



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria Informàtica



etsinf

# Un poc d'estil

Introducció a la Informàtica i a la Programació (IIP)

Curs 2019/20

Departament de Sistemes Informàtics Computació



# Continguts

- Recomanacions bàsiques: sagnat, claus, ús de majúscules i ús de l'espai en blanc

poli [format] IIP > RECURSOS

/ IIP:recursos / Software para desarrollo en Java

Software para desarrollo en Java ⓘ

☐ <http://www.bluej.org/>

☐ [Guía de instalación BlueJ en Windows](#)

☐ [Guía de instalación BlueJ en Linux](#)

☐ [Guía de instalación BlueJ en OSX](#)

☐ [Checkstyle y templates en BlueJ](#) ⓘ

Checkstyle y templates en BlueJ ⓘ

☐ [checkstyle-extension-5.4-0.jar](#)

☐ [default\\_checks.xml](#)

☐ [templates.zip](#)

/ IIP:recursos / Laboratorio

Laboratorio ⓘ

☐ [Java Platform, Standard Edition & Java Development Kit Version 11 API Specification \(Oracle\)](#) ⓘ

☐ [The Java Language Specification, Java SE 11 Edition \(Oracle\)](#) ⓘ

☐ [Convenciones Código Java.pdf](#)

☐ [Code Conventions Java.pdf](#)

☐ [Tutorial BlueJ - Eng.pdf](#)

- Bibliografia

- [Code Conventions for the Java Programming Language – Oracle](#)

(Revised April 20, 1999)

- [Checkstyle](#)

# Sagnat

- El sagnat és necessari i les instruccions de cada bloc intern han d'estar desplaçades 4 posicions a la dreta (per defecte en *BlueJ*) respecte del bloc on està inclòs

```
public class Rectangle {  
    private double base;  
    private double altura;  
  
    public Rectangle(double b, double a) {  
        base = b;  
        altura = a;  
    }  
}
```

# Claus { }

- { ha d'anar al final de la línia on comença el bloc
- } ha d'anar al principi de la línia on finalitza el bloc

```
public class Rectangle {  
  
    private double base;  
    private double altura;  
  
    public Rectangle(double b, double a) {  
        base = b;  
        altura = a;  
    }  
}
```

# Ús de majúscules

- Classes i constructors comencen per una lletra majúscula

```
public class Rectangle {  
  
    private double base;  
    private double altura;  
  
    public Rectangle(double b, double a) {  
        base = b;  
        altura = a;  
    }  
    public double getBase() {  
        return base;  
    }  
}
```

# Ús de majúscules

- En canvi, atributs, variables locals, paràmetres i la resta de mètodes comencen per una lletra minúscula

```
public class Rectangle {  
  
    private double base;  
    private double altura;  
  
    public Rectangle(double b, double a) {  
        base = b;  
        altura = a;  
    }  
  
    public double getBase() {  
        return base;  
    }  
}
```

# Ús de majúscules

- Les paraules compostes se “separen” usant una lletra majúscula

```
public class Rectangle {  
  
    private double base;  
    private double altura;  
  
    public Rectangle(double b, double a) {  
        base = b;  
        altura = a;  
    }  
    public double getBase() {  
        return base;  
    }  
}
```

# Ús de majúscules

- Les constants s'escriuen totes en majúscula i si són paraules compostes “separades” per un subguió \_

```
public class Rectangle {  
    public static final double MAX_BASE = 450.0;  
    private double base;  
    private double altura;  
  
    public Rectangle(double b, double a) {  
        base = b;  
        altura = a;  
    }  
}
```



# Ús de l'espai en blanc

- Separa els elements lèxics mitjançant espais en blanc, **EXCEPTE:**

- A la dreta de ( i a l'esquerra de )

```
return 2 * (base + altura);
```

- A l'esquerra de ( en la declaració i utilització d'un mètode

```
public Rectangle(double b, double a)  
r1.getBase();
```

- A l'esquerra de ;

```
return 2 * (base + altura);
```

- A l'esquerra de ,

```
public Rectangle(double b, double a)
```

- A l'esquerra i a la dreta de l'operador .

```
r1.getBase();
```