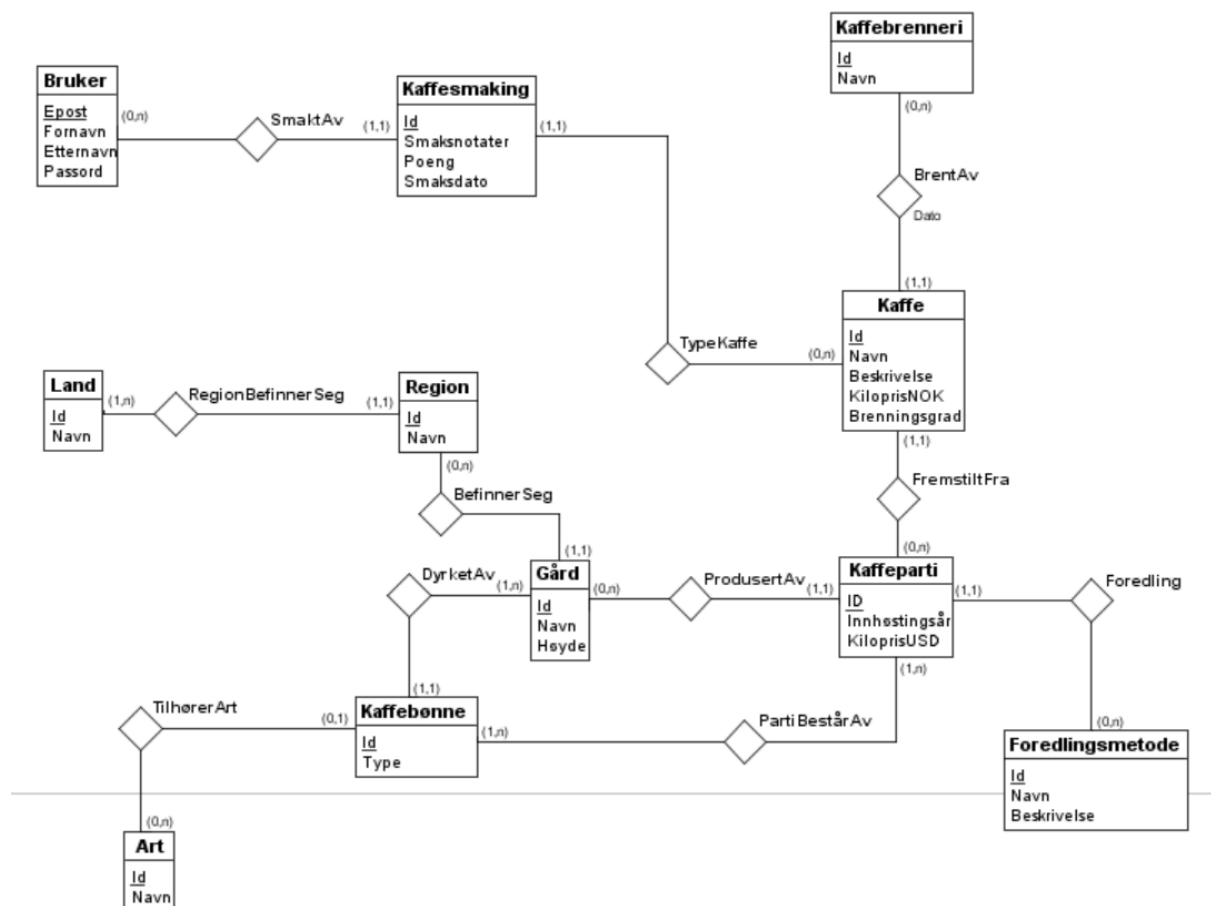


### Oppgave a



I denne ER-modellen har vi gjort noen antagelser, og disse lyder som følger:

- Vi antar at to brukere ikke skal kunne ha samme epost, noe som gjør epost til en god primærnøkkel som oppfyller kravet om entitetsintegritet.
- Fullt navn på *bruker* er delt inn i fornavn og etternavn for å forsikre at attributtet vil være atomisk.
- Antar at en *kaffe* må komme fra ett og bare ett *kaffebrenneri*, derav kardinaliteten på (1,1).
- Antar at en *kaffe* sin brenningsgrad ikke kan være NULL.

- Antar at det kan finnes ubrukte foredlingsmetoder, derav kardinaliteten (0,n) til kaffeparti.
- Antar at en *region* befinner seg i et land, og et *land* må bestå av minst én region, men gjerne flere. Videre har vi bestemt at en *gård* må befinne seg i en *region*, og en *region* kan inneholde alt fra ingen til mange gårder.
- Antar at en *kaffebønne* som inngår i et *kaffeparti* må være foredlet.
- Antar at en foredlet *kaffebønne* ikke har noen *art*. Denne antagelsen kommer på bakgrunn av oppgaveteksten.

## Oppgave b

**Bruker**(Epost, Fornavn, Etternavn, Passord)

Bruker er på BCNF siden alle funksjonelle avhengigheter er av Epost, og Epost er en supernøkkel.

**Kaffesmaking**(Id, Smaksnotater, Poeng, Smaksdato, BrukerEpost, Kaffeld)

BrukerEpost og Kaffeld er fremmednøkler mot henholdsvis Bruker og Kaffe, og de kan ikke være NULL.

Kaffesmaking er på BCNF siden alle funksjonelle avhengigheter er fra Id, og Id er en supernøkkel

**Kaffebrenneri**(Id, Navn)

Kaffebrenneri er på BCNF siden eneste funksjonelle avhengighet er Id -> Navn, og Id er en supernøkkel

**Kaffe**(Id, Navn, Beskrivelse, KiloprisNOK, Brenningsgrad, Dato, Kaffepartild, Kaffebrennerild)

Kaffepartild og Kaffebrennerild er fremmednøkler mot henholdsvis Kaffeparti og Kaffebrenneri, og de kan ikke være NULL  
Dato henviser til Dato-attributtet i relasjonsklassen BrentAv til Kaffebrenneri

Kaffe er på BCNF siden alle funksjonelle avhengigheter er av Id, og Id er en supernøkkel.

**Land**(Id, Navn)

Land er på BCNF siden eneste funksjonelle avhengighet er Id -> Navn, og Id er en supernøkkel

**Region**(Id, Navn, LandId)

LandId er fremmednøkkel mot Land og kan ikke være NULL.

Region er på BCNF siden begge funksjonelle avhengigheter i tabellen er av Id, og Id er en supernøkkel.

**Gård**(Id, Navn, Høyde, RegionId)

RegionId er en fremmednøkkel mot Region og kan ikke være NULL.

Gård er på BCNF siden alle funksjonelle avhengigheter i tabellen er av Id, og Id er en supernøkkel.

**Kaffeparti**(Id, Innhøstingsår, KiloprisUSD, GårdId, ForedlingsmetodeId)

GårdId og ForedlingsmetodeId er fremmednøkler mot henholdsvis Gård og Foredlingsmetode, og de kan ikke være NULL

Kaffeparti er på BCNF siden alle funksjonelle avhengigheter er av Id, og Id er en supernøkkel.

**Foredlingsmetode**(Id, Navn, Beskrivelse)

Foredlingsmetode er på BCNF siden begge funksjonelle avhengigheter er av Id, og Id er en supernøkkel.

**Kaffebønne**(Id, Type, ArtId, GårdId)

ArtId er fremmednøkkel mot Art og kan være NULL.

GårdId er fremmednøkkel mot Gård og kan ikke være NULL.

Kaffebønne er på BCNF siden alle funksjonelle avhengigheter er av Id, og Id er en supernøkkel.

**Art**(Id, Navn)

Art er på BCNF siden eneste funksjonelle avhengighet er Id -> Navn, og Id er en supernøkkel

**PartiBestårAv**(Kaffebønneld, KaffepartId)

Kaffebønneld og KaffepartId er fremmednøkler mot henholdsvis Kaffebønne og Kaffeparti, og kan ikke være NULL

Tabellen inneholder ingen funksjonelle avhengigheter og er derfor på BCNF.

## Oppgave c

### Brukerhistorie 1

En *bruker* kan registreres og få en relasjon til en *kaffesmaking*. Denne kaffesmakingen har blant annet attributtene *poeng* og *smaksnotater*, som respektivt kan inneholde 10 poeng og brukernotatet "Wow - en odysse for smaksløkene [...]". Dette er verdier som fint kan endres da de ikke fungerer som nøkler, og kan med andre ord fint bli tatt inn som input fra brukerens side. Videre har *kaffesmaking* en relasjon til type *kaffe*, og det er gjennom *kaffe* vi oppbevarer informasjon om navn (Vinterkaffe 2022), brenningsgrad (lysbrent), samt kiloprisen på 600 NOK. Det er også viktig at *kaffe* har en relasjon til et *kaffebrenneri* (Trondheims-brenneriet Jacobsen & Svart), med en brenningsdato (20.01.2022), som vi har satt på relasjonsklassen mellom de to entitetsklassene. Selv om det er brenneriet som gir

beskrivelsen til kaffen (her “En velsmakende og kompleks kaffe for mørketiden”), er det naturlig å sette dette som et attributt til hver unike *kaffe*. Gjennom *kaffe* sin relasjon til et *kaffeparti* kan vi oppbevare info om innhøstingsåret (2021) og kiloprisen gården fikk utbetalt (8 USD). Dette fordi en *gård* vil ha flere partier, og vi kan forvente at både denne kiloprisen vil variere fra parti til parti (dette gjelder selvfølgelig også innhøstingsåret). Hvert *kaffeparti* antar vi inneholder foredlet kaffe, og krever derfor en relasjon til *foredlingsmetode*, der navn (bærtørket) kan komme. Denne gården som produserer kaffepartiet har imidlertid informasjon vi må oppbevare, og dette gjør vi gjennom en egen *gård*-klasse, som inneholder navn (Nombre de Dios) og gården sin høyde over havet (1500 moh.). Lokasjonen til gården, her regionen Santa Ana og landet El Salvador, har vi satt som egne entitetsklasser med relasjoner seg imellom, noe alle gårder vil ha en relasjon til gjennom *region*. Til slutt vil informasjonen om kaffebønne sin type (Bourbon) ligge som et type-attributt på *kaffebønne*, mens dets tilhørende art (c. arabica) kommer via relasjonen til *art*-klassen.

## Brukerhistorie 2

Dette kan gjennomføres ved å slå sammen tabell for *bruker* og *kaffesmaking*, hente ut brukerens fornavn og etternavn, og antall unike kaffesmakinger funnet. Deretter kan tabellen sorteres med dette antallet kaffesmakinger, noe som fint kan gjøres synkende.

Dette kan vi formulere som en SQL-spørring, og det ville sett ut som følgende:

```
SELECT Bruker.Fornavn, Bruker.Etternavn, COUNT(*)
FROM Bruker INNER JOIN Kaffesmaking
ON Bruker.Epost = Kaffesmaking.BrukerEpost
GROUP BY Bruker.Fornavn, Bruker.Etternavn
```

## Brukerhistorie 3

Dette kan en se ved å slå sammen tabell for *kaffesmaking* og *kaffe*, og finne gjennomsnitt poeng gitt på hver *kaffesmaking* relatert til en type *kaffe*. Vi må også slå sammen tabellen med *kaffebrenneri* for å få ut brennerinavnet.

Dette kan vi formulere som en SQL-spørring, og det ville sett ut som følgende:

```
SELECT
Kaffebrenneri.Navn AS Brennerinavn,
Kaffe.Navn AS Kaffenavn,
AVG(Kaffesmaking.Poeng) AS Gjennomsnitt
FROM Kaffe INNER JOIN Kaffesmaking
ON Kaffe.Id = Kaffesmaking.KaffeId
INNER JOIN Kaffebrenneri
ON Kaffe.KaffebrenneriId = Kaffebrenneri.Id
GROUP BY Kaffebrenneri.Navn, Kaffe.Navn
ORDER BY Gjennomsnitt DESC
```

## Brukerhistorie 4

Beskrivelsen på en kaffe gitt av en bruker ligger på *kaffesmaking* sitt Smaksnotater-attributt, og beskrivelsen gitt av et brenneri ligger på *kaffe* sitt Beskrivelse-attributt. Vi kan derfor slå sammen disse to tabellene, *kaffesmaking* og *kaffe*, og filtrere på at attributtene skal inneholde. Men, vi må også slå sammen dette med kaffebrenneri-tabellen, da vi ønsker å hente ut navnet på kaffebrenneriet, noe vi lagrer som et attributt. Navnet på *kaffe* lagres på samme måte, og kan også hentes ut.

Dette kan vi formulere som en SQL-spørring, og det ville sett ut som følgende:

```
SELECT Kaffebrenneri.Navn AS Brennerinavn, Kaffe.Navn AS Kaffenavn
FROM Kaffe INNER JOIN Kaffesmaking
ON Kaffe.Id = Kaffesmaking.KaffeId
INNER JOIN Kaffebrenneri
ON Kaffe.kaffebrenneriId = Kaffebrenneri.Id
WHERE Kaffe.Beskrivelse LIKE '%floral%'
OR Kaffesmaking.Smaksnotater LIKE '%floral%'
```

## Brukerhistorie 5

Dette vil være mulig ved at alle typer *kaffe* er fra nøyaktig ett *kaffeparti*, og alle *kaffeparti* er foredlet med nøyaktig én metode. Dermed hører det til nøyaktig én foredlingsmetode til alle typer *kaffe*. Vi kan slik søke etter typer *kaffe* med en *foredlingsmetode* med navn ulikt "Vasket". På samme måte er hver *kaffe* fra nøyaktig ett *land*, ved at de er fra nøyaktig ett *kaffeparti*, som er koblet til nøyaktig én *gård*, som er fra nøyaktig én *region*, som ligger i nøyaktig ett *land*. Slik kan vi også filtrere på at *kaffen* skal være fra et *land* med navn lik "Rwanda" eller "Colombia". For alle typer *kaffe* som oppfyller dette kravet kan vi returnere kaffenavnet ved å hente ut verdien fra navn-attributtet til *kaffe*. Hver *kaffe* er fra nøyaktig ett *kaffebrenneri*, og vi kan slik hente ut brennerinavn ved å hente ut dette *kaffebrenneriet* sitt navn.

Dette kan vi formulere som en SQL-spørring, og det ville sett ut som følgende:

```
SELECT Kaffebrenneri.Navn AS Brennerinavn, Kaffe.Navn AS Kaffenavn
FROM Kaffebrenneri INNER JOIN Kaffe
ON Kaffebrenneri.Id = Kaffe.KaffebrenneriId
INNER JOIN Kaffeparti
ON Kaffe.KaffepartiId = Kaffeparti.Id
INNER JOIN Gård
ON Kaffeparti.GårdId = Gård.Id
INNER JOIN Region
ON Gård.RegionId = Region.Id
INNER JOIN Land
ON Region.LandId = Land.Id
INNER JOIN Foredlingsmetode
ON Kaffeparti.ForedlingsmetodeId = Foredlingsmetode.Id
WHERE (Land.Navn = 'Rwanda' OR Land.Navn = 'Colombia')
AND Foredlingsmetode.Beskrivelse != 'Vaskede'
```

