

ITK:P1 Föreläsning 1



Att programmera i
språket Java

DSV Peter Mozelius

Programmering

- Olika typer av programmering som t ex:
 - Imperativ programmering (**C**, **Pascal** m fl)
 - Funktionell programmering (**Lisp**, **Scheme**)
 - Logikprogrammering (**Prolog**)
 - Objektorienterad programmering (**C++**, **Java** m fl)
- Mest objektorienterad programmering i språket **Java** här på ITK:P1

2

Programmeringsspråket Java



- **Stark typning**
- **Explicit typning**
- **Strukturerat**
- **Hög säkerhet**

3

Viktiga byggstenar i programmering

- Marie kommer senare att prata en hel hel om just objektorientering
- Jag inleder nu med några viktiga byggstenar inom all sorts programmering
 - Kompilering och exekvering
 - Datatyper och variabler
 - Villkor, selektion och metoder

4

Kompilering och exekvering

- Koden för ett program sparas i en fil i en katalog med namnet **Program.java**
- Källkoden *kompileras* sedan genom:
C:\katalog> **javac** Program.java
- Om koden är utan fel skapas en klassfil:
Program.class som kan exekveras med:
C:\katalog> **java** Program

5

Programmering i Java

Ett första litet program:

```
public class Program {  
    public static void main(String[] arg){  
        System.out.print("Hello ");  
        System.out.println("World!");  
    }  
}  
//main  
}  
//Program
```

- I Javas klass **System** finns ett objekt **out** av typen
- **PrintStream** med *metoder* som `print()` och `println()`

6

Java primitiver

▪ Heltal

- ✓ byte 8 bitar
- ✓ short 16 bitar
- ✓ int 32 bitar
- ✓ long 64 bitar

Hur stora tal behöver du?

7

Java primitiver

▪ Flyttal

- ✓ float 32 bitar
- ✓ double 64 bitar

Hur många decimaler behöver du?

8

Java primitiver

▪ char

- ✓ 16 bitar
- ✓ *UNICODE*-tecken

▪ boolean

- ✓ 8 bitar
- ✓ true eller false

9

Booleska villkor i Java

- Villkor som är sanna eller falska

```
(23 > 8)
(4 <= x)
(a != b)
(7 == y)
```

Se upp med `=` (tilldelning) och `==` (jämförelse)

10

Strängar i Java

- Strängar är **INTE** primitiver i Java
- Strängar är instanser av klassen ***String***

```
String str = new String("Kista");
```

Eller kortvarianten:

```
String str = "Kista";
```

11

Strängar i Java

- I vår javaprogrammering kommer vi att dels använda klasser vi själva skrivit, dels använda färdiga klasser i Java
- För strängar finns den färdiga klassen `String`
`String str; OCH str = new String("Java");`
- ELLER det kortare
`String str = "Java";`

12

Strängar i Java

- Användbara *medlemsmetoder* i klassen String:
 - `toUpperCase()` metode som returnerar VERSALER
 - `equals(String s)` jämför två strängar
 - `equalsIgnoreCase(String s)` (ej cAsE sEnSiTiVe)
 - `length()` returnerar antalet tecken
- Läs mer i kursboken och i [API-dokumentationen](#)

13

Klassspecifikationer

- Specifikationer av Javas färdiga klasser:
<http://java.sun.com/javase/6/docs/api/index.html>
- **API** = Application Programmable Interface
- Två andra länkar med information om Java:
<http://java.sun.com/>
<http://www.javaworld.com>

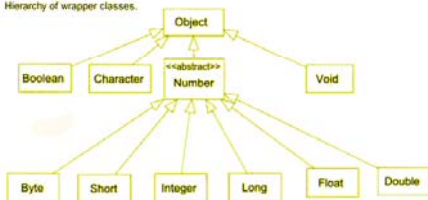
14

Paus



Omslagsklasser/wrapper classes

Hierarchy of wrapper classes.



16

Omslagsklasser/wrapper classes

- Varje primitiv datatyp har en omslagsklass som hjälper till vid t ex typomvandlingar:

```
String str = "1234";  
int i = Integer.parseInt(str);
```

```
String str = "3.14159";  
double d = Double.parseDouble(str);
```

17

Metoder

- Metod = ett kodblock som utför en uppgift
- En metod har returtyp eller void
- En metod kan ha inparametrar

Exempel:

```
public int addera(int a, int b) {  
    return a + b;  
}
```

18

Selektion i Java

- Det är ofta som programkoden inte ska exekveras i den ordning den är skriven
- Därför behöver vi olika sätt att styra vägvalet i våra program. Ett sätt är genom **if - else**:

```
if (klockan < 12)
    System.out.println("Det är förmiddag");
else
    System.out.println("Det är eftermiddag");
```

19

Kodexempel - selektion

```
public class F1_exempel2 {
    private static int timme = 11;

    public static void kontrollMetod(int klockslag){
        if (klockslag < 12)
            System.out.println("Det är förmiddag");
        else
            System.out.println("Det är eftermiddag");
    }
}
```

20

Kodexempel - selektion

```
public static void main(String[] args){
    kontrollMetod(timme);
} //main
} //F1_exempel2
```

Detta och alla andra föreläsningsexempel finns att ladda ner på kurshemsidan. Viktigt att ni testat och analyserat dessa kodexempel så att ni förstår hur de fungerar.

21

Selektion i Java

- När det finns många vägval (mer än 3) som i:

```
if (veckodag == 1)
    System.out.println("Söndag");
else if (veckodag == 2)
    System.out.println("Måndag");
...
```

Så passar det istället bättre med en **switch-sats**

22

Selektion i Java

```
switch (veckodag){
    case 1: System.out.println("Söndag"); break;
    case 2: System.out.println("Måndag"); break;
    case 3: System.out.println("Tisdag"); break;
    case 4: System.out.println("Onsdag"); break;
    case 5: System.out.println("Torsdag"); break;
    case 6: System.out.println("Fredag"); break;
    case 7: System.out.println("Lördag"); break;
    default: System.out.println("Felaktig dag");
} //switch
```

23

En fungerande Java-miljö

- ITK:P1 utgår i år ifrån JDK 6
- Finns att ladda hem från:
 - <http://java.sun.com/javase/downloads/index.jsp>
- Testa sedan att det fungerar genom:
 - PROMPT> **java -version**
 - PROMPT> **javac**

24

ITP1:s kurshemsida

- Samtliga föreläsningar, laborationer mm finns tillgängliga via nätet:

<http://www.dsv.su.se/~itp1>

Tack för mig!!



25
