

Dibujando en Java

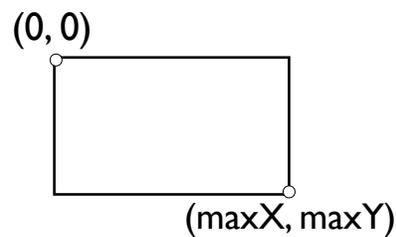
Patricio.Inostroza@dcc.uchile.cl

Consideraciones

- Independizarse del dispositivo
- Dibujar en un lugar determinado
- Mantener las características gráficas independientes del objeto que dibujo
- Dibujar en coordinación con el ambiente (sistema de ventanas)

Dónde dibujar?

- En una zona rectangular
 - Canvas
 - Component
- Bajo un sistema de coordenadas
 - Invertido !!



Propiedades Gráficas

- Hay un objeto para las propiedades gráficas:
 - `java.awt.Graphics`
- Contiene / define :
 - Colores
 - Tipo de líneas
 - Fonts
 - etc.

Cómo dibujar ?

- Graphics contiene los métodos para dibujar:
 - `g.drawLine(...)`: Dibujar una línea
 - `g.drawRect(...)`: Dibuja un rectángulo
 - `g.fillRect(...)`: Dibuja un rectángulo relleno
 - ...

Buscando un Graphics

- La clase Graphics es abstracta
 - `public abstract class Graphics`
 - NO se puede instanciar (hacer new)
- Cómo se “consigue” un Graphics ?
 - Java lo provee
 - Dónde / Cuándo... ?

Buscando un Graphics

- Todo objeto que deriva de la clase `java.awt.Component` tiene el método:
 - `public void paint(Graphics g)`
- Objetos de tipo `Component` (hijos de):
 - `Button` **Class Component**
 - `Canvas` [java.lang.Object](#)
 ↳ `java.awt.Component`
 - `Label` **All Implemented Interfaces:**
 [ImageObserver](#), [MenuContainer](#), [Serializable](#)
 - ... **Direct Known Subclasses:**
 [Button](#), [Canvas](#), [Checkbox](#), [Choice](#), [Container](#),
 [Label](#), [List](#), [Scrollbar](#), [TextComponent](#)

Buscando un Graphics

- Todo objeto que deriva de la clase `java.awt.Component` tiene el método:
 - `public void paint(Graphics g)`
- Es llamado por Java cuando es necesario dibujar el objeto
- “`paint`” NO puede ser llamado directamente
 - La clase `Graphics` es abstracta
- Pero puede ser llamado indirectamente
 - `repaint()`

Cómo dibujo ?

- Para dibujar hay que sobre-escribir el método paint.

```
class MiDibujo extends Canvas {  
    ::  
    public MiDibujo (int x1, int y1, int x2, int y2) {  
        this.x1 = x1; this.y1 = y1;  
        this.x2 = x2; this.y2 = y2;  
    }  
    public void paint(Graphics g) {  
        g.drawLine(x1, y1, x2, y2);  
    }  
    ::  
}
```

```
class Main {  
    ::  
    static public void main(String [ ] argv) {  
        MiDibujo md = new MiDibujo(1, 1, 77, 44);  
        Ventana v = new Ventana( ... );  
        v.add(md);  
        md.repaint();  
    }  
}
```