

**Oppgave 1: Databasesystemer**

- a) En database er en samling av data med relasjoner til hverandre. DBMS er programvaren som interagerer med den fysiske representasjonen av databasen. Dvs. den håndterer innsetting, sletting og spørring av data.
- b)
- 1) Program-data uavhengighet betyr at den fysiske datastrukturen som er lagret, holdes separat fra programmene som bruker den. Fordelen med dette er at ved endringer i datastrukturen, vil man kun trenge å endre DBMS-katalogen, i stedet for å endre på alle programmene som bruker dataen.
  - 2) Flerbrukerstøtte betyr at en DBMS må tilrettelegge for at flere brukere kan aksessere databasen samtidig. DBMSen må også sørge for at hvis samme data blir endret på samtidig, må det gjøres kontrollert. Fordelen med dette er at brukere ikke trenger å vente på at andre blir ferdig, og at effektiviteten øker.
  - 3) Selvbeskrivende data lagrer informasjon om databasestrukturen og dens restriksjoner som meta-data i DBMS-katalogen, eller lagrer informasjonen som selvbeskrivende-data. Fordelen med dette er at DBMSen kan gi informasjon om databasen til brukerne, uten at de må vite dette på forhånd.

**Oppgave 2: ER-modellen**

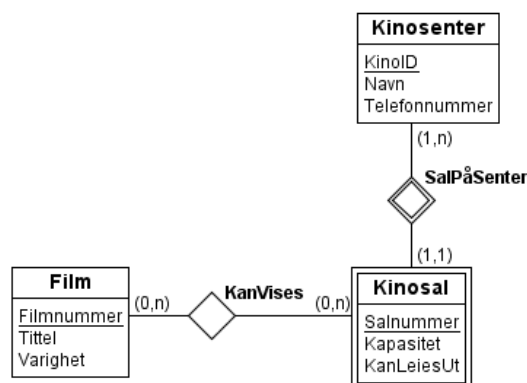
- a)
- 1) En entitet inneholder attributter og representerer et objekt i miniverden, men en entitetsklasse er en samling av entiteter med like attributter.
  - 2) En relasjon er en sammenheng mellom to eller flere entitetsklasser, og en relasjonsklasse er en mengde av slike relasjoner mellom de samme entitetsklassene.
  - 3) For å kunne indeksere all dataen i databasen må hver entitet ha en unik identifikator for sin entitetsklasse.
- b)

Påstand	Svar	Begrunnelse
1	True	Fordi det er slik ER-notasjonen ser ut. Entitetsklassen har "tittelen" Taco øverst og TacoID har en underline som indikerer at det er en nøkkelattributt.
2	True	Fordi Taco har en 0 til mange relasjon med hvert av tilbehørene
3	False	Fordi en ordre har en en til mange relasjon med Taco, som betyr at det må være minst 1 Taco i hver orde
4	True	Fordi en ordre har en en til mange relasjon med Taco, som betyr at relasjonen ikke har et maks antall bestillinger
5	False	Vi antar at hentetidspunktet vil være en tid senere enn tidbestilt og ordren vil derfor ikke være mulig å hente med en gang.

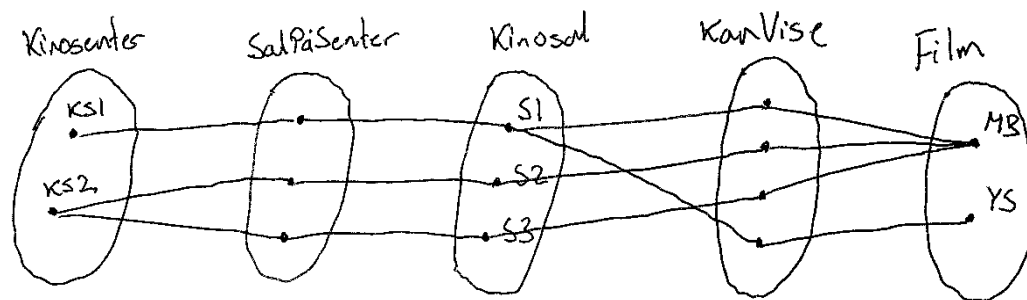
6	True	Fordi en kunde har en 0 til mange relasjon til ordre, noe som betyr at de ikke trenger å ha en orde, for å finnes i systemet.
7	False	Fordi vekten defineres i relasjonen til en taco, det kan dermed finnes grønnsaker uten en vekt.
8	True	En ansatt har en 1 til mange relasjon til butikker hvor stillingen blir definert i relasjonen. Dette betyr at en ansatt kan ha flere stillinger i flere butikker.
9	Maybe	Vi antar at den ansatte som ordren delegeres til må være tilgjengelig. Det vil derfor kunne være tilfeller der den samme ansatte får alle ordre, men i hovedsak vil den velges vilkårlig.
10	Maybe	Fordi navn-attributter kan ha en null-verdi.

### Oppgave 3: Svake klasser, forekomstdiagram og nye krav

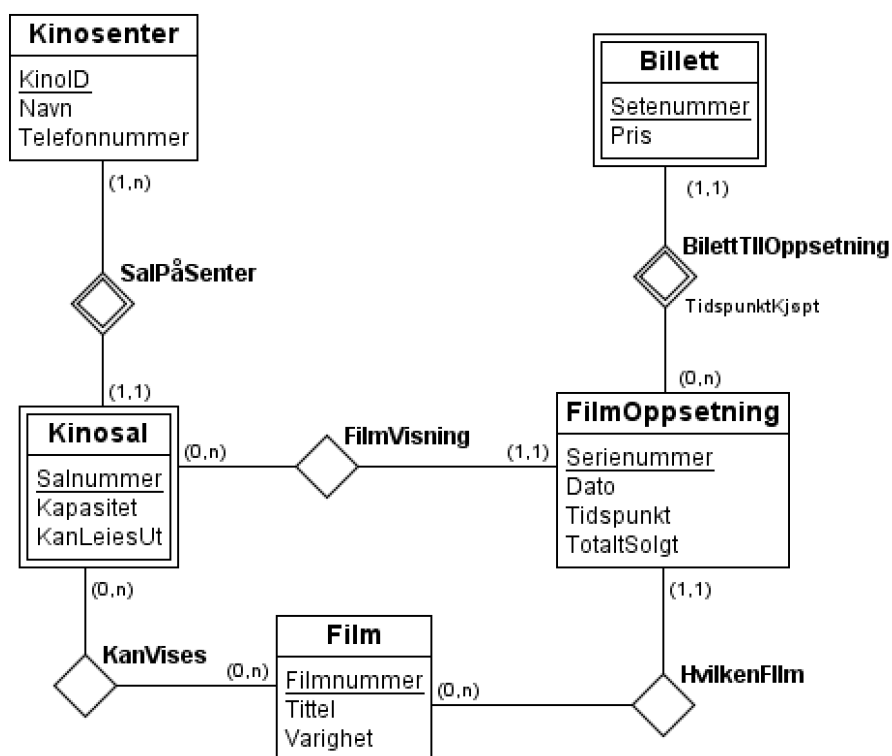
- Svake klasser vil være hensiktsmessige der vi ikke har noen naturlig unik identifikator for entitetsklassen slik som ved en kinosal. Det kan være flere kinosaler med samme salnummer. Den identifiserende entitetsklassen blir Kinosenter da det er denne som hjelper oss å gi Kinosal en unik identifikator. Den identifiserende relasjonsklassen blir SalPåSenter og attributten Salnummer er en delvis nøkkel.
- Hvis kardinaliteten endres fra (1,1) vil ikke nøkkelen være unik lenger og vil heller ikke regnes som en svak entitetsklasse. Kinosalen må ha et Kinosenter for å være unik. Og hvis den har flere Kinosenter vil den heller ikke være unik da det kan tenkes at andre Kinosenter vil ha en annen Kinosal med samme salnummer (dette gir jo ingen mening heller).
- 



d)



e)



Vi antar at FilmOppsetning entitetsklassen teller antall billetter solgt ved å se på antall entries billetter. TotaltSolgt blir dermed en avledet attributt. Man kan så selv regne ut hva man har tjent på oppvisningen ved å se på TotaltSolgt og Pris.

#### Oppgave 4: Fra miniverden til ER-modell

Antakelse 1: Antar at en person hverken må bo eller eie en leilighet. Antar at et bygg må ha minst en leilighet.

Antakelse 2: Antar at en bod kun kan tilhøre én leilighet.

Antakelse 3: En person kan bare være leder i et styre.

Antakelse 4: Antar at styret vil ha styremedlemmer i tillegg til ledere.

Antakelse 5: Antar at saksnummer er unikt.

