TDDC77 Objektorienterad Programmering

Föreläsning 1

Sahand Sadjadee
IDA, Linköpings Universitet

Outline

- Introduktion till Java
- Variabler
- Literaler
- Utmatning

Introduktion till Java

LINKÖPINGS UNIVERSITET

Java nyckelord

abstract	continue	for	new	switch
assert	default	**goto**	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

LINKÖPINGS UNIVERSITET

Java nyckelord

Nyckelord gör **grundläggande** uppgifter i varje programmeringsspråk. Några exempel i javas fall:

- Att tilldela utrymme i minnet för att lagra data. `int, float, double, char, ...`
- Att styra programflödet. `if, while, switch, ...`
- ...

Det är inget nyckelord till exempel för att göra inmatning eller utmatning!!! Hur ska vi göra massor av andra uppgifter som kommer att behövas i programmet? Ska vi implementera de själva?

ETT STORT NEJ!

LINKÖPINGS UNIVERSITET

Ett enkelt program

- Filen måste ha samma namn som klassen.
- Kommentarer med `/* ... */` och `//...`
- Filen måste innehålla en klass, `class Namn { /* body */ }`
- En `main` metod ligger i klassen som följande
  - `public static void main(String[] args){`
    - `/* body */`
  - `}`
- Metoden main gör klassen körbar, Runnable. Den anropas automatiskt av virtuella maskinen när klassen körs genom att använda kommandot `java KlassNamnet`

Det är viktigt att använda samma signatur för `main` metoden hela tiden.

```
/* Hej.java
 * Demonstrera hur man kan mata ut texter
 * till den standard outputen
 */
class Hej
{
    // Skriv ut ett enkelt meddelande
    public static void main(String[] args)
    {
     System.out.println("Hej");
     System.out.println("... IT1 studenter!");
    }
}
```

LINKÖPINGS UNIVERSITET

Förutsättning

- Vi använder följande kod tillfälligt:

```
class Main {
    public static void main(String[] args){
        //vår kod ligger här
    }
}
```

- Java filen som innehåller koden ska ha samma namn som klassen.
- Vår kod ligger alltid i `main`.

Ett enkelt program

- `class`, `Hej`, `public`, `main`, `System`, `String`, `println` är identifierare.
- Några har vi valt själva (`Hej`, `args`).
- Vissa har andra programmerare valt (`String`, `System`, `out`, `println`). **Andra programmerare!!?**
- Vissa är nyckelord (`class`, `public`, `static`, `void`) http://docs.oracle.com/javase/tutorial/java/nutsandbolts/_keywords.html

```
/* Hej.java
 * Demonstrera hur man kan mata ut texter
 * till den standard outputen
 */
class Hej
{
    // Skriv ut ett enkelt meddelande
    public static void main(String[] args)
    {
     System.out.println("Hej");
     System.out.println("... IT1 studenter!");
    }
}
```

LINKÖPINGS UNIVERSITET

Egna identifierare(identifiers)

- Kombination av bokstäver, siffror, _, och `$`.
- Kan inte börja med en siffra.
- Exempel: `Hej`, `total`, `numberOfElements`, `$var`, `NUM_SEATS`
- Men inte: `2th`, `var#5`
- Det finns skillnad mellan stora och små bokstäver: sum, Sum och SUM är inte samma identifierare.

```
/* Hej.java
 * Demonstrera hur man kan mata ut texter
 * till den standard outputen
 */
class Hej
{
    // Skriv ut ett enkelt meddelande
    public static void main(String[] args)
    {
     System.out.println("Hej");
     System.out.println("... IT1 studenter!");
    }
}
```

LINKÖPINGS UNIVERSITET

Blanktecken(White spaces)

- Java användar blanktecken för att separera ord.
- Man kan formatera sin kod på olika sätt.
- Man ska sträva efter att göra programmet så läsbart som möjligt

INTE BRA!

```
/* Demonstrera hur man kan mata ut texter
 * till den standard outputen
 */ class Hej{ // Skriv ut ett enkelt meddelande
public static void main(String[] args){ System.out.println("Hej");
System.out.println("... IT1 studenter!"); }}
```

```
/* Hej.java
 * Demonstrera hur man kan mata ut texter
 * till den standard outputen
 */
class Hej
{
    // Skriv ut ett enkelt meddelande
    public static void main(String[] args)
    {
     System.out.println("Hej");
     System.out.println("... IT1 studenter!");
    }
}
```

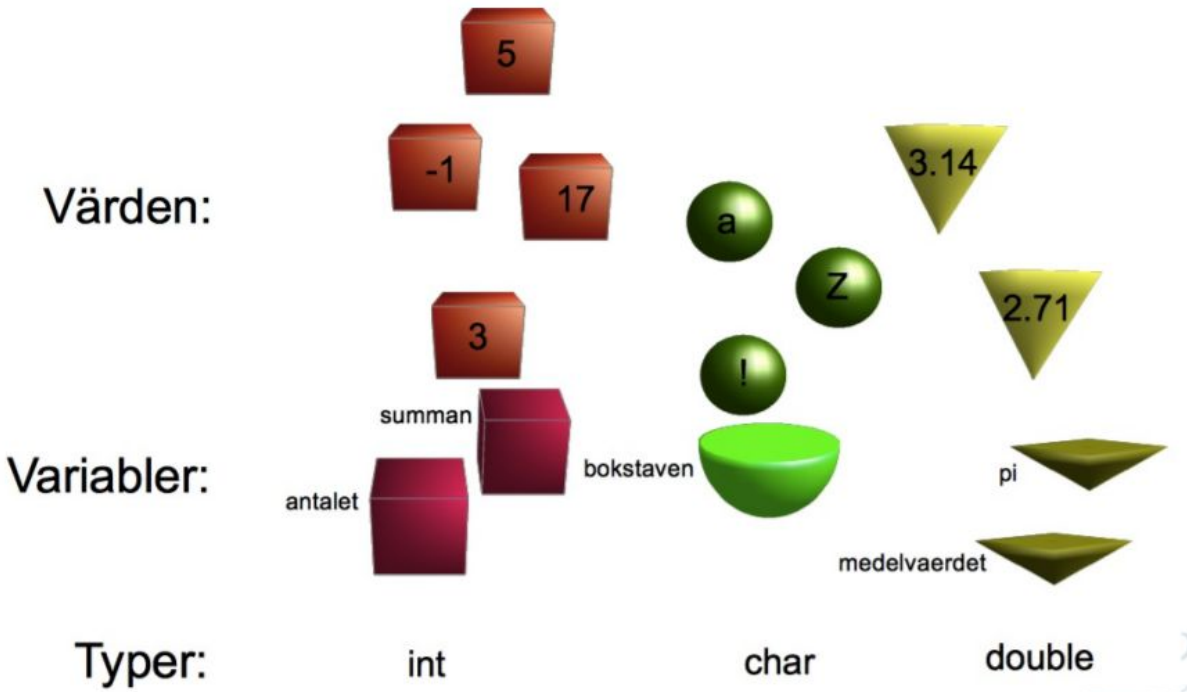
LINKÖPINGS UNIVERSITET

Variabler

Variabler

- En variabel är en plats i primärminnet.
- Den används för att lagra ett värde.
- Varje variabel har ett namn/identifierare.
- Varje variabel har också en typ.

Variabler och Datatyper



LINKÖPINGS UNIVERSITET

Primitiva Datatyper

Type	Description	Default	Size
`boolean`	true or false	false	1 bit
`byte`	two complement integer	0	8 bits
`char`	Unicode character	\u0000	16 bits
`short`	two complement integer	0	16 bits
`int`	two complement integer	0	32 bits
`long`	two complement integer	0	64 bits
`float`	IEEE 754 floating point	0.0	32 bits
`double`	IEEE 754 floating point	0.0	64 bits
`void`	None/Nothing		

LINKÖPINGS UNIVERSITET

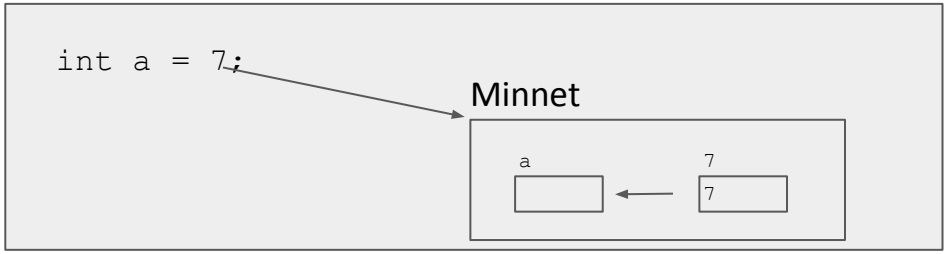
Deklaration

- Variabler måste deklareras.
- Deklarationer associerar en typ(till exempel, `int`, `char`, `String)` till ett namn(till exempel, `age, height, savingsBalance)`.
  - `int age;`
- Namnet ska vara meningfult.
- Namnet ska vara en giltig identifierare.
- Alla ord i namnet börjar med stora bokstäver förutom första ordet. Till exempel, `String myCar, int totalValue`.
- Man använder `final` för att deklarera konstanter. Namn på konstanter skrivs i stora bokstäver, t.ex: `final double PI =  3.141519, final int MAXPLACES = 7`.
- Konstanter ska initieras vid deklaration.

LINKÖPINGS UNIVERSITET

Literaler

- En literal är ett konstant värde som skrivs direkt i koden.
- Liksom variabler lagras literaler också i primärminnet.
- En literal har också en typ.
- När virtuella maskinen vill lagra en literal i minnet reserverar den ett utrymme i minnet. Storleken på utrymmet bestäms av typen av literalen. Sedan lagrar den literalen i utrymmet och nämner också utrymmet samma som värdet.
- Tilldelning är ett sätt att ge en variabel ett värde.



LINKÖPINGS UNIVERSITET

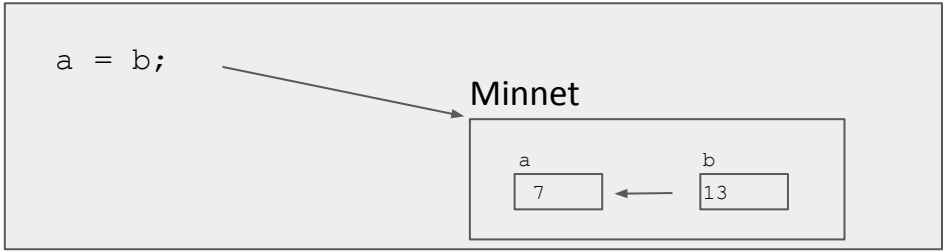
Typer på literaler

Literal	Type
‘A’, ‘c’, ‘z’	char
11, 15, 2018	byte eller short eller int(beror på storleken)
3.141519	double
3.141519f	float
3.141519d	double
true, false	boolean
“Hello TDDC77!”	String

LINKÖPINGS UNIVERSITET

Tilldelning (assignment)

- ett sätt att ge en variabel ett värde.
  - `int a = 7; eller int a; a = 7;`
  - `int b = 13; eller int b; b = 7;`
  - `a = b;`
- *En variabel ska **initieras** med något värde innan den kan användas.*
- Storleken på typen på högersidan ska vara alltid mindre än storleken på typen på vänstersidan.



LINKÖPINGS UNIVERSITET

Tilldelning

Left-side	Right-side
boolean	boolean
char	char, short
byte	byte
short	char, byte
int	byte, char, short
long	byte, char, short, int
float	byte, char, short, int
double	byte, char, short, int, float

LINKÖPINGS UNIVERSITET

Utmatning: System.out.print och System.out.println

- Att skriva till skärmen, en fil, nätverk ... alltså data som kommer från programmet.
- Brukar vara ganska lätthanterligt.
- `System.out.print` och `System.out.println` kommer ni att använda oftast.

Utmaning: println

```java
/* StraengVar.java
 *
 * Illustrera konceptet av en sträng variabeln
 */
public class StraengVar{

    // Metoden deklarera och tilldela en sträng variabel
    public static void main(String[] args){
     String exampleNamn="TDDC77";
     String otherName = exampleNamn;
     System.out.println("Kursen heter: ");
     System.out.println(exampleNamn);
     System.out.println(otherName);
    }
}
```

LINKÖPINGS UNIVERSITET

Strängar

- Exempel på en icke-primitiv datatyp, eller så kallade referenstyper
- “Rader av tecken”, “TDDC77:are”
- Man kan inte ändra på en sträng efter att ha skapat den.

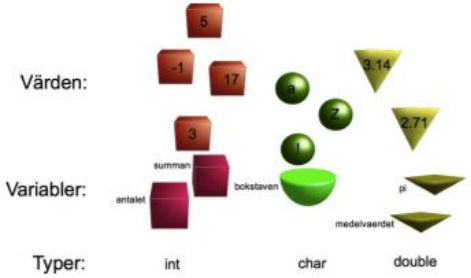
Att konkatenera sträng litteraler (string literals)

```
/* KlaraPlus.java
 *
 * Illustrera Strängskonkatenering
 */

public class KlaraPlus
{
    // Metoden konkatenera och skriver ut "klara färdiga gå!"
    public static void main(String[] args){
     System.out.print("Klara" + "... färdiga" + "... ... gå!!!");
    }
}
```

LINKÖPINGS UNIVERSITET

Variabler och Datatyper




```
public class DeklarationOchTilldelning
{
    public static void main(String[] args)
    {
     int talet=4, sum=0;
     double pi=3.14159265359, meddelvaerde;
     char bokstaven='!';
     String namnet;

     System.out.println("talet värde är: " + talet);
    }
}
```

LINKÖPINGS UNIVERSITET

Strängar

```
/* Str.java
 * Programmet demonstrerar hur man kan använda sig
 * av Java APIet för att manipulera strängar
 */
class Str{

    // Metoden anropar på olika Sträng metoder från APIet
    static public void main(String[] args){
     String kursnamn = "TDDC77", are = ":are!";

     System.out.println("char at index 1: " + kursnamn.charAt(1));

     System.out.println("compare to \"TDDC77\" gives: " + kursnamn.compareTo("TDDC77"))

     System.out.println("compare to \"Z\" gives: " + kursnamn.compareTo("Z"));

     System.out.println("Hej " + kursnamn.concat(are));
     System.out.println("Hej " + kursnamn + are);

     System.out.println("TDDC77 equals tddc77: " + kursnamn.equals("tddc77"));

     System.out.println("TDDC77 equals ignore case: "
                 + kursnamn.equalsIgnoreCase("tddc77"));

     System.out.println("Length of TDDC77: " + kursnamn.length());

     System.out.println("Replace D by d in TDDC77: "
                 + kursnamn.replace('D', 'd'));

     System.out.println("Substring of TDDC77 from 2 to 5: "
                 + kursnamn.substring(2,5));

     System.out.println("To lower case: " + kursnamn.toLowerCase());
```

LINKÖPINGS UNIVERSITET

Tack för att du lyssnade!

LINKÖPINGS UNIVERSITET