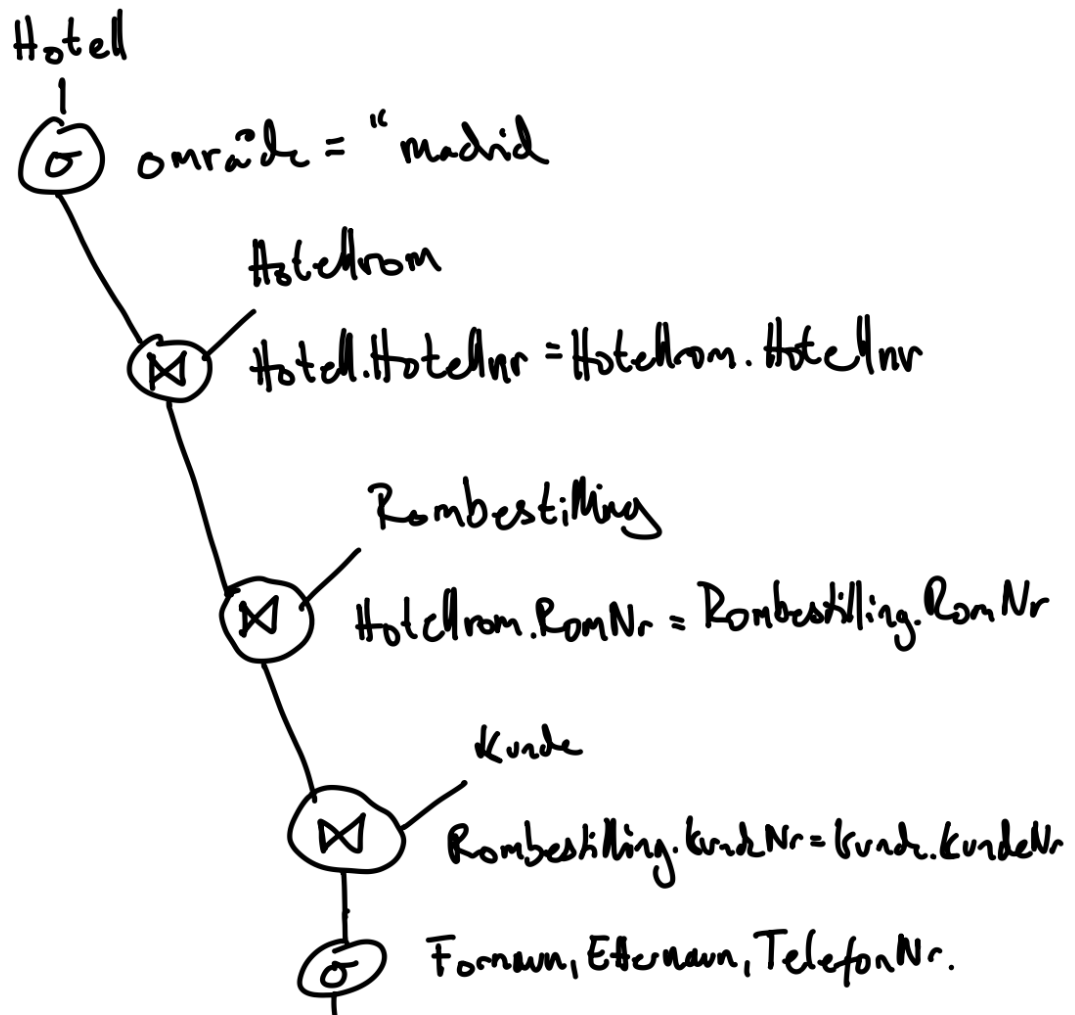
Kundenr er fremmednøkkel mot Kunde og kan ikke være NULL. RomNr er fremmednøkkel mot Hotellrom og kan ikke være NULL.

Skriv spørringer i relasjonsalgebra for databasen over, som gjør følgende:

1. Hent ut hotellnr og navn på alle hoteller.
2. Hent ut hotellnr og navn på alle hoteller som ligger i Barcelona (Område = "Barcelona")
3. Finn romnr og hotellnavn for de rom som er over 100 kvadratmeter.
4. Finn antallet bestillinger på hotellrom under 8 kvadratmeter og med varighet lengre enn 7 dager.
5. Finn fullt navn og telefonnummer på de kundene som har en rombestilling i Madrid.
6. Finn varigheten til alle rombestillinger bestilt av kunder med navn "Ole Hansen". Sorter resultatet på varighet i stigende rekkefølge.



5)



6)

