



## Øving 2

TDAT1005 Databaser med videregående programmering  
Institutt for datateknologi og informatikk (IDI), NTNU

---

Tema for denne oppgaven er tabeller av strenger og samarbeid mellom objekter.

Du skal lage et enkelt system for reservasjon av bord på en restaurant. Alle bordene er like store, og et bord er enten reservert eller ikke reservert. Dersom bordet er reservert lagrer vi navnet på den personen som har reservert bordet. Vi ser bort fra tidspunkter for reservasjon.

### Objektorientert design

Du skal lage to klasser, Bord og Restaurant.

Et objekt av klassen Restaurant skal ha kunnskap knyttet til restauranten, i dette tilfellet nøyer vi oss med navn og etableringsår. Objektet skal kunne svare på følgende henvendelser:

- Finn navnet på restauranten
- Sett et nytt navn på restauranten
- Finn etableringsåret
- Finn hvor gammel restauranten er
- Finn antall ledige bord
- Finn antall bord opptatt
- Reserver et bestemt antall bord på En person (et navn)
- Finn hvilke bord en bestemt person har reservert, bordnumrene skal returnes som `int[]`
- Frigi bord som er ryddet, bordnumrene skal sendes inn som `int[]`

Det vil si at Restaurant-objektet også skal kunne svare på henvendelser ang reservasjon av bord. Et objekt av klassen Bord inneholder informasjon om *alle* bordene i restauranten. ***Et restaurantobjekt må altså samarbeide med et tilhørende bordobjekt for å finne informasjon knyttet til bordene i restauranten.*** Her er dermed aktuelle operasjoner knyttet til Bord-objektet (det kan være flere operasjoner, det kommer an på hvordan du løser oppgaven):

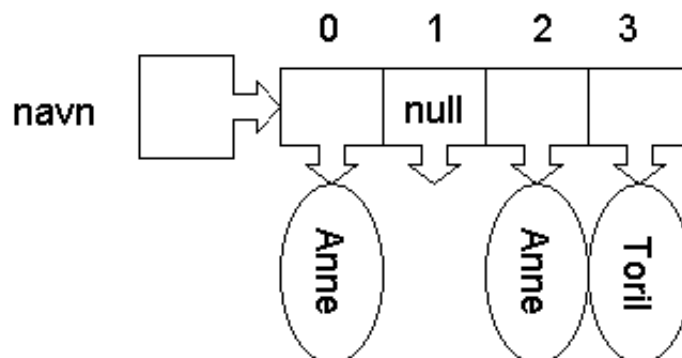
- Finn antall ledige bord
- Finn antall bord opptatt
- Frigi bord som er ryddet
- Reserver bord

## Litt om implementasjon

Klassen Restaurant skal, i tillegg til navn og etableringsår, ha et objekt av klassen Bord som objektvariabel. Det siste er nødvendig for at samarbeidet beskrevet foran skal fungere på mest mulig elegante måte.

Klassen Bord skal ha en tabell av navn som eneste objektvariabel. Lengden på denne tabellen skal være lik antall bord i restauranten. Et bordnummer svarer til et indeksnummer i tabellen, det vil si at 0 er laveste bordnummer. Dersom et bord er opptatt peker det tilsvarende tabellelementet til en streng som inneholder navnet på den som har reservert bordet. Dersom et bord er ledig er verdien til det tilsvarende tabellelementet lik null.

### Eksempel



Her har vi fire bord. Bord nr 0 og 2 er reservert av Anne, mens bord nr 3 er reservert av Toril. Bord nr 1 er ikke reservert for øyeblikket.

## Oppgave

Programmer klassene. Du må gjerne lage flere metoder enn det som er foreslått foran.

Merk at det her ikke er noe krav til at samme person ikke kan være registrert flere ganger. At samme navn er registrert flere ganger, betyr bare at denne personen har reservert flere bord.

Lag et enkelt testprogram.

## Frivillig tillegg:

Lag et menystyrt klientprogram etter følgende mal:

```
spør brukeren om restaurantnavn, etableringsår, og antall bord.  
opprett et Restaurant-objekt  
do {      valg = vis liste over mulige valg  
    switch (valg) {  
        case reserverBord: // reservere et antall  
bord på et bestemt navn  
            ....les inn navn og antall bord, og kall metode...  
        case finnBordnr: // finne alle bordene som er  
reservert på et bestemt navn  
            ....les inn navn, og kall metode...  
        case frigiBord: // frigi en rekke bord,  
bordnummer er gitt  
            ....les inn aktuelle bordnummer og kall metode...  
        case avslutt:  
            ....  
    } while (valg != avslutt)
```

Brukeren skal kunne velge mellom følgende alternativer:

- reserver bord (les inn navn og antall bord)
- finn hvilke bord en bestemt person har reservert (les inn navn)
- frigi bord (les inn aktuelle bordnummer)
- avslutt

Sørg også for at metodene som finner alderen og endrer restaurantnavnet blir prøvd ut.

Lykke til!

---

Institutt for datateknologi og informatikk (IDI), NTNU