

Applets

Indice

- ☐ Arquitectura. Estructura de una applet.
- ☐ Métodos gráficos simples.
- ☐ La ventana de estado.
- ☐ Gestión de eventos.
- ☐ Etiqueta APPLET de HTML.
- ☐ Paso de parámetros a las applets.
- ☐ Salida por consola.

Definición de applet

- ☐ Las applets son pequeñas aplicaciones a las que
 - se accede en un servidor de Internet,
 - se transmiten a través de la red,
 - se instalan automáticamente y
 - se ejecutan como parte de un documento de la red (página web).
- ☐ Las clases interactúan con el usuario a través del **AWT** (y **no** con las clases de E/S basada en la consola).
- ☐ Cada applet es una **subclase** de **applet**.

Funcionamiento

- ☐ Las applets están guiadas por eventos.
 - El **AWT** notifica a la applet el evento que se ha producido llamando a un gestor de eventos que la applet ha sobrescrito.
 - La applet toma una acción y devuelve el control al **AWT**.
- ☐ La applet se ejecuta en un navegador compatible Java cuando encuentre una etiqueta **APPLET** en el archivo **HTML**.
- ☐ La clase **Applet** extiende la clase **Panel** de **AWT**.
 - **Panel** extiende la clase **Container**, que a su vez extiende la clase **Component**.

El método paint()

- ☐ Definido en **AWT** (*import java.awt.**),
- ☐ Debe ser sobrescrito por la applet.
- ☐ La applet llama a **paint()** cada vez que quiere mostrar su salida.
- ☐ El parámetro **Graphics**: contiene el contexto gráfico que describe el *entorno gráfico* (para presentar la salida) donde se ejecuta.

public void drawString (String mensaje, int x, int y);
la posición 0, 0 corresponde a la esquina sup. Izq.
(no reconoce los caracteres de nueva línea).

El Primer Ejemplo: Applet0.java

```
import java.awt.*;  
import java.applet.*;  
  
public class Applet0 extends Applet {  
    public void paint (Graphics g) {  
        g.drawString ("Una applet sencilla",20,20);  
    }  
}
```

Ejecución de una applet

- ☐ Las applets **no** son ejecutadas por el **intérprete de Java**.
- ☐ Ejecución:
 - Dentro de un navegador web compatible con Java.
 - Visualizador de applet: **appletviewer** (JDK)
- ☐ Pasos:
 - Editar un archivo fuente de Java.
 - Compilar el programa.
 - Ejecutar el visualizador de applets especificando el archivo fuente de la applet.

appletviewer Applet0.html

```
<applet code= "Applet0" width=200 height=60>  
</applet>
```



El paquete java.applet

Interfaces

AppletContext

AppletStub

AudioClip

Clases

Applet

Estructura: Inicialización

☐ **void init()**

- Es el primer método que se ejecuta. Sólo se ejecuta una vez.

☐ **void start()**

- Se ejecuta después de **init()**.
- La reinicializa después de que haya sido parada.

☐ **void paint()**

- Se ejecuta cada vez que se debe volver a pintar la salida de una applet.

Estructura: finalización

□ void stop()

- Se ejecuta cuando un navegador deja el documento HTML que contiene la applet (por ejem. Se va a otra pag.).
- Debe utilizarse para suspender hilos

□ void destroy()

- Se ejecuta cuando el entorno determina que la applet tiene que ser borrada completamente de memoria.
- El método **stop()** siempre se ejecuta antes que **destroy()**.

```
//Estructura de una Applet
import java.awt.*;
import java.applet.*;

public class Applet1 extends Applet {
    public void init(){//primer método.
        //inicialización.
    }

    public void start(){ /*2ª llamada, reanudación */
        //comienzo o reanudación de la ejecución.
    }

    public void stop(){ /*cuando se para la applet*/
        //suspende la ejecución.
    }

    public void destroy(){/*Al terminar la applet*/
        //realiza la finalización de las actividades.
    }

    public void paint (Graphics g) { /*Cuando se restaura*/
        //vuelve a dibujar la ventana.
    }
}
```

Sobrescritura del método update()

- ☐ El método está definido en **AWT**.
- ☐ Se ejecuta cuando una applet ha solicitado que se vuelva a pintar una parte de su ventana.
- ☐ La versión por defecto:
 - rellena una applet con el color de fondo y
 - después llama al método **paint()**.
- ☐ Una forma de evitar esto es sobrescribir **update()**.

```
public void update(Graphics g){  
    /*vuelve a pintar ventana*/  
}  
public void paint(Graphics g){  
    update(g);  
}
```

Métodos de gráficos simples (I)

- ☐ Pertenecen a la clase **Component**.
- ☐ **void setBackground(Color nuevoColor)**
 - Da un color a la ventana de una applet.
- ☐ **Color getBackground()**
 - Obtiene el color actual de fondo.
- ☐ **void setForeground(Color nuevoColor)**
 - Da el color de primer plano, por ejemplo al texto.
- ☐ **Color getForeground()**
 - Obtiene el color de primer plano.

La clase Color define las siguientes constantes para especificar colores

Color.black	Color.magenta
Color.blue	Color.orange
Color.cyan	Color.pink
Color.darkGray	Color.red
Color.gray	Color.white
Color.green	Color.yellow
	Color.lightGray

❑ Ejemplo:

setBackground(Color.green)

setForeground(Color.red)

Ejemplo: Applet3.java

Volver a pintar, repaint()

❑ Para actualizar la información mostrada.

void repaint()

void repaint(int izq, int sup, int ancho, int alto)

void repaint(long tiempo) en milisegundos

void repaint(long t, int izq, int sup, int anch, int alt)

❑ Ejemplo: Applet4.java

Gestión de eventos

- ☐ Hay varios tipos de eventos, los más comunes son generados por el ratón, teclado y elementos de control.
- ☐ Todos los eventos están dentro de un objeto **Event**.
- ☐ La clase **Event** es parte del **AWT**.
- ☐ Los métodos de los eventos del ratón son heredados por **Applet** y los hereda de **Component** (y los puede sobrescribir)

Los eventos del ratón

- ☐ **boolean mouseDown(Event o, int x, int y)**
 - Se ejecuta cuando se pulsa un botón del ratón.
- ☐ **boolean mouseDrag(Event o, int x, int y)**
 - Se ejecuta cuando se mueve el ratón con un botón pulsado.
- ☐ **boolean mouseEnter(Event o, int x, int y)**
 - Se ejecuta cuando el ratón entra en la ventana.
- ☐ **boolean mouseExit(Event o, int x, int y)**
 - Se ejecuta cuando el ratón sale de la ventana.
- ☐ **boolean mouseMove(Event o, int x, int y)**
 - Se ejecuta cuando el ratón se mueve.
- ☐ **boolean mouseUp(Event o, int x, int y)**
 - Se ejecuta cuando se deja de pulsar un botón del ratón.

Los eventos del teclado

- ❑ Si el usuario pulsa una tecla cuando la applet está seleccionada se produce un evento de teclado.
- ❑ **boolean keyDown(Event o, int tecla)**
 - Se ejecuta cuando se pulsa un botón del ratón.
- ❑ **boolean keyUp(Event o, int tecla)**
 - Se ejecuta cuando se pulsa un botón del ratón.
- ❑ Para letras, números y símbolos de puntuación, el valor contenido en *tecla* se puede convertir a **char**.
- ❑ Ejemplo: Applet7.java

Los eventos del teclado: teclas especiales

Constante	Correspondencia
F1	Tecla de función F1
... F12	... Tecla de función F1
PGDOWN	AvPág
PGUP	RePág.
LEFT	Cursor izquierdo
RIGHT	Cursor derecho
UP	Cursor arriba
DOWN	Cursor abajo
HOME	Inicio
END	Fin

Etiqueta **APPLET** de HTML

<APPLET

[**CODEBASE** = *URLBase*]

[**CODE** = *Archivo*]

[**ALT** = *TextoAlternativo*]

[**NAME** = *nombreInstanciaApplet*]

[**ALIGN** = *Alineamiento*]

[**VSPACE** = *pixeles*]

>

[**PARAM NAME** = *NombreAtributo* **VALUE** = *ValorAtrib*]

[Código HTML que se visualiza en ausencia de Java]

Fundamentos
de Telemática

<APPLET>

21

Área de Ingeniería Telemática

Universidad de Sevilla

Opciones

- ☐ **CODEBASE**: URL del código de la applet, que es el directorio en el que busca el archivo de clase ejecutable de la applet.
- ☐ **CODE**: Nombre del archivo compilado (**.class**).
- ☐ **ALT**: Texto que se muestra si el navegador comprende la etiqueta **APPLET** pero no puede ejecutar applets de Java.
- ☐ **NAME**: Especifica un nombre para la instancia de la applet.
- ☐ **WIDTH (HEIGHT)**: Tamaño en pixeles.
- ☐ **ALIGN**: Alineamiento de la applet.
- ☐ **VSPACE** y **HSPACE**: Espaciado en pixeles encima y debajo de la applet.
- ☐ **PARAM NAME** y **VALUE**: Es la forma de pasar argumentos específicos de la applet desde una página HTML.

Fundamentos
de Telemática

22

Área de Ingeniería Telemática

Universidad de Sevilla