Objektorienterad Programkonstruktion

KTH Datavetenskap
och kommunikation

Föreläsning 6

Kodkonventioner

Christian Smith

ccs@kth.se

Kodkonventioner

KTH Datavetenskap
och kommunikation

- Regler för hur man ska skriva kod
- Kod bli mer lättläst och lättare att förstå och ändra i, om alla skriver på samma sätt
- Man brukar ha som tumregel att 80% av arbetet med ett program handlar om att underhålla existerande kod
- Framgångsrik kod används och underhålls ofta av någon annan än originalförfattaren
- Denna föreläsning baserar sig på skriften "Code Conventions for the Java(TM) Programming Language" utgiven av Sun Microsystems, den ursprungliga Javautvecklaren

Källfiler

- En källfil bör inte innehålla mer än 2000 rader
- Varje del av en källfil (fältdeklarationer, metoddeklartioner, mm) bör skiljas åt från sin omgivning med en tom rad. Kommentarer som beskriver ett block med kod bör komma precis innan det de kommenterar, utan åtskiljande tom rad
- Om flera klasser definieras i samma fil, bör den publika klassen beskrivas först

KTH Datavetenskap
och kommunikation

Ordning inom en källfil

- **Inledande kommentarer**
- *import* **-** rader
- klass- eller gränssnitt definition(er)

```
/*
 * Classname
 *
 * Version information
 *
 * Date
 *
 * Copyright notice
 */
```

KTH Datavetenskap och kommunikation

Ordning inom en källfil

- Inledande kommentarer
- ***import* - rader**
- **klass- eller gränssnitt definition(er)**

KTH Datavetenskap
och kommunikation

```
* Copyright notice
*/

import java.awt.*;

/**
* The class MyClass provides useful functionality
*/
public class MyClass extends yourClass{

   ...
   ...

}
```

Ordning inom en definition av en klass eller ett gränssnitt

- Kommentar om klassen/gränsnittet för dokumentation (/**...*/)
- Klass- eller gränssnittsdeklarationen
- Kommentar angående klass-/gränsnittsimplementation som inte passar i dokumentationen (/*...*/)
- Klassvariabler (static) -public, protected, package, private
- Instansvariabler - public, protected, package, private
- Konstruktorer
- Metoder - Dessa bör grupperas efter funktion och inte efter åtkomst. Det är OK att stoppa in en privat klassmetod mellan två publika instansmetoder om de tematiskt hör ihop.

- Kommentarer till var och en av dessa bör komma precis ovan deklarationen

KTH Datavetenskap och kommunikation

Rader

- Alla indenteringar ska vara 4 mellanslag (eller tab, om det blir lika långt)
- Rader ska vara max 80 tecken långa, inklusive blanksteg och tabbar
- Radbrytningar:
  - efter komma
  - före operatorer
  - mellan två element av högre nivå
  - placera ny rad vid samma nivå som motsvarande element på tidigare rad
  - Flytta in ny rad 8 tecken om det blir bättre så

KTH Datavetenskap och kommunikation

Exempel

```
someMethod(longExpression1, longExpression2, longExpression3,
           longExpression4, longExpression5);

myVar = someMethod1(longExpression1,
                    someMethod2(longExpression2,
                                longExpression3));

private static synchronized horkingLongMethodName(int anArg,
        Object anotherArg, String yetAnotherArg,
        Object andStillAnother) {
    ...
}
```

```
longName1 = longName2 * (longName3 + longName4 - longName5)
            + 4 * longname6; // BRA

longName1 = longName2 * (longName3 + longName4
                         - longName5) + 4 * longname6; // DÅLIG
```

KTH Datavetenskap och kommunikation

Mer Exempel

KTH Datavetenskap
och kommunikation

```
if ((condition1 && condition2)
        || (condition3 && condition4)
        ||!(condition5 && condition6)) {
    doSomethingAboutIt();
}

if ((condition1 && condition2) || (condition3 && condition4)
        ||!(condition5 && condition6)) {
    doSomethingAboutIt();
}

//EXEMPEL PÅ DÅLIG FORMATTERING:
if ((condition1 && condition2)
    || (condition3 && condition4)
    ||!(condition5 && condition6)) {
    doSomethingAboutIt();    //DEN HÄR RADEN ÄR LÄTT ATT MISSA
}
```

Kommentarer

- Före varje deklaration kommer en blockkommentar som beskriver fältet eller metoden. Kom ihåg att `/**` inleder en dokumentationskommentar
- Blockkommentarer indenteras till samma nivå som koden de beskriver
- Blockkommentarer inne i kod föregås av en tom rad

```
/**
* Returns the name as a String
*/
public String getName(){
    String returnString;

    /* build the return string */
    returnString = stringA + stringB;
    ...
}
```

KTH Datavetenskap och kommunikation

Kommentarer på samma rad

- Korta kommentarer om implementationen kan skrivas på samma rad.
- Om man har flera kommentarer i högerkant bör de indenteras lika

```
if (a == 2) {
    return TRUE;            /* special case */
} else {
    return isPrime(a);      /* works only for odd a */
}
```

KTH Datavetenskap
och kommunikation

Kommentarer på samma rad

- Korta kommentarer om implementationen kan skrivas på samma rad. Man kan använda dubbla snedstreck // för att kommentera bort slutet av en rad. Används också till att (tillfälligt) plocka bort några rader ur koden

```
if (foo > 1) {

    // Do a double-flip.
    ...
}
else{
    return false;       // Explain why here.
}
//if (bar > 1) {
//
//    // Do a triple-flip.
//    ...
//}
```

KTH Datavetenskap och kommunikation

Variabeldeklarationer

- Försök begränsa koden till en deklaration per rad, och undvik under alla omständigheter att deklarera olika typer på samma rad

KTH Datavetenskap
och kommunikation

```
int level; // indentation level
int size;  // size of table

int     level;          // indentation level
int     size;           // size of table
Object  currentEntry;   // currently selected table entry
```

Variabeldeklarationer

- Deklarera variabler i början av det block där de används
- Undvik att återanvända samma namn på olika nivåer
- Initialisera om möjligt lokala variabler där de deklareras

KTH Datavetenskap
och kommunikation

```
public void myMethod() {
    int int1 = 0;           // beginning of method block

    if (condition) {
        int int2 = 0;       // beginning of "if" block
        ...
    }
}
```

Klasser, gränsnitt och metoder

- Sätt första klammern { sist på deklarationsraden
- Indentera avslutande klammer } till samma position
- Tom rad mellan metoder

```
class Sample extends Object {
    int ivar1;
    int ivar2;

    Sample(int i, int j) {
        ivar1 = i;
        ivar2 = j;
    }

    int emptyMethod() {}
}
```

KTH Datavetenskap och kommunikation

Mellanslag

- Använd ett mellanslag mellan ett nyckelord och parentesen med dess argument
- Sätt mellanslag efter komman, och mellan **binära** operatorer (operator mellan två argument)

```
a += c + d;
a = (a + b) / (c * d);

for (int i = 0;  i<5; i++) {
   n++;
}
printSize("size is " + foo + "\n");
doSomethingNice(argument1, agument2);
```

KTH Datavetenskap
och kommunikation

Namn

KTH Datavetenskap
och kommunikation

- Klassnamn är substantiv, med inledande versal på alla ingående termer, tex `FileViewer, MyButton, MemoryHog`
- Gränssnitt följer samma regler som klasser, men kan vara adjektiv, tex `Viewable`
- Metoder inleds med verb, där alla ingående termer utom den första har inledande versal, ex: `getName`, `printResultToScreen`
- Variabler är substantiv, där alla ingående termer utom den första har inledande versal, ex: `name`, `currentObject`
- Lokala variabler, som bara används i ett begränsat sammanhag kan vara en-bokstaviga om det underlättar läsbarheten i formler (undvik 'l'), ex: `i, n, a`
- Konstanter är helt i versaler med understreck mellan delarna, ex: `PI, MAX_AGE`

Om XML

KTH Datavetenskap
och kommunikation

- e**X**tensible **M**arkup **L**anguage
- Använder 'taggar' för att förse texter med metainformation, så att resultatet kan läsas relativt enkelt av både människor och maskiner
- All data organiseras som element.
- Ett element kan bestå av ett par av taggar (en start-tag och en slut-tag) med text emellan. Taggarna anger då meta-information om texten mellan dem.
- Ett element kan också vara en ensam (empty element) tag, som kodar någon specialfunktion
- Start-taggar och ensamma taggar kan dessutom innehålla attribut som ger extra information.

```
<text font="Courier">Detta är lite text</text>
```

XML-exempel

```
<kapitel nummer="1">
  <rubrik>Lorem Ipsum </rubrik>
  <text>
    Lorem ipsum dolor sit amet, consectetur
    adipisicing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua. Ut
    enim ad minim veniam, quis nostrud exercitation
    ullamco <fetstil>laboris</fetstil> nisi ut
    aliquip ex ea commodo consequat.<radbryt />
    Duis aute irure dolor in reprehenderit in
    voluptate velit esse cillum dolore eu fugiat
    nulla pariatur.
  </text>
</kapitel>
```

KTH Datavetenskap och kommunikation