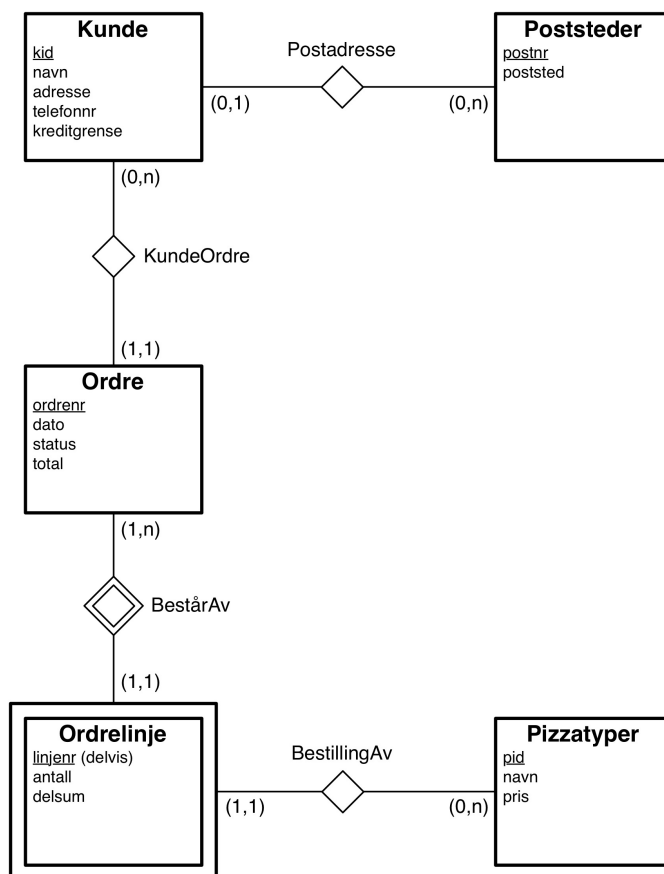


TDT4145 Datamodellering og databasesystemer

SQL-oppgave 1

Miniverden og ER-modell

Vi tar utgangspunkt i en enkel modell for en pizza-restaurant, der følgende ER-diagram beskriver databasen:



Relasjonsdatabase-skjema

En tilsvarende modell i relasjonsdatabase-utgave:

Kunde(kid, navn, adresse, postnr, telefonnr, kreditgrense)

-- der postnr er fremmednøkkel mot Poststeder-tabellen.

Poststeder(postnr, poststed)

Pizzatyper(pid, navn, pris)

Ordre(ordrenr, dato, status, total, kunde)

-- der kunde er fremmednøkkel mot Kunde-tabellen.

Ordrelinje(ordrenr, linjenr, pizza, antall, delsum)

-- der ordrenr er fremmednøkkel mot Ordre-tabellen, og
-- der pizza er fremmednøkkel mot Pizzatyper-tabellen.

SQL-versjon

Et SQL-skript for å opprette en kjørbart relasjonsdatabase:

```
-- oppretter tabeller
create table poststeder (
    postnr      char(4),
    poststed   varchar(30) not null,
    constraint poststeder_pk primary key (postnr) );

create table kunde (
    kid         integer,
    navn        varchar(30) not null,
    adresse     varchar(30),
    postnr      char(4),
    tlfnr       char(8) not null,
    kredittgrense integer,
    constraint kunde_pk primary key (kid),
    constraint kunde_fk1 foreign key (postnr)
        references poststeder(postnr)
        on update cascade
        on delete set null );

create table pizzatyper (
    pid        integer,
    navn       varchar(30) not null,
    pris       integer not null,
    constraint pizza_pk primary key (pid) );

create table ordre (
    ordrenr integer,
    dato     date,
    status   varchar(10),
    total    integer,
    kunde    integer,
    constraint ordre_pk primary key (ordrenr),
    constraint ordre_fk1 foreign key (kunde)
        references kunde(kid)
        on update cascade
        on delete cascade);

create table ordrelinje (
    ordrenr integer,
    linjenr integer,
    pizza   integer not null,
    antall  integer not null,
    delsum  integer not null,
    constraint ordrelinje_pk primary key (ordrenr, linjenr),
    constraint ordrelinje_fk1 foreign key (ordrenr)
        references ordre(ordrenr)
        on update cascade
        on delete cascade,
    constraint ordrelinje_fk2 foreign key (pizzanr)
        references pizzatyper(pid)
        on update cascade
        on delete restrict );
```

Innlegging av data

SQL-kode for å legge inn data i databasen:

```
-- legger inn data
-- poststeder
insert into poststeder values (7000, 'Trondheim');
insert into poststeder values (8000, 'Bodø');
insert into poststeder values (9000, 'Tromsø');
insert into poststeder values (9090, 'Kirkenes');
insert into poststeder values (7046, 'Trondheim');

-- kunder
insert into kunde
values (1, 'Ola', 'Kirkegata 1', 7000, 12341234, 10000);

insert into kunde
values (2, 'Kari', 'Elgesetergate 12', 7000, 12344321, 10000);

insert into kunde
values (3, 'Per', 'Munkegata 10', 7000, 11223344, 5000);

insert into kunde
values (4, 'Ola', 'Kongens gate 12', 8000, 88667755, 5000);

-- pizzaer
insert into pizzatyper values (1, 'Cheese & Tomato', 170);
insert into pizzatyper values (2, 'The Tropical', 228);
insert into pizzatyper values (3, 'Pepper Steak', 228);
insert into pizzatyper values (4, 'Thai Chicken', 254);
insert into pizzatyper values (5, 'Heavy Heaven', 258);

-- ordrer
insert into ordre values (1, '2014-01-10', 'Levert', 170, 1);
insert into ordre values (2, '2014-01-10', 'Levert', 568, 2);
insert into ordre values (3, '2014-01-11', 'Levert', 710, 3);
insert into ordre values (4, '2014-01-11', 'Levert', 254, 3);
insert into ordre values (5, '2014-01-12', 'Levert', 482, 2);

-- ordrelinje
insert into ordrelinje values(1, 1, 1, 1, 170);
insert into ordrelinje values(2, 1, 1, 2, 340);
insert into ordrelinje values(2, 2, 2, 1, 228);
insert into ordrelinje values(3, 1, 2, 1, 228);
insert into ordrelinje values(3, 2, 3, 1, 228);
insert into ordrelinje values(3, 3, 4, 1, 254);
insert into ordrelinje values(4, 1, 4, 1, 254);
insert into ordrelinje values(5, 1, 4, 1, 254);
insert into ordrelinje values(5, 2, 3, 1, 228);
```

I filen **pizzaDB.sql** finner du SQL-koden som oppretter databasen og legger inn data.

Spørringer

Spørring 1

Finn kunder på postnummer 7000. Ta med navn og adresse.

Spørring 2

Hvor mange kunder er det i databasen? I resultatet skal kolonnen ha navnet "Antall kunder".

Spørring 3

Finn alle poststeder. Hver poststed skal være med bare en gang, og resultatet skal sorteres alfabetisk.

Spørring 4

Skriv ut kid, navn, adresse, postnummer og poststed for alle kunder.

Spørring 5

Hvor mange "Thai Chicken" er solgt og for hvor mange kroner til sammen?

Spørring 6

Finn alle pizzatyper som noen har kjøpt. Unngår å gjenta pizzanavn i resultatet og sorter resultatet alfabetisk.

Spørring 7

Finn kid, navn og adresse for alle personer som bor på adresser som inneholder "gata".

Spørring 8

Finn alle pizzatyper som kunden "Kari" ikke har kjøpt ennå.

Spørring 9

Finn alle navn som enten er brukt på en kunde eller på en pizzatype.

Spørring 10

Ta ut postnr, poststed og antall personer som bor på et postnummer. I resultatet skal alle postnumre være med, og det skal sorteres etter synkende antall kunder.

Spørring 11

Finn alle personer (kid og navn) som *ikke* har kjøpt "Thai Chicken".

Innsetting, oppdatering og sletting

Oppgave 1

Kunde nummer 2, Kari , bestiller i dag (30. januar 2014) 1 "Pepper Steak" og 3 "Thai chicken". Legg inn denne bestillingen.

Oppgave 2

Øk prisen på alle pizzatyper med 10 %.

Oppgave 3

Prøv å slette pizzatypen "Thai Chicken". Hva skjer? Hvorfor? Hva må du evt. gjøre for å få slettet denne pizzatypen?

Roger Midtstraum, 30.12.2014