

Programmeringsteknik och Matlab

Övning 5

Dagens program

Övningsgrupp 2 (Sal Q22/E32)

Johannes Hjorth
hjorth@nada.kth.se
Rum 4538 på plan 5 i D-huset
08 - 790 69 02

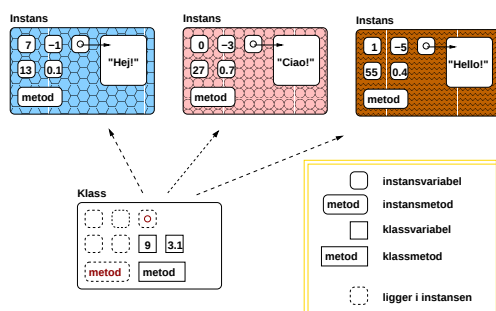
Kurshemsida:
<http://www.nada.kth.se/kurser/kth/2D1312>

Övningsanteckningar:
<http://www.nada.kth.se/~hjorth/teaching/prgi05>

- Mer om klasser och instanser
Klassmetoder, instansmetoder, klassvariabler och instansvariabler
- Felhantering, hur gör vi om det blir fel?
Vi använder try och catch
- Inläsning från fil
Om vi skriver ett dokument vill vi kunna lagra det
- Frågestund om Lab5
Vad tycker ni är svårt?

Mer om klasser och instanser

Vi skapar flera instanser med vår klass som mall.



I klassen *beskrivs* alla variabler och metoder.

Klassvariablerna ligger lagrade i klassen medan instansvariablernas värden finns i instanserna.

Det står `static` framför klassvariabler och klassmetoder. Vår `main` är en klassmetod.

Vad skiljer dem åt?

Vad skiljer instansmetoder och klassmetoder?

En klassmetod vet vilken klass den tillhör så kan använda klassvariabler ur den klassen.

En instansmetod vet vilken instans den tillhör (och därmed också vilken klass), så den kan använda både instansvariabler och klassvariabler.

Kan en klassmetod använda instansvariabler?

Svaret är nej – klassmetoden vet inte vilken instans den ska titta inuti för att hitta variabeln.

Repetition - hur skapar vi en Jacuzzi?

I filen `Jacuzzi.java` skapar vi en ny klass:

```
public class Jacuzzi {

    private int temperatur; // instansvariabel
    private String märke;   // instansvariabel

    Jacuzzi(int temperatur, String märke) { // konstruktor
        this.temperatur = temperatur;
        this.märke = märke;
    }

    Jacuzzi(int temperatur) { // konstruktor
        this.temperatur = temperatur;
        märke = "okänt";
    }

    public void setTemperatur(int t) { // instansmetod
        temperatur = t;
    }

    public String toString() { // instansmetod
        return "Jacuzzin av märket " + märke
            + " har temperatur " + temperatur + " grader";
    }
}
```

Instansvariablerna är `private` eftersom vi inte vill att de ska synas utifrån. Bara objektets egna metoder ska kunna ändra på deras värden.

Repetition ArrayList med instanser

Ifall vi inte på förhand vet hur många objekt vi behöver spara använder vi oss av `ArrayList`.

```
import java.util.*;

public class Badhus {

    public static void main(String[] inparametrar) {
        String märke = "startvärde, så vi kommer in i loopen";
        int temp;

        ArrayList<Jacuzzi> utbud = new ArrayList<Jacuzzi>();
        Scanner scan = new Scanner(System.in);

        System.out.print("Jacuzzi märke: "); // Fråga första
        märke = scan.nextLine();

        while(märke.length() > 0) { // Ska vi fortsätta?

            System.out.print("Temperatur: ");
            temp = scan.nextInt();
            scan.nextLine(); // Rensa bort radslut mm.
            utbud.add(new Jacuzzi(temp, märke)); // spara i ArrayList

            System.out.print("Jacuzzi märke: "); // Fråga nästa
            märke = scan.nextLine();
        }

        for(int i = 0; i < utbud.size(); i++) { // Skriv ut *alla*
            Jacuzzi tmp = utbud.get(i);
            System.out.println(tmp); // här anropas tmp.toString()
        }
    }
}
```

Vi kompilerar och kör `Badhus.java`

```
>javac Badhus.java
>java Badhus
Jacuzzi märke: Sundance
Temperatur: 47
Jacuzzi märke: IKEA
Temperatur: 42
Jacuzzi märke:
Jacuzzin av märket Sundance har temperatur 47 grader
Jacuzzin av märket IKEA har temperatur 42 grader
```

Tjusigt.

Antag nu att vi har en klåfingrig lillebror som försöker köra programmet...

```
>java Badhus
Jacuzzi märke: Teknikmagasinet
Temperatur: -237.15
Exception in thread "main" java.util.InputMismatchException
    at java.util.Scanner.throwFor(Scanner.java:819)
    at java.util.Scanner.next(Scanner.java:1431)
    at java.util.Scanner.nextInt(Scanner.java:2040)
    at java.util.Scanner.nextInt(Scanner.java:2000)
    at Badhus.main(Badhus.java:18)
```

Ack! Hur gör vi om vi inte kan låsa in brorsan?

Felhantering med `try` och `catch`

Vi fick ett `java.util.InputMismatchException`.

Med hjälp av en `try/catch`-sats kan vi fånga upp felet och ta hand om det så att programmet inte kraschar!

```
try { // Dags att vara försiktig...
    System.out.print("Temperatur: ");
    temp = scan.nextInt(); // här kan det gå snett...
    scan.nextLine();
    utbud.add(new Jacuzzi(temp, märke));

    System.out.print("Jacuzzi märke: ");
    märke = scan.nextLine();
}
catch(java.util.InputMismatchException e) { // fånga felet!
    System.out.println("Åhaj... försök igen!");
    scan.nextLine();
}
```

När ett exception kommer i `try`-satsen hoppar vi genast till `catch`-satsen. Där skriver vi ut ett felmeddelande och rensar den felaktiga raden.

Vad händer sen, vad gör programmet efter att felmeddelandet skrivits ut?

Den färdiga Badhus.java

```
import java.util.*;

public class Badhus {

    public static void main(String[] inparametrar) {
        String märke = "startvärde, så vi kommer in i loopen";
        int temp;

        ArrayList<Jacuzzi> utbud = new ArrayList<Jacuzzi>();
        Scanner scan = new Scanner(System.in);

        System.out.print("Jacuzzi märke: ");
        märke = scan.nextLine();

        while(märke.length() > 0) {
            try { // Dags att vara försiktig...
                System.out.print("Temperatur: ");
                temp = scan.nextInt();
                scan.nextLine();
                utbud.add(new Jacuzzi(temp, märke));

                System.out.print("Jacuzzi märke: ");
                märke = scan.nextLine();
            }
            catch(java.util.InputMismatchException e) { // fånga felet!
                System.out.println("Ajaj... försök igen!");
                scan.nextLine();
            }
        }

        for(int i = 0; i < utbud.size(); i++) {
            Jacuzzi tmp = utbud.get(i);
            System.out.println(tmp); // här anropas tmp.toString()
        }
    }
}
```

Vi kompilerar och testar Badhus.java

```
>javac Badhus.java
>java Badhus
Jacuzzi märke: Nordpolen
Temperatur: iskallt!
Ajaj... försök igen!
Temperatur: 4
Jacuzzi märke:
Jacuzzin av märket Nordpolen har temperatur 4 grader
```

Nu kraschar inte längre om vi skriver fel!



Inläsning från fil

Ibland behöver vi kunna läsa in data från en fil.

```
import java.util.*;
import java.io.*;

public class FilInläsning {

    public static void main(String[] args) {

        // Varför måste scan deklarerars utanför try-satsen?
        Scanner scan = null;
        File inFil;

        ArrayList<String> rader = new ArrayList<String>();

        try {
            inFil = new File(args[0]); // Öppna en ny fil
            scan = new Scanner(inFil); // Koppla scan till filen
        }
        catch (FileNotFoundException e) { // Fånga upp eventuella fel
            System.out.println("Kan inte öppna filen!");
            System.exit(0);
        }

        while(scan.hasNextLine()) { // Läs så länge det finns rader kvar
            String s = scan.nextLine();
            rader.add(s);
        }

        for(int i = 0; i < rader.size(); i++) {
            System.out.println(rader.get(i));
        }
    }
}
```

Vi kör Filinläsning.java

```
>javac FilInläsning.java
>java FilInläsning maewest.txt
When I'm good I'm very good, but when I'm bad I'm better.
He who hesitates is last.
Why don't you come on up and see me sometime..
when I've got nothin' on but the radio.
I generally avoid temptation unless I can't resist it.
When a girl goes wrong, men go right... after her
Is that a gun in your pocket or are you just happy to see me?
Those who are easily shocked... should be shocked more often
When choosing between two evils,
I always like to try the one I've never tried before.
```

Det ser ut att fungera...

Frågestund!

Är det något som känns oklart? Exempel på frågor:

- Vad skiljer klassmetoder och instansmetoder åt?
- Hur skapar vi klassvariabler och instansvariabler?
- Vad är `public` och `private` bra för?
- Om vi är osäkra på något om `ArrayList`, hur kollar vi då upp det i Java API?
- Vad händer om vi delar heltalet 7 med 3?
- När använder vi `=` och `==` i vårt program?
- I vilken klass ska vi ha `main`-metoden?