

## 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 vil si at det bare kan eksisterer entiteter i form av subklassene, ikke form av superklassen. Dersom det er en delvis spesialisering kan entitene eksistere i form av både super- og subklassene.

Disjunkt betyr at en entitet ikke kan eksistere i form av flere subklasser, den vil kun eksistere i form av én av subklassene. Overlappende derimot tillater eksistensen av entiteter innenfor flere subklasser.

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.

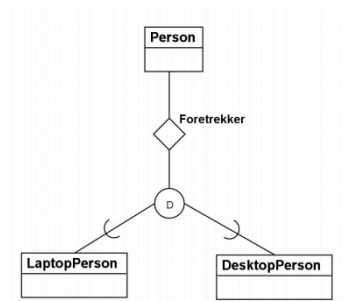
i) **Disjunkt og total:** La oss si at vi skal modellere en dyrehage der vi kun kan lagre dyreart-objekter som ape, tiger, løve osv. Da vil dyr være super klassen, og dyrearter være subklassene.

ii) **Disjunkt og delvis:** Dersom vi skal modellere en database som lagrer biler, og det opp til brukeren om han/hun vil registrere hva slags bil det er. Da vil bil være superklassen (som entiter kan eksistere innenfor), og f.eks. Lastebil, personbil og traktor vil være subklassene.

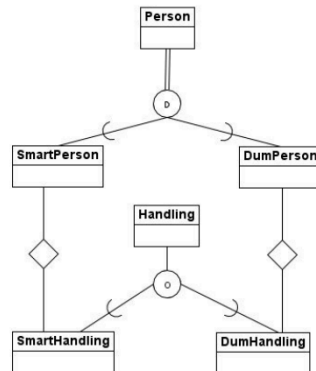
iii) **Overlappende og total:** En butikk som selger kameraer har en database som skal lagre forskjellige typer kameraer. Et kamera kan eksistere innenfor flere subklasser siden et kamera både kan være et videokamera og et DSLR kamera. Men man kan ikke lagre et kamera som superklassen siden dette ikke vil være informativt.

iv) **Overlappende og delvis:** I et legesystem vil vi lagre vanlige leger samt dermatologer og kirurger. Her er lege superklassen (som kan eksistere), og dermatolog og kirurg subklassene (hvor en lege kan både være en dermatolog og kirurg).

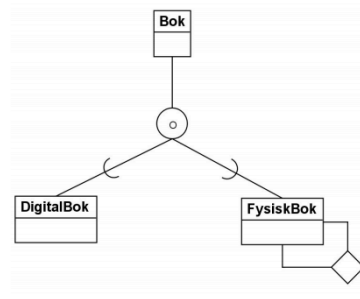
c) Hvilke av de følgende ER-diagrammene under (figur 1-4) er (syntaktisk) gale?  
Begrunn svaret ditt.



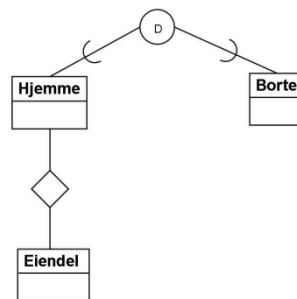
Figur 1



Figur 2



Figur 3



Figur 4

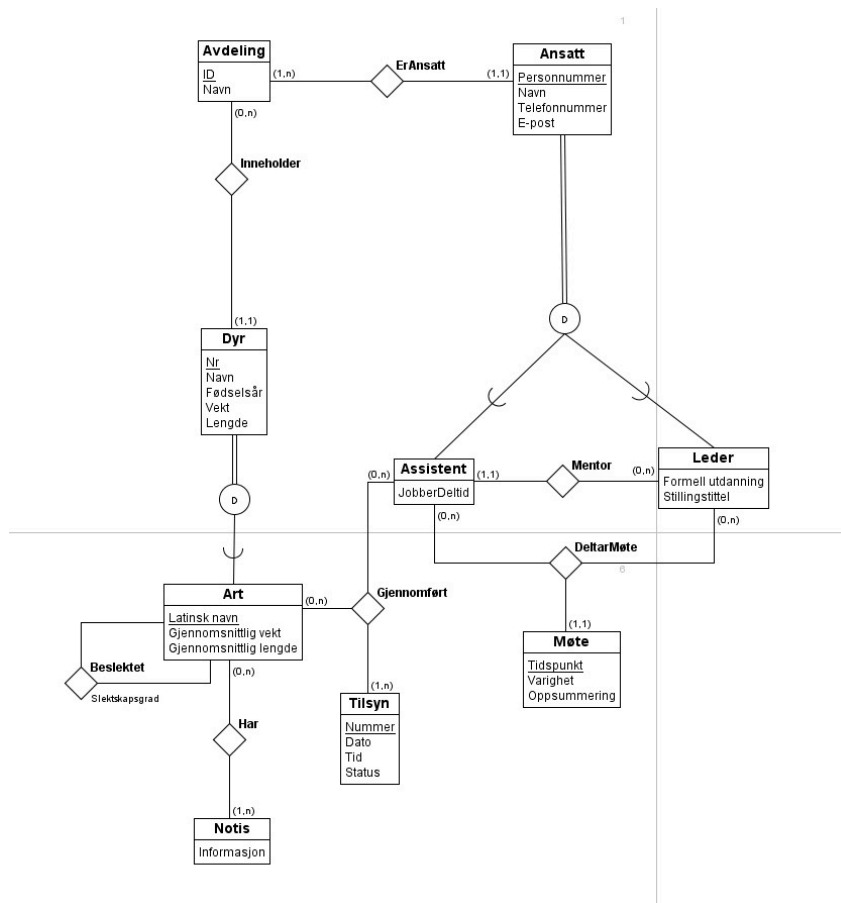
**Figur 1:** Galt, det gir ikke noe mening at et person-objekt foretrekker et laptop- eller desktopperson-objekt.

**Figur 2:** Riktig, her kan ikke en person eksistere, slik at alle personer er enten dumme eller smarte. En handling kan eksistere alene, eller eksistere som en smart og/eller dum handling (logisk sett gir ikke dette mening). En handling kan både være smart og dum avhengig av synspunktet til den som ser på det. SmartPerson kan ha en relasjon til en SmartHandling og vice versa.

**Figur 3:** Riktig, en bok kan eksistere som en bop, digitalbok og/eller fysiskbok. En fysiskbok kan ha en relasjon til en annen fysiskbok.

**Figur 4:** Galt, Hjemme og Borte kan ikke var subclasser til ingenting.

## Oppgave 2: ER-modellering av dyrehage



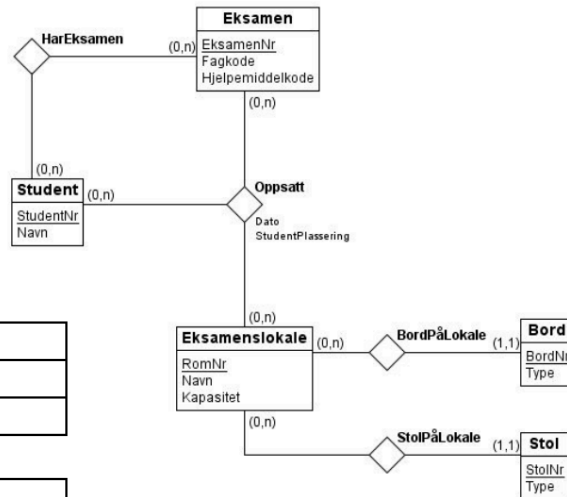
## Oppgave 3: Viktige begreper i relasjonsdatabasemodellen

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

Hver tabell har én primærnøkkel, den må være i stand til unikt å kunne identifisere enhver rad i en tabell, og kan ikke være NULL. Entitetsintegritet handler om at repetitive rader ikke er tillatt. Dermed gitt en primærnøkkel og at entitetsintegritet er opprettholdt vil alle nøkler og tilhørende verdier ha et 1-til-1 forhold slik at rett verdi alltid vil bli funnet. En fremmednøkkel brukes som en kobling mellom tabeller, der en fremmednøkkel i én tabell, er en referanse til en primærnøkkel i en annen tabell. Referanseintegritet regelen sier at alle fremmednøkler må referere til en rad som finnes i tabellen eller bare bestå av NULL-verdier. Dermed fører dette forholdet til at alle fremmednøkler referer til en entydig rad.

## Oppgave 4: Modelloversetting og relasjonsalgebra

a) Oversett dette ER-diagrammet til tabellform.



**Student**

<u>StudentNr</u>	Navn
1354686432	Erik Galler
4684640284	Nils-Martin Hillesund

**HarEksamen**

<u>StudentNr</u>	<u>EksamensNr</u>
4684640284	020696
1354686432	150799

**Eksamen**

<u>EksamensNr</u>	Fagkode	Hjelpemiddelkode
150799	TEP4100	C
020696	TDT4171	D

**Oppsatt**

<u>StudentNr</u>	<u>EksamensNr</u>	RomNr	Dato	Studentplassering
4684640284	020696	8976	15.06.2021	A-3
1354686432	150799	4322	02.05.2021	C-1

**Eksamenslokale**

<u>RomNr</u>	Navn	Kapasitet
4322	Lise	600
8976	Muren	5

**Bord** (trenger ikke relasjons tabell når kardinalitet (1,1))

<u>BordNr</u>	Type	RomNr
2398	Langbord	8976
4578	Pult	8976

**Stol** (trenger ikke relasjons tabell når kardinalitet (1,1))

<u>StolNr</u>	Type	RomNr
8721	Rullestol	4322
9998	Krakk	8976

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)

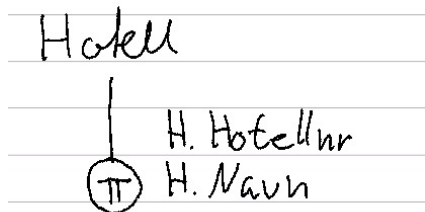
Kundenr er fremmednøkkel mot Kunde og kan ikke være NULL.

RomNr er fremmednøkkel mot Hotellrom og kan ikke være NULL.

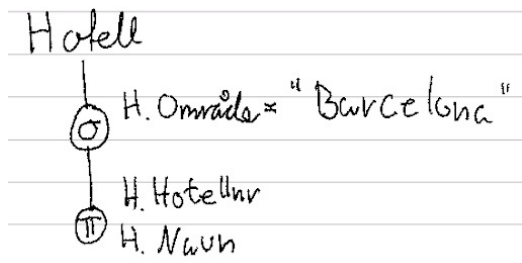
Skriv spørringer i relasjonsalgebra for databasen over, som gjør følgende:

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.

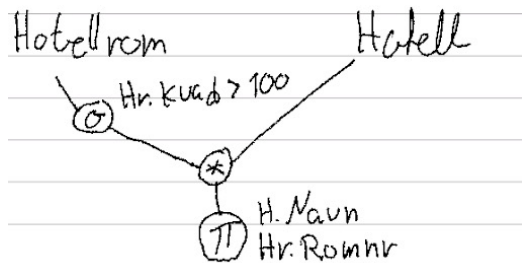
1.



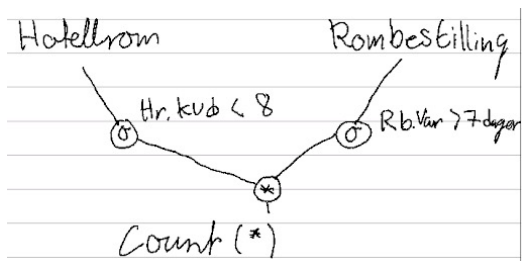
2.



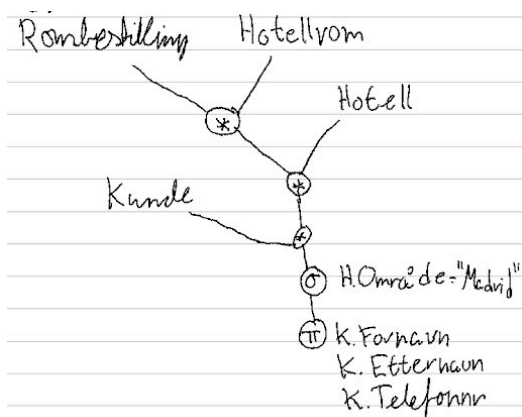
3.



4.



5.



6.

