Ghid de comentare a codului

# De ce sa comentam sursele?

O sursa buna nu este doar o sursa ar carei cod face exact ceea ce trebuie, ci o sursa foarte bine documentata care face ca oricare persoana ce o va citi sa inteleaga totul in detaliu.

# Utilitar pentru comentarea codului?

Tinand cont de faptul ca pentru implementarea aplicatiei vom folosi ca IDE – Eclipse, iar acesta permite generare documentatie foarte usor, am decis sa folosim JAVADOC pentru documentarea surselor.

# Good Practice

* este bine sa se evite duplicarea informatiei prezentata intr-o forma clara in interiorul codului;
* este bine sa se evite comentarii care ar putea deveni neactuale pe masura ce timpul trece;
* nu este bine sa abuzam de comentarii, nu fiecare linie de cod trebuie comentata;

# Tipuri de comentarii

Programele Java pot avea 2 feluri de comentarii:

1. comentarii de documentatie
2. comentarii de implementare

## Comentarii de documentatie

Mai sunt cunoscute si sub denumirea de comentarii javadoc si sunt specifice limbajului Java.

Acestea descriu:

* clasele
* interfetele
* constructorii
* metodele
* atributele

Ele aduc clarificari asupra rolului clasei, metodei si nu persoanelor care vor sautilizeze clasele sau metodele , dar nu sunt interesati de modul de implementare al acestora.

Acest tip de comentarii nu trebuie pozitionat in interiorul vreunei metode/constructor.

Comentariile de documentatie se scriu intre **/\*\***  si **\*/**

*Exemple:*

*-comentariu de clasa:*

***/\*\****

***\* Exemplu comentariu de clasa***

***\****

***\* MyClass- clasa pentru exemplificare comentarii***

***\****

***\* @author Andrada Costea***

***\* @version 1.00, 12 Nov 2012***

***\*/***

**public myClass {**

**}**

*-comentariu de metoda*

***/\*\****

***\* Exemplu comentariu de metoda***

***\* Scurta descriere a ceea ce face metoda myFirstMethod, 1-2 randuri***

***\****

***\* @param param1 ce reprezinta parametrul param1 al metodei***

***\* @param param2 ce reprezinta parametrul param2 al metodei***

***\* @return ce intoarce metoda***

***\* @exception e descriere exceptie***

***\****

***\*/***

**public ReturnedType myFirstMethod ( ParamType1 param1, ParamType2 param2) throws IllegalArgumentException {**

**}**

## Comentarii de implementare

Aduc clarificari asupra sursei si se adreseaza programatorilor care lucreaza sau vor lucra cu acest cod.

In acesta categorie avem 2 tipuri de comentarii:

-pe o singura linie **//**

-multilinie -intre **/\*** si **\*/**

*Exemple:*

**public Student returnStudent () {**

**int id; *//numarul matricol al elevului***

**String subject=”MPS”; *//materia pe care o studiaza elevul***

**Student student = new Student();**

***/\****

***Acesta este un comentariu multilinie.El exemplifica alte prelucrari care***

***adauga obiectului student informatii.***

***\*/***

**student.setId(id);**

**student.setSubject(subject);**

**return student;**

**}**

*Observatii:*

* Comentariile pe o singura linie (“ // ” ) dintr-o bucata de cod vor trebui sa aiba aceeasi indentare;
* Pentru o mai buna comunicare intre dezvoltatori putem folosi taguri de TODO (pentru a implementa ceva nou ) sau FIXME (pentru a repara eventualele greseli sau probleme).
* Comentariile vor fi scrise in intregime fie in romana, fie in engleza.
* Pentru alte functionalitati accesati pagina:

<http://www.oracle.com/technetwork/java/javase/documentation/index-137868.html>