

# TDT4145 – Datamodellering og databasesystemer: Prosjektoppgave Del 1

ER-DIAGRAM, RELASJONSDATABASE OG SQL

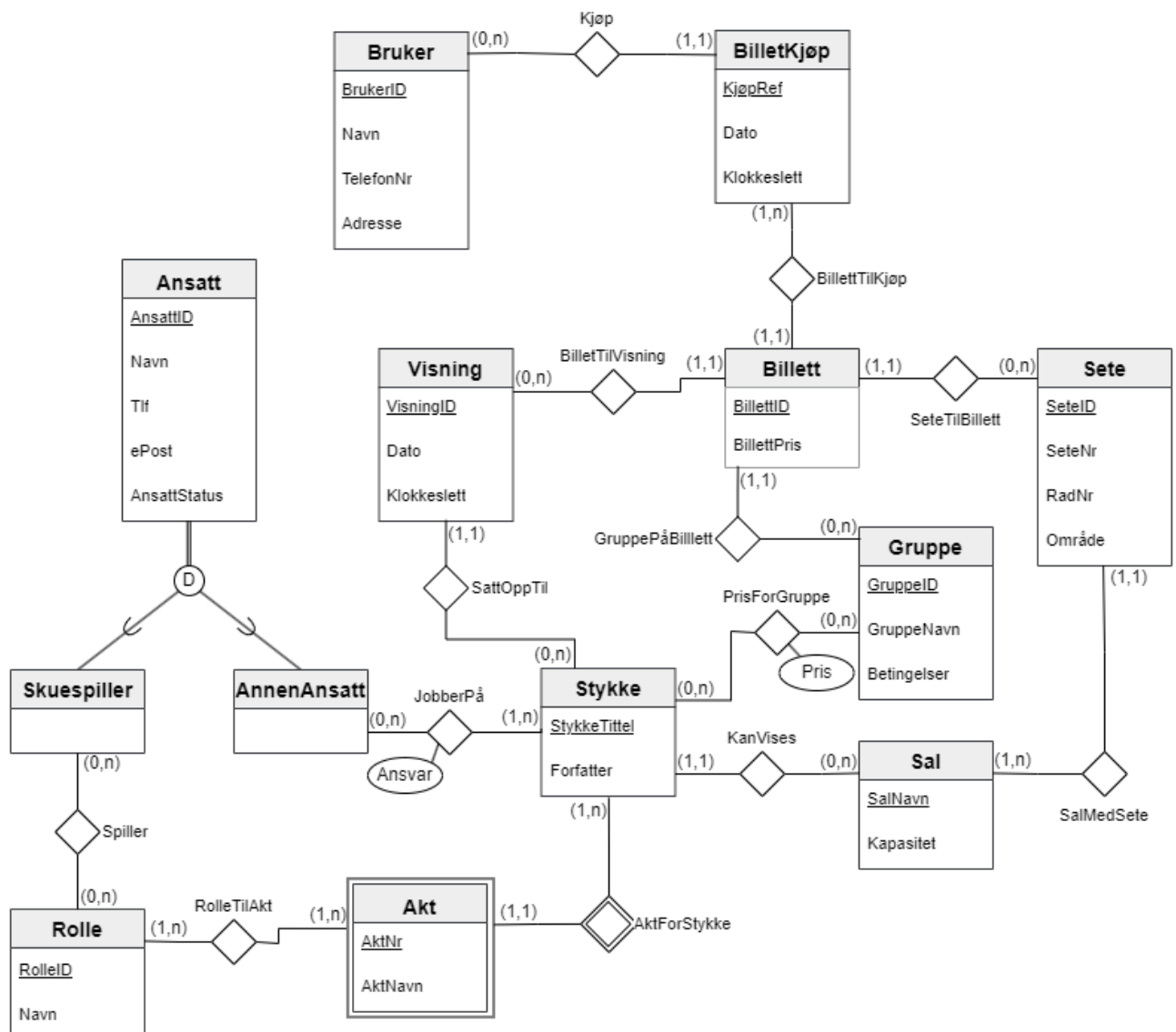
MARIUS BJERKE STJERNSTEDT

GRUPPE: 254

## Innhold

Del 1 – Part 1: ER-Diagram .....	2
Forutsetninger og antagelser .....	2
Del 1 – Part 2: Relasjonsdatabaseskjema .....	3
Tabelloversikt .....	3
Normalform for tabeller .....	4
Del 1 – Part 3: SQL: betingelser og antagelser .....	6

## Part 1: ER-Diagram



## Forutsetninger og antagelser

- En bruker kan ligge i systemet uten å ha utført et billett-kjøp
- Grupper kan ligge i systemet uten noen form for tilknytning til et stykke
- En rolle må være knyttet til en akt, men må ikke være knyttet til en skuespiller for å ligge i systemet
- Et stykke må ha minst en arbeider sammen med en sal lagt inn for å kunne ligge i databasen
- Alle ansatte i databasen må ikke nødvendigvis delta som skuespiller eller andre ansatte til de opplagte stykkene
- Alle seter ligger i databasen, men må være i en sal for å eksistere

- Siden vi vet at «Kongsemmene» bare vises på Hovedscenen og at «Størst av alt er kjærligheten» bare vises på Gamle scene, vil det alltid være en 1 til 1 knytning fra stykke til sal, men dekan være flere saler i databasen uten at det skal foregå stykker i dem.
- PrisForGruppe er en egen relasjon som brukes for å lage en pris-tabell for forskjellige stykker og gruppeklasser uavhengig av individuelle billetter. Det antas også at verdien til attributtet «BillettPris» i Billett klassen hentes prislisen og settes in her gjennom et ekstern system når en billett generers i systemet.
- Det antas at en ansatt er enten skuespiller eller annen ansatt. En ansatt kan for eksempel ikke spille en rolle i et stykke og samtidig være snekker i et annet ett
- Det antas også at en ansatt kan bare ha ett type ansvar til et stykke. i.e. den ansatte kan ikke både være lydtekniker og statist.
- Antar at et stykke må ha minst en akt, hvis det ikke er akter har stykke antall akter lik 1.
- Det kan være flere roller med samme navn, men de er fortsatt unike roller. Dersom samme rolle oppstår i to forskjellige stykker, vil de være to innslag i roll-tabellen slik at de kan knyttes mot to ulike skuespillere og ulike akter.

## Part 2: Relasjonsdatabaseskjema

### Tabelloversikt

#### Gjeldene for alle tabeller:

- Atributt(er) understreket er primærnøkkel for tabellen

**Bruker**(BrukerID, Navn, Tlf, Adresse)

**Sal**(SalNavn, Kapasitet)

**Gruppe**(GruppelD, GruppeNavn, Betingelser)

**Rolle**(RolleNavn)

**Skuespiller**(AnsattID, Navn, Tlf, E-Post, AnsattStatus)

**AnnenAnsatt**(AnsattID, Navn, Tlf, E-Post, AnsattStatus)

**BillettKjøp**(KjøpRef, Dato, Klokkeslett, BrukerID)

- BrukerID er FK mot **Bruker**(BrukerID), kan ikke ha NULL verdi

**Stykke**(StykkeTittel, Forfatter, SalNavn)

- SalNavn er FK mot **Sal**(SalNavn) , kan ikke ha NULL verdi

**Visning**(VisningsID, Dato, Klokkeslett, StykkeTittel)

- StykkeTittel er FK mot **Stykke**(StykkeTittel) , kan ikke ha NULL verdi

**Sete**(SeteID, SeteNr, RadNr, Omeråde, SalNavn)

- SalNavn er FK mot **Sal**(SalNavn) , kan ikke ha NULL verdi

**Billett**(BillettID, BillettPris, BillettRef, VisningsID, SeteID)

- BillettRef er FK mot **BillettKjøp**(BillettRef) , kan ikke ha NULL verdi
- VisningsID er FK mot **Visning**(VisningsID) , kan ikke ha NULL verdi
- SeteID er FK mot **Sete**(SeteID) , kan ikke ha NULL verdi

**Akt**(AktNr, AktNavn, StykkeTittel)

- StykkeTittel er FK mot **Stykke**(StykkeTittel) , kan ikke ha NULL verdi

**PrisForGruppe**(StykkeTittel, GruppelD, Pris)

- GruppelD er FK mot **Gruppe**(GruppelD) , kan ikke ha NULL verdi
- StykkeTittel er FK mot **Stykke**(StykkeTittel) , kan ikke ha NULL verdi

**GruppePåBillett**(BillettID, GruppelD)

- GruppelD er FK mot **Gruppe**(GruppelD) , kan ikke ha NULL verdi
- BillettID er FK mot **Billett**(BillettID) , kan ikke ha NULL verdi

**JobberPå**(AnsattID, StykkeTittel, Ansvar)

- AnsattID er FK mot **Ansatt**(AnsattID) , kan ikke ha NULL verdi
- StykkeTittel er FK mot **Stykke**(StykkeTittel) , kan ikke ha NULL verdi

**Spiller**(AnsattID, RolleID)

- AnsattID er FK mot **Ansatt**(AnsattID) , kan ikke ha NULL verdi
- RolleID er FK mot **Rolle**(RolleID) , kan ikke ha NULL verdi

**RolleTilAkt**(RolleID, AktNr)

- RolleID er FK mot **Rolle**(RolleID) , kan ikke ha NULL verdi
- AktNr er FK mot **Akt**(AktNr) , kan ikke ha NULL verdi

## Normalform for tabeller

### Bruker-Tabellen

- Tabellen inneholder en kandidatnøkkel med ett nøkkelattributt. Tabellens andre attributter er bestemt av nøkkelattributtet **BrukerID**. Dette betyr at det er ingen delvis avhengighet av kandidatnøkler. Tabellen er dermed på andre normal form (2NF).
- Man kan videre se at alle funksjonelle avhengigheter i tabellen er fra kandidatnøkkelen til alle ikke nøkkel attributter, hvor kandidatnøkkelen alltid vil være en supernøkkel. Tabellen er derfor på tredje normalform (3NF) og Boyce-Codd normalform (BCNF)
- Tabellen har ingen «multi-value dependency» (MVD-er), grunnet at den ikke har noen flerverdi attributter.
- Tabellen er både på BCNF og inneholder ingen MVD-er kan man konkludere at tabellen er på fjerde normalform (4NF).

### Rolle-Tabellen

- Tabellens ikke nøkkelattributter er bestemt av en kandidatnøkkel med ett nøkkelattributt (RolleID), som sier oss at det er ingen delvis avhengigheter for av kandidatnøkler. Rolle tabellen er på 2NF.
- Siden det er kun en funksjonell avhengighet mellom kandidatnøkkel og ikke-nøkkel attributt vil kandidatnøkkelen være en supernøkkel som sier oss at tabellen er på BCNF
- Ingen flerverdi attributter (MVD-er) sammen med BCNF gir oss at tabellen er på 4NF.

### Skuespiller-Tabellen

- Tabellen inneholder en kandidatnøkkel med ett nøkkelattributt. Tabellens andre attributter er bestemt av nøkkelattributtet AnsattID. Dette betyr at det er ingen delvis avhengighet av kandidatnøkler. Tabellen er dermed på andre normal form (2NF).
- Man kan videre se at alle funksjonelle avhengigheter i tabellen er fra kandidatnøkkel til alle ikke nøkkel attributter, hvor kandidatnøkkel alltid vil være en supernøkkel. Tabellen er derfor på tredje normalform (3NF) og Boyce-Codd normalform (BCNF)
- Tabellen har ingen «multi-value dependency» (MVD-er), grunnet at den ikke har noen flerverdi attributter.
- Tabellen er både på BCNF og inneholder ingen MVD-er kan man konkludere at tabellen er på fjerde normalform (4NF)

### AnnenAnsatt-Tabellen

- Tabellen er tilnærmet identisk som skuespiller tabellen. Tabellen er på 4NF, tilsvarende som Skuespiller tabellen.

### Gruppe-Tabellen

- Bare en kandidatnøkkel med ett nøkkelattributt (GruppeID) som bestemmer andre attributter (GruppeNavn & Betingelser). Fra samme begrunnelse som for Bruker tabellen kan vi konkludere at Gruppe tabellen er på 4NF.

### BillettKjøp-Tabellen

- Bare en kandidatnøkkel med ett nøkkelattributt (KjøpRef) som bestemmer andre attributter (Dato & Klokkeslett). Fra tilsvarende begrunnelse som for Bruker tabellen kan vi konkludere at BillettKjøp tabellen er på 4NF.

### Stykke-Tabellen

- Tabellen har en kandidatnøkkel som består av ett nøkkelattributt, StykkeTittel. Det er bare nøkkelattributtet som bestemmer det attributtet (Forfatter). Fra tilsvarende begrunnelse som for Bruker tabellen kan vi konkludere at Stykke tabellen er på 4NF.

### Visning-Tabellen

- Bare en kandidatnøkkel med ett nøkkelattributt (VisningsID) som bestemmer andre attributter (Dato & Klokkeslett). Fra tilsvarende begrunnelse som for Bruker tabellen kan vi konkludere at Visnings tabellen er på 4NF.

### Sal-Tabellen

- Bare en kandidatnøkkel med ett nøkkelattributt (SalNavn) som bestemmer andre attributter (Kapasitet). Fra samme begrunnelse som for Bruker tabellen kan vi konkludere at Sal tabellen er på 4NF.

### **Sete-Tabellen**

- Bare en kandidatnøkkel med ett nøkkelattributt (SeteID) som bestemmer andre attributter (SeteNr, RadNr & Områd). Fra samme begrunnelse som for Bruker tabellen kan vi konkludere at Sete tabellen er på 4NF.

### **Billett-Tabellen**

- Bare en kandidatnøkkel med ett nøkkelattributt (BillettID) som bestemmer andre attributter (BillettPris). Fra samme begrunnelse som for Bruker tabellen kan vi konkludere at Billett tabellen er på 4NF.

### **Akt-Tabellen**

- Tabellen har en kandidatnøkkel som består av to nøkkelattributter (AktNr & StykkeTittel), med ett ikke-nøkkel attributter (AktNavn). Grunnet tilsvarende begrunnelse som for Bruker tabellen og at det vil bare tilsvare et AktNav til en kandidatnøkkel er tabellen på 4NF.

### **JobberPå-Tabellen**

- Tabellen har en kandidatnøkkel som består av to nøkkelattributter (AnsattID & StykkeTittel). Siden tabellen har ett ikke-nøkkel attributter som er bestemt av kandidatnøkkel og ingen MVD-er, kan vi konkludere at tabellen er på 4NF.

### **Spiller-Tabellen**

- Tabellen har en kandidatnøkkel som består av to nøkkelattributter (AnsattID & RolleID). Siden tabellen har ingen ikke-nøkkel attributter og ingen MVD-er kan vi konkludere at tabellen er på 4NF.

### **RolleTilAkt-Tabellen**

- Tabellen har en kandidatnøkkel som består av to nøkkelattributter (RolleID & AktNr). Siden tabellen har ingen ikke-nøkkel attributter og ingen MVD-er kan vi konkludere at tabellen er på 4NF.

### **PrisForGruppe-Tabellen**

- Tabellen har en kandidatnøkkel som består av to nøkkelattributter (StykkeTittel & GruppeID). Siden tabellen har ett ikke-nøkkel attributter som er bestemt av kandidatnøkkel og ingen MVD-er, kan vi konkludere at tabellen er på 4NF.

### **GruppePåBillett-Tabellen**

- Tabellen har en kandidatnøkkel som består av to nøkkelattributter (BillettID & GruppeID). Siden tabellen har ingen ikke-nøkkel attributter og ingen MVD-er kan vi konkludere at tabellen er på 4NF.

## **Part 3: SQL: betingelser og antagelser**

- SQL skripte for å lage de tomme tabellene til databasen er vedlagt som en SQL tekstfil TeaterDB.

- Grunnet mangel på støtte av «Ø» og «Å», er disse bokstavene heller representert med «O» og «A» i TeaterDB scriptet.
- Det brukes « ON UPDATE CASCADE» for alle fremmednøkkeler slik at hvis fremmednøkkelen til en tabell oppdateres i en annen tabell, vil fremmednøkkelen også oppdateres.
- «ON DELETE CASCADE» er normalt brukt. Utføres i tilfeller hvor det gir mening å fjerne dataen hvis dataen til fremmednøkkelen er fjernet.
- Bruken av « ON DELETE SET NULL» med tanke på spesifikke tilfeller hvor en fremmednøkkel slettes fra systemet, men vi vil fortsatt opprettholde data andre steder i systemet. Eksempel på dette er at en bruker har hatt diverse billett kjøp, men bestemmer seg for å slette brukeren sin. Her vil BrukerID settes til NULL i BillettKjøp tabellen slik at man fortsatt kan se alle tidligere billetter kjøpt. Selv om brukeren ikke eksisterer lenger.
  - Steder der dette blir brukt er
    - BillettKjøp
    - Stykke
    - VisningsID og SeteID i Billett
- Det brukes « ON DELETE NO ACTION» for fremmednøkkeler i tabeller der det ikke gir mening å kunne fjernes.
  - Dette er brukt i
    - Stykke
    - Visning
    - Sete
- Antar at alle telefonnummer er norske og bruker derfor VARCHAR(8) for dette.
- Bruker standar datatype for epost adresser som VARCHAR(320), 64 karakterer for den locale delen av mailen, en karakter for «@» symbolet, og 254 karakterer for domenenavnet.
- Grunnet SQLite's mangel på støtte for TIME og DATE datatypene er heller henholdsvis VARCHAR(5) for «hh:mm» og VARCHAR(10) for «dd.mm.yyyy» brukt i stede.
- Har også definert atributter som ikke kan ha «NULL» verdi ved bruk av «NO NULL» constraint.