

Tips og triks for *printf* i Java

Daniel Aaron Salwerowicz

3. oktober 2017

Innhold

1	Om printf	2
2	Oppbygning av printf	2
3	Viktigste plassholdere	3
4	Ekstra funksjonalitet	3
4.1	Tallformattering	3
4.2	Stringformattering	3
4.3	Indeksering av argumenter	4
5	Praktiske eksempler	4

1 Om printf

Det er flere måter å skrive ut tekst til konsoll i Java. Men den som gir deg best kontroll over hvordan teksten din ser ut er `System.out.printf()` metoden. Ved å bruke den har du mulighet for å skrive en tekststreng med såkalte plassholdere hvor du kan sette inn formaterte tall, tekststrenger osv.

La oss si at du skal skrive ut et tekst med brukerens navn, høyde i meter og vekt i kilogram. Om du benytter deg av `println` så vil resultatet vil være:

```
System.out.println("Hei " + name + " din hoyde er: " + height + "m  
og vekten er: " + weight + "kg.");
```

```
// Resultat:
```

```
Hei Daniel din hoyde er: 1.750000m og vekten er: 68.900000kg.
```

Dersom du bruker `printf` får du et bedre resultat:

```
System.out.printf("Hei %s din hoyde er: %.2fm og vekten er:  
%.2fkg. %n", name, height, weight);
```

```
// Resultat:
```

```
Hei Daniel din hoyde er: 1.75m og vekten er: 68.90kg.
```

2 Oppbygning av printf

Printf funksjonen er oppbygd av to komponenter. Den første inneholder en tekststreng med noen plassholdere inn i seg, den andre en liste med argumenter som blir brukt i tekststrengen. Vanligvis så må det være likt antall argumenter som plassholdere i teksten, og rekkefølgen på argumenter må være samme som i teksten.

```
// Tekststreng med plassholdere  
String text = "Hei %s din kundenummer er: %d %n";  
// Tekststrengen og argumenter  
System.out.printf(text, name, clientNumber);
```

```
// Resultat:
```

```
Hei Daniel din kundenummer er: 15832.
```

3 Viktigste plassholdere

Vanligste plassholdere:

Plassholder	Brukes for å
%d	Skrive ut heltall
%f	Skrive ut desimaltall
%b	Skrive ut boolean verdier
%s	Skrive ut tekst
%c	Skrive ut enkelt tegn (char)
%t	Skrive ut dato og tid (krever ekstra plassholder kommandoer)
\n & %n	Linjeskifte
\t	Tab tegn

Tid og dato plassholdere:

Plassholder	Brukes for å få
%tF	Dato på form 2017-09-31
%tY	År i 0000-9999 format
%tm	Måned i 00-12 format
%td	Dag i 00-31 format
%tT	Tid på form 23:05:32
%tH	Time i 00-23 format
%tM	Minutt i 00-59 format
%tS	Sekund i 00-60 format

4 Ekstra funksjonalitet

4.1 Tallformattering

For å sette inn tall i teksten benytter du deg av %d for heltall og %f for flyttall. Det finnes også noen hjelpefunksjoner for å formattere de videre. Disse er da:

Plassholder	Resultat
%6d	" 42"
%06d	"000042"
%-6d	"42 "
%+6d	" +42"
%,d	"10,000,000"
%6.2f	" 42.32"

4.2 Stringformattering

Man kan bruke samme hjelpefunksjoner for stringformattering som man bruker for tallformattering, bortsett fra den ene funksjonen. %.5s definerer *maksimum* lengde på strengen, så:

```
System.out.printf("%.5s %n", "Hello world!");
```

vil skrive ut Hello.

4.3 Indeksering av argumenter

Av og til skal du skrive ut samme tall eller string flere ganger i teksten din. Du kan selvfølgelig skrive variabelnavn flere ganger i argumentlisten til printf funksjoner, men det finnes enklere måte å gjøre dette på.

Det er såkalt indeksering. Ved å skrive et tall og \$ etter % kan man spesifisere hvilken argument fra argumentlisten skal man benytte seg av. For eksempel:

```
System.out.printf("Du sa at du er %d aar gammel, men er du  
sikkert at du er %1$d aar? %n", age);
```

5 Praktiske eksempler

Til slutt legger jeg ved en kodeeksempel med noen eksempler på hvordan printf kan brukes. MEN det viktigste rådet er: praksis. Prøv å bruke printf selv, prøv å erstatte println med printf i alle tilfeller du skal skrive ut noe til konsoll. println egner best dersom du skal skrive statiske stringer som ikke endrer seg, eller om du bare skal legge til en variabel på slutten eller starten av strengen.

```
public class PrintfExamples {  
  
    private static java.util.Random rnd = new  
        java.util.Random(System.nanoTime() * 420);  
  
    public static void main(String[] args) {  
  
        printChars('-', 80);  
  
        double weight = 68.900000;  
        double height = 1.750000;  
        String name = "Daniel";  
  
        System.out.println("Hei " + name + " din hoyde er: " + height  
            + "m og vekten er: " + weight + "kg.");  
  
        System.out.printf("Hei %s din hoyde er: %.2fm og vekten er:  
            %.2fkg. %n", name, height, weight);  
  
        printChars('-', 80);  
  
        int clientNumber = 15832;  
        // Tekststreng med plassholdere  
        String text = "Hei %s ditt kundenummer er: %d %n";  
        // Tekststrengen og argumenter  
        System.out.printf(text, name, clientNumber);  
  
        printChars('-', 80);  
    }  
}
```

```

System.out.printf("%.5s %n", "Hello world!");

int age = 17;

System.out.printf("Du sa at du er %d aar gammel, men er du
    sikkert at du er %1$d aar? %n", age);

printChars('-', 80);

java.util.Calendar cal = new java.util.GregorianCalendar();
java.text.SimpleDateFormat formatter = new
    java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

System.out.printf("Naavaerende dato er: %tF %1$tT %n", cal);
System.out.println("Naavaerende dato er: " + cal.get(cal.YEAR)
    + "-" + cal.get(cal.MONTH) + "...");
System.out.println("Naavaerende dato er: " +
    formatter.format(cal.getTime()));

printChars('-', 80);

boolean over16 = true;

System.out.printf("Du er %d aar gamel, derfor er du%s myndig
    %n", age, age > 18 ? "" : " ikke");
System.out.printf("Du er %d aar gamel, derfor faar du%s kjoere
    moped %n", age, over16 ? "" : " ikke");
System.out.printf("Alderen din er%s gyldig. %n",
    isValidAge(age) ? "" : " ikke");

printChars('-', 80);

System.out.println(
    ",-----,");
System.out.printf("| %-15s | %-15s | %-10s |%n", "Fornavn",
    "Etternavn", "Alder");
System.out.println(
    "|-----|");

for (int i = 1; i <= 10; i++)
    System.out.printf("| %-15s | %-15s | %10d |%n",
        rndString(12), rndString(12), rndInt(80));

System.out.println(
    ",-----,");
}

```

```

private static void printChars(char c, int l) {
    System.out.println();
    for (int i = 0; i < l; i++) System.out.print(c);
    System.out.println("\n");
}

private static boolean isValidAge(int age) {
    return age > 0 && age < 125;
}

private static String rndString(int length) {
    String chars =
        "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
    StringBuilder sb = new StringBuilder(length);
    for (int i = 0; i < length; i++)
        sb.append(chars.charAt(rndInt(chars.length())));
    return sb.toString();
}

private static int rndInt(int bound) {
    return rnd.nextInt(bound);
}
}

```