

# Intro til databaserelasjoner og INNER JOIN

- **En-til-mange (1-M):** Én rad i tabell A kan ha mange rader i tabell B. Fremmednøkkelen ligger i tabell B.
- **Mange-til-en (M-1):** Samme relasjon sett fra "mange"-siden: mange rader i B peker til én rad i A.
- **INNER JOIN:** viser bare rader som **matcher** i begge tabeller.

Eksempel på mønster (uten alias):

```
SELECT kolonner
FROM tabell2
INNER JOIN tabell1 ON tabell2.fk = tabell1.pk
WHERE ...
ORDER BY ... ASC;
```

## Database 1 : Klasse og elev (2 tabeller)

### Beskrivelse (to tabeller)

- **klasse** (1) → (M) **elev** Én klasse kan ha mange elever. Mange elever tilhører én klasse.

Kolonner (du kan bruke nøyaktig disse)

- **klasse**
  - **klasse\_id** (PK)
  - **navn** (f.eks. "2ITB")
- **elev**
  - **elev\_id** (PK)
  - **fornavn**
  - **etternavn**
  - **klasse\_id** (FK til **klasse.klasse\_id**)

Vist eksempel: lag tabeller

```
CREATE TABLE klasse (
    klasse_id SERIAL PRIMARY KEY,
    navn TEXT NOT NULL
);

CREATE TABLE elev (
    elev_id SERIAL PRIMARY KEY,
    fornavn TEXT NOT NULL,
    etternavn TEXT NOT NULL,
```

```

klasse_id INT NOT NULL,
FOREIGN KEY (klasse_id) REFERENCES klasse (klasse_id)
);

```

Vist eksempel: enkel INNER JOIN

```

SELECT
    elev.fornavn,
    elev.etternavn,
    klasse.navn
FROM elev
INNER JOIN klasse ON elev.klasse_id = klasse.klasse_id
ORDER BY klasse.navn ASC, elev.etternavn ASC;

```

## Oppgaver (10) – Klasse/elev

1. Lag tabellen **klasse** med **klasse\_id** som primærnøkkel og **navn** som **NOT NULL**.
2. Lag tabellen **elev** med primærnøkkel og **klasse\_id** som fremmednøkkel til **klasse**.
3. Sett inn minst 3 klasser i **klasse**.
4. Sett inn minst 10 elever i **elev**, fordelt på klassene.
5. Skriv en spørring som viser alle elever med klassenavn (INNER JOIN).
6. Skriv en spørring som viser bare elever i klassen "2ITB" (INNER JOIN + WHERE).
7. Skriv en spørring som finner elever der **fornavn** = 'Ali' OR **fornavn** = 'Sara' (INNER JOIN + WHERE OR).
8. Skriv en spørring som finner elever som **ikke** har etternavn "Hansen" (INNER JOIN + WHERE med **<>**).
9. Skriv en spørring som viser elever i "2ITA" **og** med etternavn "Nilsen" (INNER JOIN + WHERE AND).
10. Skriv en spørring som viser alle elever sortert på **etternavn** stigende, og deretter **fornavn** synkende (ORDER BY ASC/DESC).

## Database 2: Nettbutikk

### Relasjoner

- **kunde** (1) → (M) **ordre**
- **ordre** (M) → (1) **produkt** (altså mange ordrelinjer peker til ett produkt – men her bruker vi "enkle ordre" der én ordre peker til ett produkt for å holde det lett)

Dette er ikke "perfekt" nettbutikk-modell (normalt har man ordrelinjer), men den er super for intro: flere tabeller, tydelige FK-er og enkle INNER JOIN.

### Kolonner (bruk disse)

- **kunde**
  - **kunde\_id** (PK)

- navn
- epost (unik)
- produkt
  - produkt\_id (PK)
  - navn
  - pris (NUMERIC)
- ordre
  - ordre\_id (PK)
  - kunde\_id (FK)
  - produkt\_id (FK)
  - antall
  - ordre\_dato (DATE)

Vist eksempel: lag tabeller

```

CREATE TABLE kunde (
  kunde_id SERIAL PRIMARY KEY,
  navn TEXT NOT NULL,
  epost TEXT NOT NULL UNIQUE
);

CREATE TABLE produkt (
  produkt_id SERIAL PRIMARY KEY,
  navn TEXT NOT NULL,
  pris NUMERIC(10,2) NOT NULL
);

CREATE TABLE ordre (
  ordre_id SERIAL PRIMARY KEY,
  kunde_id INT NOT NULL,
  produkt_id INT NOT NULL,
  antall INT NOT NULL,
  ordre_dato DATE NOT NULL,
  FOREIGN KEY (kunde_id) REFERENCES kunde (kunde_id),
  FOREIGN KEY (produkt_id) REFERENCES produkt (produkt_id)
);

```

Vist eksempel: INNER JOIN over 3 tabeller

```

SELECT
  kunde.navn,
  kunde.epost,
  produkt.navn,
  produkt.pris,
  ordre.antall,

```

```

ordre.ordre_dato
FROM ordre
INNER JOIN kunde ON ordre.kunde_id = kunde.kunde_id
INNER JOIN produkt ON ordre.produkt_id = produkt.produkt_id
ORDER BY ordre.ordre_dato DESC;

```

## Oppgaver (10) – Nettbutikk

1. Lag tabellene **kunde**, **produkt**, **ordre** med riktige primærnøkler og fremmednøkler.
2. Sett inn minst 5 kunder og 6 produkter.
3. Sett inn minst 12 ordre (flere ordre per kunde).
4. Skriv en spørring som viser alle ordre med kundenavn og produktnavn (INNER JOIN).
5. Skriv en spørring som viser ordre for kunden med **epost = 'test@example.com'** (INNER JOIN + WHERE).
6. Skriv en spørring som viser ordre der **ordre.antall >= 2** (INNER JOIN + WHERE med **>=**).
7. Skriv en spørring som viser ordre der produktpris er **> 500** (INNER JOIN + WHERE).
8. Skriv en spørring som viser ordre der kundenavn er "**Ola Nordmann**" **eller** produktnavn er "**Mus**" (WHERE OR).
9. Skriv en spørring som viser ordre der produktpris er mellom 200 og 800 (logisk uttrykk med AND: **pris >= 200 AND pris <= 800**).
10. Skriv en spørring som viser ordre sortert på **kunde.navn ASC** og **ordre.ordre\_dato DESC**.

## Database 3: Hotell (Del 3 )

Dere må selv lage tabeller med PK/FK riktig, sette inn testdata, og skrive spørringer med **INNER JOIN + WHERE (AND/OR) + ORDER BY (ASC/DESC)**.

### Krav / beskrivelse

Dere skal lage 3 tabeller:

#### Tabell 1: **gjest**

- **gjest\_id** (PK)
- **fornavn** (NOT NULL)
- **etternavn** (NOT NULL)
- **telefon** (kan være NULL)

#### Tabell 2: **rom**

- **rom\_id** (PK)
- **romnummer** (UNIQUE, NOT NULL)
- **type** (NOT NULL) (f.eks. "enkelt", "dobbelt")
- **pris\_per\_natt** (NOT NULL)

#### Tabell 3: **booking**

- `booking_id` (PK)
- `gjest_id` (FK til `gjest.gjest_id`, NOT NULL)
- `rom_id` (FK til `rom.rom_id`, NOT NULL)
- `fra_dato` (DATE, NOT NULL)
- `til_dato` (DATE, NOT NULL)

## Relasjoner

- `gjest` (1) → (M) `booking`
- `rom` (1) → (M) `booking`
- `booking` er "mange"-tabellen som har begge fremmednøklene.

## Oppgaver (10) – Hotell (del 3)

1. Skriv `CREATE TABLE` for `gjest` med riktige datatyper og constraints (NOT NULL der det står).
2. Skriv `CREATE TABLE` for `rom`, og sorg for at `romnummer` er unikt.
3. Skriv `CREATE TABLE` for `booking` med to fremmednøkler (til `gjest` og `rom`) og NOT NULL på FK-ene.
4. Sett inn minst 6 gjester.
5. Sett inn minst 5 rom med ulike typer og priser.
6. Sett inn minst 10 bookinger. En gjest skal ha minst 2 bookinger, og ett rom skal være booket flere ganger.
7. Skriv en spørring som viser alle bookinger med: gjestens fornavn, etternavn, romnummer, `fra_dato`, `til_dato` (INNER JOIN).
8. Skriv en spørring som viser bookinger for romnummer `101` (INNER JOIN + WHERE).
9. Skriv en spørring som viser bookinger der romtype er "dobbelt" og pris\_per\_natt er >= `1200` (INNER JOIN + WHERE AND).
10. Skriv en spørring som viser bookinger der gjestens etternavn er "Hansen" eller romtype er "enkelt", sortert på `fra_dato` ASC (INNER JOIN + WHERE OR + ORDER BY).