

# Øving 06 - Fra tekstlig beskrivelse til SQL

---

Av: John I. Eriksen

## Oppgave a)

---

### Tabellskisse 1

---

Under følger en oversikt over entiteter og attributter som er skissert i oppgaveteksten. Antar for enkelhetsskyld at hverken kandidat eller bedrift har mer enn ett telefonnummer eller epost.

#### Kandidat:

- Fornavn
- Etternavn
- Telefon
- Epost

#### Bedrift:

- Organisasjonsnr (unik)
- Navn
- Telefon
- Epost

#### Oppdrag:

- Oppdragsnummer (unik)
- Bedriftsnavn
- Nødvendig kvalifikasjon
- Startdato (foreløpig)
- Sluttdato (foreløpig)

#### Kvalifikasjon:

- Kvalifikasjons-ID (unik)
- Beskrivelse

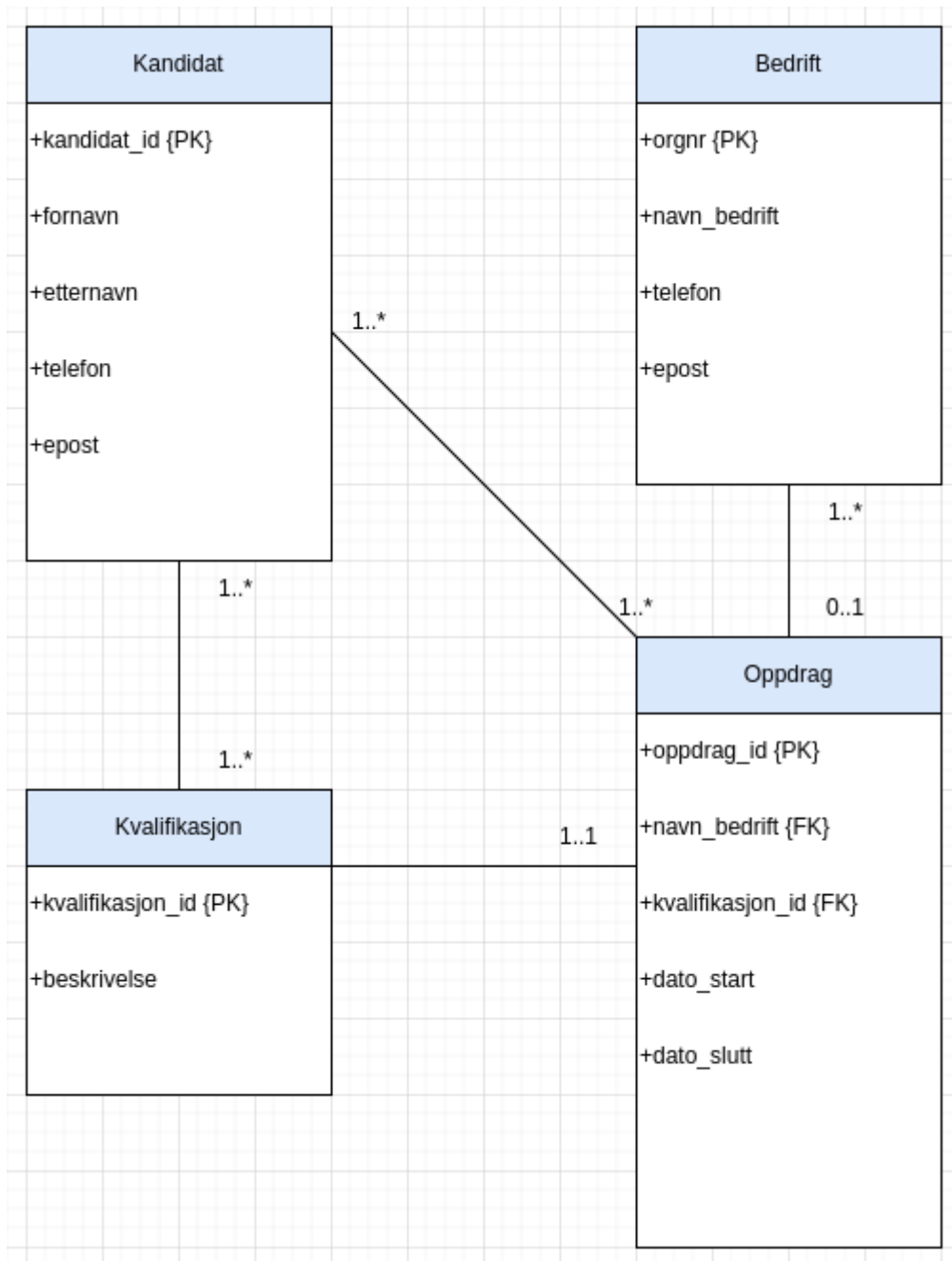
### Relasjoner:

- En **Kandidat** kan ha mange **Kvalifikasjoner**, og en **Kvalifikasjon** kan være tilknyttet mange **Kandidater** (Mange-til-mange).
- En **Bedrift** kan etterspørre mange **Oppdrag**, men hvert **Oppdrag** er tilknyttet nøyaktig én **Bedrift** (En-til-mange).
- Hvert **Oppdrag** krever nøyaktig én **Kvalifikasjon** (En-til-en).

- Et **Oppdrag** kan være utført av en **Kandidat**, og en **Kandidat** kan utføre mange **Oppdrag** (Mange-til-mange).

## UML-diagrammer

Diagram 1 - Eksplisitte tabeller fra oppgaven



Dette diagrammet inneholde de fire "eksplisitte" tabellene man kan hente direkte fra å lese oppgaven.

## Tabellskisse 2

Utvider databasen med følgende tabeller:

### KandidatKvalifikasjon

- kandidat\_id
- kvalifikasjon\_id

## KandidatOppdrag

- kandidat\_id
- oppdrag\_id
- dato\_start\_reell
- dato\_slutt\_reell
- timer\_antall

## Diagram 2 - Utvidet, basert på oppgave



Dette diagrammet inneholder flere tabeller for å gjøre databasen bedre å jobbe med.

## Oppgave b)

Fra oppgaven leser man enkelt ut at det trengs minimum tabeller for Kandidat, Kvalifikasjon, Bedrift og Oppdrag. Gitt relasjonene skissert ovenfor, trenger vi tabeller som registrerer det faktiske oppdraget fra en kandidats perspektiv og utførelse, i tillegg til en tabell med en kandidats kvalifikasjoner.

Forslag til tabeller.

**Alternativ 1:** Primærnøkkel markert med [PK] og fremmednøkkel markert med stjerne\* og [FK]:

```
1 Kandidat(kandidat_id [PK], fornavn, etternavn, telefon, epost)
2
3 Bedrift(org_nr [PK], navn_bedrift, telefon, epost)
4
5 Oppdrag(oppdrag_id [PK], navn_bedrift***** [FK], kvalifikasjon_id***** [FK],
   dato_start, dato_slutt)
6
7 Kvalifikasjon(kvalifikasjon_id [PK], beskrivelse)
8
9 KandidatKvalifikasjon(kandidat_id***** [FK], kvalifikasjon_id***** [FK])
10
11 KandidatOppdrag(kandidat_id* [FK], oppdrag_id* [FK], dato_start_reell,
   dato_slutt_reell, timer_antall)
```

**Alternativ 2:** Primærnøkkel med strek under, og fremmednøkkel med stjerne, i stedet for i kodeblokk:

**Kandidat**(kandidat\_id, fornavn, etternavn, telefon, epost)

**Bedrift**(org\_nr, navn\_bedrift, telefon, epost)

**Oppdrag**(oppdrag\_id, org\_nr\*, kvalifikasjon\_id\*, dato\_start, dato\_slutt)

**Kvalifikasjon**(kvalifikasjon\_id, beskrivelse)

**KandidatKvalifikasjon**(kandidat\_id\*, kvalifikasjon\_id\*)

**KandidatOppdrag**(kandidat\_id\*, oppdrag\_id\*, dato\_start\_reell, dato\_slutt\_reell, timer\_antall)

Er det rimelig at noen av fremmednøkklene kan være NULL? Hva betyr det i tilfelle?

Gitt forretningslogikken, vil det ikke være rimelig å ha fremmednøkler som er `NULL`. Det vil i så fall innebære at det ikke finnes tilknytninger mellom entitetene, hvilket gjør at logikken ikke går opp.

## Oppgave c)

Oppretter tabellene, som satt opp i oppgave b).

```
1 CREATE TABLE Kandidat (
2   kandidat_id INT AUTO_INCREMENT PRIMARY KEY,
3   fornavn VARCHAR(50),
4   etternavn VARCHAR(50),
5   telefon VARCHAR(8),
6   epost VARCHAR(100) UNIQUE,
7   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
8   updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
9 );
10
```

```

11 CREATE TABLE Bedrift (
12     org_nr INT AUTO_INCREMENT PRIMARY KEY,
13     navn VARCHAR(80),
14     telefon VARCHAR(8),
15     epost VARCHAR(100) UNIQUE,
16     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
17     updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
18 );
19
20 CREATE TABLE Kvalifikasjon (
21     kvalifikasjon_id INT AUTO_INCREMENT PRIMARY KEY,
22     beskrivelse VARCHAR(255),
23     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
24     updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
25 );
26
27 CREATE TABLE Oppdrag (
28     oppdrag_id INT AUTO_INCREMENT PRIMARY KEY,
29     org_nr INT,
30     kvalifikasjon_id INT,
31     dato_start DATE,
32     dato_slutt DATE,
33     FOREIGN KEY (org_nr) REFERENCES Bedrift(org_nr),
34     FOREIGN KEY (kvalifikasjon_id) REFERENCES Kvalifikasjon(kvalifikasjon_id),
35     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
36     updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
37 );
38
39 CREATE TABLE KandidatKvalifikasjon (
40     kandidat_id INT,
41     kvalifikasjon_id INT,
42     PRIMARY KEY (kandidat_id, kvalifikasjon_id),
43     FOREIGN KEY (kandidat_id) REFERENCES Kandidat(kandidat_id),
44     FOREIGN KEY (kvalifikasjon_id) REFERENCES Kvalifikasjon(kvalifikasjon_id),
45     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
46     updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
47 );
48
49 CREATE TABLE KandidatOppdrag (
50     kandidat_id INT,
51     oppdrag_id INT,
52     dato_start_reell DATE,
53     dato_slutt_reell DATE,
54     timer_antall INT,
55     PRIMARY KEY (kandidat_id, oppdrag_id),
56     FOREIGN KEY (kandidat_id) REFERENCES Kandidat(kandidat_id),
57     FOREIGN KEY (oppdrag_id) REFERENCES Oppdrag(oppdrag_id),
58     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
59     updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
60 );

```

Legger til eksempeldata.

```
1  -- Legger til 10 Kandidater
2  INSERT INTO Kandidat (fornavn, etternavn, telefon, epost)
3  VALUES
4      ('Kari-Alice', 'Smaldal', 12345678, 'kari-alice@email.com'),
5      ('Ola-Bob', 'Bratthei', 98765432, 'ola-bob@email.com'),
6      ('Kjell Pell', 'Potetskrell', 19283746, 'kjellpell@email.com'),
7      ('Per', 'Parker', 65748392, 'per.parker@email.com'),
8      ('Anne', 'Droid', 34567890, 'anne.droid@email.com'),
9      ('Ruth', 'Less', 54321987, 'ruth.less@email.com'),
10     ('Ada', 'Line', 13467982, 'ada.line@email.com'),
11     ('Bob', 'Builder', 24681357, 'bob.builder@email.com'),
12     ('Ellen', 'Generics', 86429731, 'ellen.dg@email.com'),
13     ('John', 'Doe', 11122334, 'john.doe@email.com');
14
15  INSERT INTO Kvalifikasjon (beskrivelse)
16  VALUES
17      ('Full-Stack Developer'),
18      ('Half-Stack Developer'),
19      ('Seaman'),
20      ('Healer');
21
22  -- Legger til 5 Bedrifter
23  INSERT INTO Bedrift (org_nr, navn, telefon, epost)
24  VALUES
25      (1, 'EEE Solutions Inc.', 66006600, 'eee.solutions@email.com'),
26      (2, 'Asparagus Kalkulatorix', 55009090, 'asparagus.kx@email.com'),
27      (3, 'Bananarama', 44008800, 'bananarama@email.com'),
28      (4, 'Cucumber Studios', 33007700, 'cucumber.studios@email.com'),
29      (5, 'Daring Dynamics', 22004400, 'daring.dyn@email.com');
30
31  -- Legger til flere Oppdrag
32  INSERT INTO Oppdrag (org_nr, kvalifikasjon_id, dato_start, dato_slutt)
33  VALUES
34      (1, 1, DATE('2023-01-24'), DATE('2023-02-02')),
35      (2, 2, DATE('2023-02-14'), DATE('2023-04-12')),
36      (3, 3, DATE('2023-03-01'), DATE('2023-05-10')),
37      (4, 4, DATE('2023-04-01'), DATE('2023-07-10')),
38      (5, 1, DATE('2023-05-11'), DATE('2023-06-15')),
39      (3, 2, DATE('2023-07-01'), DATE('2023-08-15')),
40      (4, 3, DATE('2023-08-10'), DATE('2023-10-20')),
41      (1, 4, DATE('2023-10-25'), DATE('2023-12-31'));
42
43  INSERT INTO KandidatKvalifikasjon (kandidat_id, kvalifikasjon_id)
44  VALUES
45      (1, 1),
46      (2, 2),
47      (3, 3),
48      (4, 4),
```

```

49      (5, 1),
50      (6, 2),
51      (7, 3);
52
53  -- Tildeler flere oppdrag til kandidater
54  INSERT INTO KandidatOppdrag (kandidat_id, oppdrag_id, dato_start_reell,
55      dato_slutt_reell, timer_antall)
56  VALUES
57      (1, 1, DATE('2023-01-25'), DATE('2023-02-01'), 50),
58      (2, 2, DATE('2023-02-15'), DATE('2023-04-11'), 80),
59      (3, 3, DATE('2023-03-02'), DATE('2023-05-09'), 100),
60      (4, 4, DATE('2023-04-02'), DATE('2023-07-09'), 120),
61      (5, 5, DATE('2023-05-12'), DATE('2023-06-14'), 90),
62      (6, 6, DATE('2023-07-02'), DATE('2023-08-14'), 70),
63      (7, 7, DATE('2023-08-11'), DATE('2023-10-19'), 60),
64      (8, 8, DATE('2023-10-26'), DATE('2023-12-30'), 100),
65      (9, 1, DATE('2023-01-26'), DATE('2023-02-02'), 40),
66      (1, 2, DATE('2023-02-16'), DATE('2023-04-12'), 90);

```

Sånn.

## Oppgave d)

Sett opp `SELECT`-setninger.

### d1

Liste over alle bedriftene, med navn, telefon og e-post:

```

1 | SELECT navn, telefon, epost FROM Bedrift;

```

Output:

navn	telefon	epost
EEE Solutions Inc.	66006600	<a href="mailto:eee.solutions@email.com">eee.solutions@email.com</a>
Asparagus Kalkulatorix	55009090	<a href="mailto:asparagus.kx@email.com">asparagus.kx@email.com</a>
Bananarama	44008800	<a href="mailto:bananarama@email.com">bananarama@email.com</a>
Cucumber Studios	33007700	<a href="mailto:cucumber.studios@email.com">cucumber.studios@email.com</a>
Daring Dynamics	22004400	<a href="mailto:daring.dyn@email.com">daring.dyn@email.com</a>

## d2

Liste over alle oppdrag, med oppdragsnummer, bedriftsnavn og telefonnummer:

```
1 SELECT
2   Oppdrag.oppdrag_id,
3   Bedrift.navn,
4   Bedrift.telefon
5 FROM Oppdrag
6   JOIN Bedrift ON Oppdrag.org_nr = Bedrift.org_nr;
```

Output:

oppdrag_id	navn	telefon
1	EEE Solutions Inc.	66006600
8	EEE Solutions Inc.	66006600
2	Asparagus Kalkulatorix	55009090
3	Bananarama	44008800
6	Bananarama	44008800
4	Cucumber Studios	33007700
7	Cucumber Studios	33007700
5	Daring Dynamics	22004400

## d3

Liste over kandidater og kvalifikasjoner:

```
1 SELECT
2   Kandidat.kandidat_id,
3   CONCAT(Kandidat.fornavn, ' ', Kandidat.etternavn) AS Kandidatnavn,
4   Kvalifikasjon.kvalifikasjon_id,
5   Kvalifikasjon.beskrivelse
6 FROM KandidatKvalifikasjon
7   JOIN Kandidat ON KandidatKvalifikasjon.kandidat_id = Kandidat.kandidat_id
8   JOIN Kvalifikasjon ON KandidatKvalifikasjon.kvalifikasjon_id =
   Kvalifikasjon.kvalifikasjon_id;
```

Output:



kandidat_id	Kandidatnavn	kvalifikasjon_id	beskrivelse
1	Kari-Alice Smaldal	1	Full-Stack Developer
5	Anne Droid	1	Full-Stack Developer
2	Ola-Bob Bratthei	2	Half-Stack Developer
6	Ruth Less	2	Half-Stack Developer
3	Kjell Pell Potetskrell	3	Seaman
7	Ada Line	3	Seaman
4	Per Parker	4	Healer

## d4

Som forrige, men inkludert kandidater *uten* kvalifikasjoner ved å bruke `LEFT JOIN`:

```

1 SELECT
2   Kandidat.kandidat_id,
3   CONCAT(Kandidat.fornavn, ' ', Kandidat.etternavn) AS Kandidatnavn,
4   Kvalifikasjon.kvalifikasjon_id,
5   Kvalifikasjon.beskrivelse
6 FROM Kandidat
7   LEFT JOIN KandidatKvalifikasjon ON Kandidat.kandidat_id =
  KandidatKvalifikasjon.kandidat_id
8   LEFT JOIN Kvalifikasjon ON KandidatKvalifikasjon.kvalifikasjon_id =
  Kvalifikasjon.kvalifikasjon_id;
```

Output:

kandidat_id	Kandidatnavn	kvalifikasjon_id	beskrivelse
1	Kari-Alice Smaldal	1	Full-Stack Developer
2	Ola-Bob Bratthei	2	Half-Stack Developer
3	Kjell Pell Potetskrell	3	Seaman
4	Per Parker	4	Healer
5	Anne Droid	1	Full-Stack Developer
6	Ruth Less	2	Half-Stack Developer
7	Ada Line	3	Seaman
8	Bob Builder	null	null
9	Ellen DeGeneres	null	null

kandidat_id	Kandidatnavn	kvalifikasjon_id	beskrivelse
10	John Doe	null	null

## d5

Jobbhistorikken til en bestemt vikar, gitt kandidatnummer:

```

1  SELECT CONCAT
2    (Kandidat.fornavn, ' ', Kandidat.etternavn) AS Vikarnavn,
3    KandidatOppdrag.dato_slutt_reell AS Sluttdato,
4    KandidatOppdrag.oppdrag_id,
5    Bedrift.navn AS Bedriftsnavn,
6    Kvalifikasjon.beskrivelse AS Kvalifikasjon
7  FROM KandidatOppdrag
8    JOIN Kandidat ON KandidatOppdrag.kandidat_id = Kandidat.kandidat_id
9    JOIN Oppdrag ON KandidatOppdrag.oppdrag_id = Oppdrag.oppdrag_id
10   JOIN KandidatKvalifikasjon ON KandidatKvalifikasjon.kandidat_id =
Kandidat.kandidat_id
11   JOIN Kvalifikasjon ON KandidatKvalifikasjon.kvalifikasjon_id =
Kvalifikasjon.kvalifikasjon_id
12   JOIN Bedrift ON Oppdrag.org_nr = Bedrift.org_nr
13   WHERE Kandidat.kandidat_id = GITT_KANDIDAT_ID;
```

Her erstattes `GITT_KANDIDAT_ID` med den kandidaten man ønsker å se jobbhistorikken til.

Output, med `GITT_KANDIDAT_ID` satt til `1`:

Vikarnavn	Sluttdato	oppdrag_id	Bedriftsnavn	Kvalifikasjon
Kari-Alice Smaldal	2023-02-01	1	EEE Solutions Inc.	Full-Stack Developer
Kari-Alice Smaldal	2023-04-12	2	Asparagus Kalkulatorix	Full-Stack Developer