Pragmatisk programmering

Mer om arv og subklasser

Tema for uken

- Polymorfisme
  - Annotasjon: `@Override`
  - Nøkkelord: `super`
- Litt om try / catch
- Mer om referanser og typekonvertering
- Andre ønsker/behov?
  - Behov for å bruke mer tid på forrige ukes pensum?
  - (dette tar forrang!)

Polymorfisme

- En subklasse kan redefinere metoden i superklassen
  - Metoden har altså samme signatur (navn og parametre)
  - Men ulikt innhold
  - Annotasjon: @Override
- Det er metoden lengst nede i objektets klassehierarki som utføres

```java
class Hund {
    public void bjeff() {
        System.out.println("Vov-vov");
    }
}

class Rasehund extends Hund {
    @Override
    public void bjeff() {
        System.out.println("Voff-voff");
    }
}
```

```java
public static void main(String[] args) {
    Hund h = new Hund();
    Rasehund r = new Rasehund();

    h.bjeff();  // skriver ut "Vov-vov"
    r.bjeff();  // skriver ut "Voff-voff"
}
```

super

- super brukes for å aksessere variabler/metoder i objektets superklasse

```java
class Person {
    private String navn;

    Person(String navn) {
        this.navn = navn;
    }

    public void skrivUt() {
        System.out.println("Jeg heter " + navn);
    }
}
```

```java
class Student extends Person {
    private String studieprogram;

    Student(String navn, String studie) {
        super(navn);
        studieprogram = studie;
    }

    @Override
    public void skrivUt() {
        super.skrivUt();
        System.out.println("Jeg studerer " + studieprogram);
    }
}
```

Try / catch

- Brukes hovedsaklig for feilhåndtering

```java
import java.util.Scanner;

class Main {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        // vi prøver å kjøre denne kodesnutten
        try {
            int x = 10;
            int y = s.nextInt();
            double svar = x/y;

            System.out.println(x + " delt på " + y + " er " + svar);
        // hvis noe går galt, får vi denne beskjeden i stedet for at programmet krasjer
        } catch (Exception e) {
            System.out.println("Noe gikk galt");
        }
    }
}
```

```
lisbethyang@macbookair IN1010 % java Main.java
5
10 delt på 5 er 2.0
lisbethyang@macbookair IN1010 % java Main.java
fem
Noe gikk galt
lisbethyang@macbookair IN1010 % java Main.java
0
Noe gikk galt
lisbethyang@macbookair IN1010 %
```

Referanser (quiz!)

```java
public static void main(String[] args) {
    Dyr d; Hund h; Katt k; Pomeranian p;

    d = new Hund();

    // Hva er lovlige neste linjer av kode?
    h = new Hund();
    h = d;
    h = (Hund) d;
    h = new Pomeranian();
    p = (Pomeranian) d;
    k = d;
    k = (Katt) d;
}
```

Referanser (quiz!)

```
public static void main(String[] args) {
    Dyr d; Hund h; Katt k; Pomeranian p;

    d = new Hund();

    // Hva er lovlige neste linjer av kode?
    h = new Hund();          // lovlig!
    h = d;                   // kompilerer ikke
    h = (Hund) d;            // lovlig!
    h = new Pomeranian();    // lovlig!
    p = (Pomeranian) d;      // kompilerer, men kjører ikke
    k = d;                   // kompilerer ikke
    k = (Katt) d;            // kompilerer, men kjører ikke
}
```