Uke 7

Repetisjonstime

Grensensitt, referanser og typekonverting

- Typen til peker kan være annerledes fra typen til verdi
  - peker kan være interface
- To forskjellige former for casting
- Implisitt og eksplisitt
  - Snevrer inn eller ut?
  - Int -> Float er automatisk
  - Float -> Eksplisitt
    - ```
      int minInt = (int) minFloat
      ```
- Egendefinerte klasser må man passe på at verdien pekeren peker på har de samme egenskapene
      ```

Oppgave 1

Du skal lage et enkelt program som simulerer en kattekafé. I denne kaféen kan besøkende interagere med forskjellige katter. Katterne har ulike egenskaper og adferd, men de deler noen felles trekk.

1. **Interface:**
   - Definer et interface som heter `Katt` som inneholder metoden `void lek()`. Denne metoden skal representere at en katt leker eller utfører en aktivitet.
2. **Klasser:**
   - Lag to klasser, `HusKatt` og `VillKatt`, som begge implementerer `Katt`-interfacet.
     - `HusKatt` skal ha en metode `void kos()` som skriver ut at katten koser seg med besøkende.
     - `VillKatt` skal ha en metode `void jakt()` som skriver ut at katten jakter på mus.
3. **Referanser og typekonvertering:**
   - I hovedprogrammet ditt, opprett en liste av typen `Katt` som inneholder både `HusKatt`- og `VillKatt`-objekter.
   - Gå gjennom listen og kall på `lek()`-metoden for hver katt. Kall på lek() uten å bruke instanceof og konvertering
   - Bruk typekonvertering (`instanceof`) for å sjekke om en katt er en `HusKatt` eller `VillKatt`, og kall på de spesifikke metodene `kos()` eller `jakt()` avhengig av hvilken type katt det er.

Eksamensopg:

5 Pekersjonglering

```
class Person {
    String navn;
    Person (String n) { navn = n; }
}

class LagEiere {
    public static void main (String[] args) {
        Person eva = new Person("Eva");
        Person ole = new Person("Ole");
        Person per = new Person("Per");
        Person meg, du, nabo, vennen, kollega;

        meg = eva;
        du = ole;
        nabo = per;
        kollega = du;
        vennen = eva;
        du = vennen;
        System.out.println("Nr 1: " + du.navn);
        System.out.println("Nr 2: " + meg.navn);
        System.out.println("Nr 3: " + kollega.navn)
    }
}
```

Hvilke navn skrives ut?

	Eva	Ole	Per
Nr 1	○	○	○
Nr 2	○	○	○
Nr 3	○	○	○

Oppgave 2

a) Vi skal lage en enkel modell av en næringskjede. På toppen av næringskjeden finnes rovdyrene og på bunnen finnes planteeterne. Skriv to grensesnitt; ett for rovdyr og ett for planteetere. Rovdyrene skal kunne jakte, mens planteeterne skal kunne beskytte seg selv.

b) Skriv klassene Bjorn, Ulv, Elg og Sau som implementerer de riktige grensesnittene over. Metodene som må implementeres trenger kun å printe ut en beskjed om hva dyret gjør. Opprett objekter av klassene og legg alle rovdyrene i en array av rovdyr, mens du legger alle planteeterne i en array av planteetere. Test programmet ditt. *Diskusjon: hva gjør man med Bjørn; er den et rovdyr eller en planteeter, eller kan vi gjøre den til begge deler?*

c) (Ekstra) Tegn opp klassehierarkiet med alle klasser og grensesnitt.