



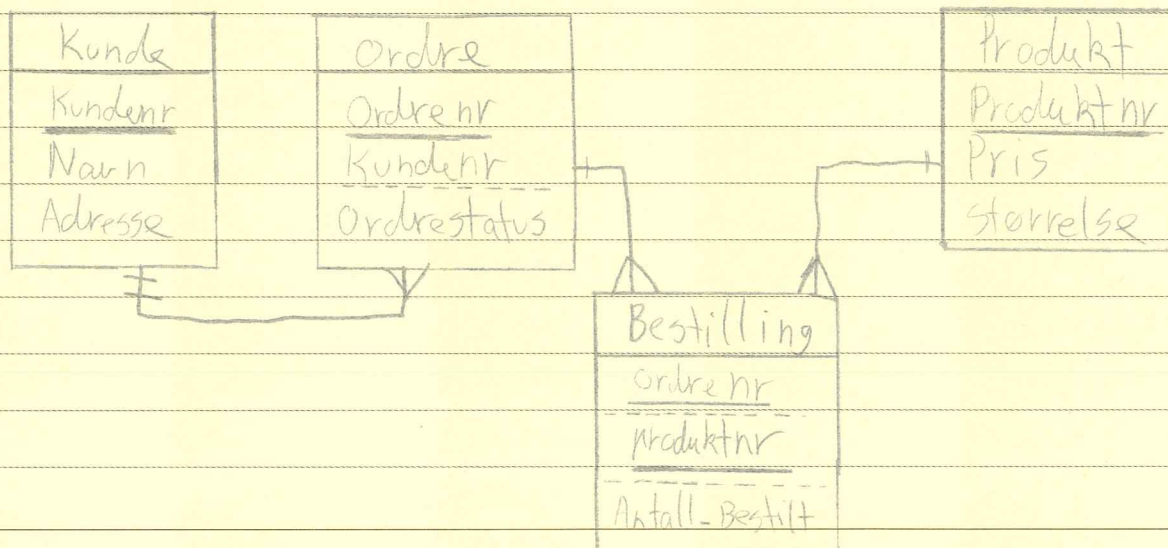
Emnekode : 15 - 201
Kandidatnr. : 4237
Dato : 26-11-2013
Ark nr. : 1 av 7

①

a) Ulempen med filprosesseringsystem er at det er lett å overse hvilke data som mangler, og at selve spørringen kan ta lenger tid å utføre. Når det gjelder med hensyn til applikasjon og data så kan f.eks fordel være at du jobber rett mot databasen slik at spørring går fortere og har kontroll på hvilke data som går inn og ut. Her trenger du heller ikke lagre data i fil for så å overføre til en annen disk f.eks.

b) Mange til mange sammenheng i en ER modell gjøres ved at du lager en ny tabell i mellom dem, deretter tar du primærnøklerne fra tabellene og gjør dem til sammensatt primærnøkkel og fremmednøkkel i den nye.

Eksempel: Vi lager en ny tabell, den heter bestilling.





Emnekode : 15-201
 Kandidatnr. : 4237
 Dato : 26-11-2013
 Ark nr. : 2 av 7

①

② Commit lagrer og utfører dataene slik at noen annen bruker kan gå inn ved en senere anledning kan gå inn å bruke eller se og endre dataene.

Rollback "Ruller" dataene tilbake til forrige versjon av databasen, altså COMMIT Det er ikke alle databaser som støtter Rollback, InnoDB f.eks, denne gjør dette.

Eksempel:

select * FROM kunde; →

KundeID	Navn	TLF
1	OLE	98617133

INSERT INTO kunde (KundeID, Navn, TLF)
 Values ('2', 'PER', '113');

KundeID	Navn	TLF
1	OLE	98617133
2	PER	113

ROLLBACK;

select * FROM kunde; →

KundeID	Navn	TLF
1	OLE	98617133

INSERT INTO kunde (KundeID, Navn, TLF)
 Values ('2', 'PER', '113');

KundeID	Navn	TLF
1	OLE	98617133
2	PER	113

COMMIT;

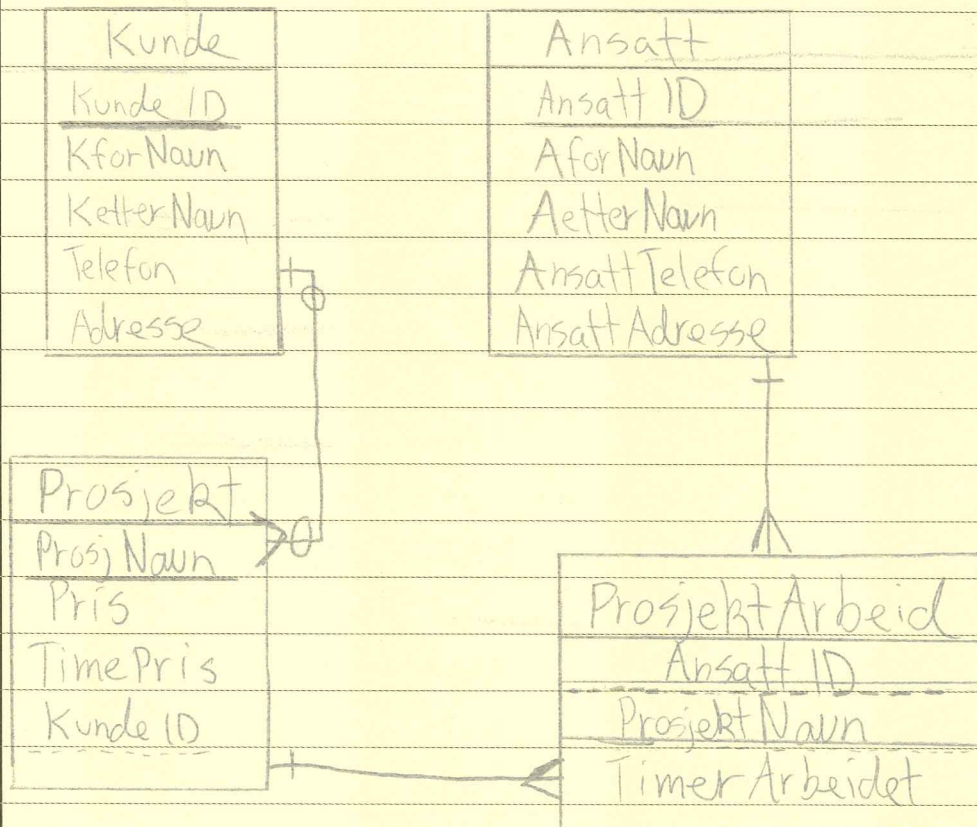
ROLLBACK;

SELECT * FROM kunde; →

① d) En Constraint er en Regel/Betingelse i databasen som ikke brukeren kan bryte. For eksempel hvis du setter KundeID til å være PN og INT, så må altså kundeID inneholde ett tall og ingen bokstaver.

Eksempel: Alter Table Kunde
 Add Constraint KundeID_pk PRIMARY KEY (KundeID)

②





③ a) Primær Nøkklene i tabellen kan og bør mest sannsynlig være ClientNo, propertyNo og ownerNo. Men på grunn av at rapporten/tabellen er i 1nf så fører det til duplikasjoner og dermed fungerer ikke primærnøkklene så bra.

b) propertyNo, pAddress er noen av flerverdi attributtene.

c) For å få tabellen på 1NF må vi finne de funksjonelle avhengighetene. Spørsmålet vi kan stille oss for å finne det er: For ex X, hvor mange Y? Svaret er maks 1.

d) Tabellen (clientNo, cName, propertyNo, pAddress, rent_start, rent_finish, rent, ownerNo, oName)

Bytte om plassering

e) clientNo → cName
propertyNo → pAddress
rent → rent_start, rent_finish
ownerNo → oName, propertyNo



Emnekode : 15-201
Kandidatnr. : 4237
Dato : 26-11-2011
Ark nr. : 5 av 7

③

f) 2NF:

Owner (ownerNo, oName, oPhone, oAddress)
property (propertyNo, pAddress, ~~rentPrice~~, clientNo)
Client (clientNo, clientName, ownerNo)
Rent (rentPrice, rentStart, rentFinish)
LoanOut (propertyNo, rentPrice, ownerNo)

3NF: i 3NF skal transitiv avhengighet bli funnet, feks hvis postnr og poststed hadde vært med så hadde det vært Transitiv avhengighet, siden vi ikke fant noen, her nå så er det 2NF som gjelder.



Emnekode : 15-201
Kandidatnr. : 4237
Dato : 26-11-2013
Ark nr. : 6 av 7

④

a) select LastName from Patient
INNER JOIN Nurse ON patient.NurseName
= Nurse.NurseName INNER JOIN Ward
ON Nurse.WardNo = Ward.WardNo
WHERE WardNo = '12';

b) select LastName from Patient
WHERE PatientNo NOT IN (select patientNo
FROM patient INNER JOIN Diagnosis ON
patient.PatientNo = Diagnosis.PatientNo);

c) Select (COUNT(NoOfBeds) AntallSenger,
COUNT(PatientNo) AntallPasient, SUM(AntallSenger-
AntallPasient)) FROM Ward INNER JOIN
Nurse ON Ward.WardNo = Nurse.WardNo
INNER JOIN Patient ON Nurse.NurseName
= patient.NurseName
GROUP BY WardNo;

- Jeg tolket oppgaven slik at det
skulle sjekkes hvor mange senger
som var TILGJENGELIG i hver avdeling,
derfor skrev jeg spørringen slik.



Emnekode : 15-201
Kandidatnr. : 4237
Dato : 26-11-2013
Ark nr. : 7 av 7

④

d) Et VIEW er en virtuell tabell laget av en spørring. Den viser dataene, grunnen til at vi bruker et database database er for å begrense adgang til forskjellige brukere, slik at noen brukere bare får se enkelte data som vi ønsker. Med andre ord så brukes view for å gjøre database sikrere.

Eksempel på View: CREATE VIEW Pasient as
(
Select PatientNo, FirstName, LastName FROM
Patient INNER JOIN Nurse ON Patient.NurseName =
Nurse.NurseName INNER JOIN Ward ON
Nurse.WardNo = Ward.WardNo
ORDER BY LastName asc;