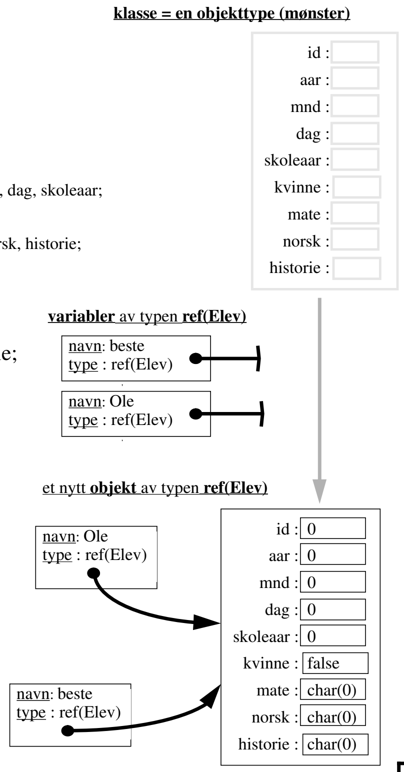
Klasser, objekter, referanse-variabler




```
begin

  class Elev;
   begin
     integer id, aar, mnd, dag, skoleaar;
     boolean kvinne;
     character mate, norsk, historie;
   end ;

  ref(Elev) beste, Ole;

  Ole :- new Elev ;

  beste :- Ole ;
```


Skole-program

```
begin
   class Elev;
     begin
       integer id, aar, mnd, dag, skoleaar;
       boolean mann;
       character mate, norsk, historie;
     end;

   integer antall, maks=1500;
   ref(Elev) array Skole(1:maks);
   character ordre;

   ordre := be_om_char("Ordre > ");
   while ordre <> 'q' do
     begin
       if ordre='?' then Hjelp
       else if ordre='n' then Ny_Elev
       else if ordre='v' then Vis_Elev
       else if ordre='l' then Vis_Liste
       else if ordre='e' then Endre_Elev
       else if ordre='f' then Fjern_Elev
       else if ordre='s' then Skriv_til_Fil
       else if ordre='h' then Hent_til_Fil
       else if ordre='d' then Daarlige_karakterer
       else skriv("Dette var en ulovlig ordre");
       ordre := be_om_char("Neste ordre > ")
     end;
end
```

```
character procedure be_om_char(prompt);
                                    text prompt;
begin
   outtext(prompt); outimage;
   inimage; be_om_char :=inchar;
end;
```


Ny_Elev

```
procedure Ny_Elev;
 if antall >= maks then skriv("Ikke mer plass")
 else
  begin
   ref(Elev) ny;
   ny :- new Elev;
  inspect ny do
   begin
     ny . id := be_om_int("ID nummer: ");
     ny . aar := be_om_int("Fodselsaar: ");
     ny . mnd := be_om_int("Fodselsmaaned: ");
     ny . dag := be_om_int("Fodselsdag: ");
     ny . skoleaar := be_om_int("Skoleaar: ");
     ny . mann := be_om_bool("Mann ?: ");
     ny . mate := be_om_char("Karakter i mate: ");
     ny . mate := be_om_char("Karakter i norsk:');
     ny . mate := be_om_char("Karakter i historie:');
  end inspect;
   antall := antall+1;
   Skole(antall) :- ny;
  end;
```

```
class Elev;
begin
 integer id, aar, mnd, dag, skoleaar;
 boolean mann;
 character mate, norsk, historie;

 procedure les;
 begin
  id:=be_om_int("ID nummer:");
  aar:=be_om_int("Fodselsaar:");
  mnd:=be_om_int("Fodselsmaaned:");
  dag:=be_om_int("Fodselsdag:");

 end;
end;
```

```
procedure Ny_Elev;
 if antall >= maks then skriv("Ikke mer plass")
 else
  begin
   ref(Elev) ny;
   ny :- new Elev;
   ny . les;
   antall := antall+1;
   Skole(antall) :- ny;
  end;
```


Vis_Elev

```
class Elev;
begin
   integer id, aar, mnd, dag, skoleaar;
   boolean mann;
   character mate, norsk, historie;
   procedure les; <...>;
   procedure vis;
   begin
      outtext("Elev nr. "); outint(id,6); outtext(", fodt ");
      outint(dag,2); outchar('/');outint(mnd,2); outchar('/'); outint(aar,4);
      outtext(if mann then ", mann." else ", kvinne."); outimage;
      outtext("Skoleaar: "); outint(skoleaar,1); outimage;
      outtext("Karakter - matematikk: "); outchar(mate); outimage;
      outtext("                  norsk: "); outchar(norsk); outimage;
      outtext("               historie: "); outchar(historie); outimage;
   end;
end ;
```

```
ref(Elev) procedure finn_Elev(nr);
 integer nr;
begin
   integer i; boolean OK;
   i := 1; OK:=false;
   while i <= maks and not OK do
      if     Skole(i) =/= none and then
          Skole(i).id = nr then OK := true
                        else i := i+1;
   if OK then finn :- Skole(i)
         else finn :- none;
end;
```

```
procedure Vis_Elev;
begin
   integer id_nr; ref(Elev) en_elev;
   id_nr := be_om_int("Hvilken elev ? : ");
   en_elev :- <finn eleven med id_nr i tabellen Skole> ;
   if en_elev =/= none then
     en_elev . vis
   else skriv("Det finnes ingen elev med dette nummeret.");
end;
```

Fjern_Elev, Vis_Liste, Daarlige_karakterer

```
class Elev;
 begin
  . . .
  character procedure verste; verste := max(mate, max(norsk,historie));
  . . .
 end ;
. . .
procedure Vis_Liste;
 begin integer i;
    for i:=1 step 1 until maks do
       if Skole(i) =/= none then Skole(i).vis;
 end;

procedure Fjern_Elev ;
 begin integer nr, i; boolean OK;
    nr := be_om_int("Hvilken elev skal fjernes (id):") ;
    i := 1; OK := false;
    while i <= maks and not OK do
      begin
       if Skole(i) =/= none and then Skole(i).id = nr then OK:=true
       else i := i+1;
      end;
    if OK then Skole(i) :- none
    else skriv("Det finnes ingen elev med dette nummeret.");
 end;

procedure Daarlige_karakterer;
begin integer i; character grense;
  grense := be_om_char("Karaktergrense : ");
  for i:=1 step 1 until maks do
     if Skole(i )=/= none and then Skole(i).verste >= grense then Skole(i).vis;
end;
```


Klassen Elev med prosedyrer og skjulte attributter

```
class Elev;
protected aar, mnd, dag, skoleaar, mann, mate, norsk, historie;
begin

   integer id,
           aar, mnd, dag, skoleaar;
   boolean mann;
   character mate, norsk, historie;

   procedure les; <...>;
   procedure vis;<...> ;
   procedure endre; <...>;
   character procedure verste;<...> ;
end ;

integer antall, maks=1500;
ref(Elev) array Skole(1:maks);

procedure Ny_Elev;  <...> ;
procedure Vis_Elev;  <...>;
procedure Endre_Elev;  <...>;
ref(Elev) procedure finn_Elev(nr); integer nr;  <...> ;

procedure Fjern_Elev;  <...> ;
procedure Vis_Liste;  <...> ;
procedure Daarlige_karakterer;  <...> ;
```


Global og lokal datastruktur

```
begin
 class Elev;
 protected aar, mnd, dag, skoleaar, mann, mate, norsk, historie;
  begin
   integer id,
            aar, mnd, dag, skoleaar;
   boolean mann;
   character mate, norsk, historie;

   procedure les; <...>;
   procedure vis; <...>;
   procedure endre; <...>;
   character procedure verste;<...> ;
  end ;
 integer antall, maks=1500;
 ref(Elev) array Skole(1:maks);

 ref(Elev) procedure finn_Elev(nr); integer nr; ;
 procedure Ny_Elev; <...>;
 procedure Vis_Elev;<...> ;
 procedure Endre_Elev;<...> ;
 procedure Fjern_Elev; <...>;
 procedure Vis_Liste;<...>;
 procedure Daarlige_karakterer; <...>;
 . . . . .
end;
```

lokale data:
synlige
usynlige

grensesnittet
til Elev-objekter
(data access procedures)

global datastruktur

En Skole verktøykasse

```
begin
 class Elev;
 protected aar, mnd, dag, skoleaar, mann, mate, norsk, historie;
  begin
   integer id, aar, mnd, dag, skoleaar;
   boolean mann;
   character mate, norsk, historie;
   procedure les; <...>;
   procedure vis; <...>;
   procedure endre;<...> ;
   character procedure verste;<...> ;
  end ;
 integer antall, maks=1500;
 ref(Elev) array Skole(1:maks);
 ref(Elev) procedure finn_Elev(nr); integer nr; ;
 procedure Ny_Elev; <...>;
 procedure Vis_Elev; <...>;
 procedure Endre_Elev; <...>;
 procedure Fjern_Elev;<...>;
 procedure Vis_Liste;<...>;
 procedure Daarlige_karakterer;<...> ;
end;
```

```
class Skole_kasse ;
protected Elev, Skole, antall, maks, finn_Elev ;
begin
   class Elev;
    begin
     integer id, aar, mnd, dag, skoleaar;
     boolean mann;
     character mate, norsk, historie;
     procedure les;<...> ;
     procedure vis; <...>;
     procedure endre; <...>;
     character procedure verste; <...>;
    end ;
   integer antall;  integer maks = 1500;
   ref(Elev) array Skole(1:maks);
   ref(Elev) procedure finn_Elev(nr); integer nr; ;
   procedure Ny_Elev;<...>;
   procedure Vis_Elev;<...> ;
   procedure Endre_Elev; <...>;
   procedure Fjern_Elev; <...>;
   procedure Vis_Liste;<...>;
   procedure Daarlige_karakterer; <...>;
   integer procedure Elever; Elever:=antall;
 end;
 ref(Skole_kasse) en_Skole;
```

Bruk av Skole_kasse'n

begin

class Skole_Kasse; . . . ;

ref(Skole_kasse) en_skole;
en_skole :- **new** Skole_kasse ;

. . .
while ordre <> 'q' **do**
begin
if ordre='n' **then** en_skole . Ny_Elev
else if ordre='f' **then** en_skole . Fjern_Elev
. . .
end;

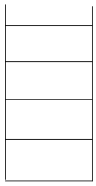
end

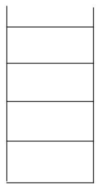
Et lite problem ?

```
class Skole_kasse;
protected Elev, Skole, antall, maks, . . . ;
begin
    class Elev; . . . ;
    integer antall;   integer maks=1500;
    ref(Elev) array Skole(1:maks);
    . . .
    integer procedure Elever; Elever:=antall;

    procedure Ny_Elev;
     if antall >= maks then skriv("Ikke mer plass")
     else begin
        ref(Elev) ny;
        ny :- new Elev;
        ny . les;
        antall := antall+1;
        Skole(antall) :- ny;
     end;

    . . .

end;
```

```
procedure Fjern_Elev ;
 begin integer nr, i; boolean OK;
    nr := be_om_int("Hvilken elev skal fjernes (id): ");
    i := 1; OK := false;
    while i <= maks and not OK do
     begin
      if Skole(i) =/= none and then
        Skole(i).id = nr then OK:=true
      else i := i+1;
     end;
    if OK then Skole(i) :- none
    else skriv("Ingen elev med dette nummeret.");
 end;
```

Ny_Elev ;
Ny_Elev ;
Ny_Elev ;
Fjern_Elev ;
Ny_Elev ;





Data*invariant*

"definerer" datastrukturen

- skal uttrykke sammenhengen mellom forskjellige elementer av datastrukturen (hos oss mellom *antall, maks* og *Skole*-tabellen)
- skal opprettholdes av alle prosedyrene som oppdaterer datastrukturen

1) antall elever er aldri større en maks

antall <= *maks*

2) antall skal være det faktiske antall elever i skolen
hvis *antall* < *maks* skal det være mulig å sette inn en ny elev

for alle 1<=i<=*antall* : *Skole*(i) =/= **none** (peker på en elev)
for alle *antall*<i<=*maks* : *Skole*(i) == **none**

3) det finnes høyst 1 elev med et gitt id-nummer

for alle 1<=i,j<=antall : **if** (i<>j) **then** Skole(i).id<>Skole(j).id

Sjekk om datainvarianten holder:

a) dersom datastrukturen initialiseres – se om invarianten holder etter initialisering,
b) for hver oppdateringsprosedyre - se om antagelsen at invarianten holder før kallet
.... impliserer at den vil også holde etter at prosedyren har returnert.

Vi har to oppdateringsprosedyrer: Ny_Elev og Fjern_Elev

antall=X, X<=*maks*,
(i, 1<=i<=*antal* : *Skole*(i).id=Nr) = B
 	
Ny_Elev ! med id = Nr ;
 	
if X=*maks* **then**
Skole, antall, maks uforandret
else if B **then**
Skole, antall, maks uforandret
else
antall=X+1, *Skole*(*antall*).id=Nr
	antall=X, X<=*maks*,
	(i, 1<=i<=*antall*: *Skole*(i).id=Nr) = B
	
	Fjern_Elev ! med id = Nr ;
	
	if not B **then**
	Skole, antall, maks uforandret
	else
	antall=X-1,
	Skole(X) == **none**,
	for all 1<=i<= *antall*: **not** *Skole*(i).id=Nr

Opprettholdelse av Invarianten

- antall <= maks
- for 1<=i<=antall : Skole(i) peker på et Elev-objekt (=/= none)
- for antall<i<=maks : Skole(i) == none
- alle elever (Skole(1)...Skole(antall)) har forskjellige verdier på id-attributtet

```
class Skole_kasse;
protected Elev, Skole, antall, maks, . . . ;
begin
   class Elev;   procedure les_id; id:=be_om_int("ID nummer:"); procedure les; ;. . ;
   integer antall;   integer maks=1500;
   ref(Elev) array Skole(1:maks);
   . . .
   integer procedure Elever; Elever:=antall;
   . . .
   procedure Ny_Elev;
    if antall >= maks then skriv("Ikke mer plass")
    else begin
       ref(Elev) ny;
       ny :- new Elev;
       ny . les_id;
       if finn_Elev(ny.id) =/= none then
         skriv("Eleven finns fra for")
       else begin
         ny . les;
         antall := antall+1;
         Skole(antall) :- ny;
       end;
    end;

   . . .

end;
```

```
procedure Fjern_Elev ;
 begin integer nr, i; boolean OK;
    nr := be_om_int("Hvem skal fjernes (id): ");
    i := 1; OK := false;
    while i <= maks and not OK do
     begin
      if Skole(i) =/= none and then
        Skole(i).id = nr then OK:=true
      else i := i+1;
     end;
    if OK then begin
     if i < antall then Skole(i) :– Skole(antall) ;
     Skole(antall) :- none ;
     antall := antall - 1;
    end else skriv("Eleven finns ikke.");
 end;
```