

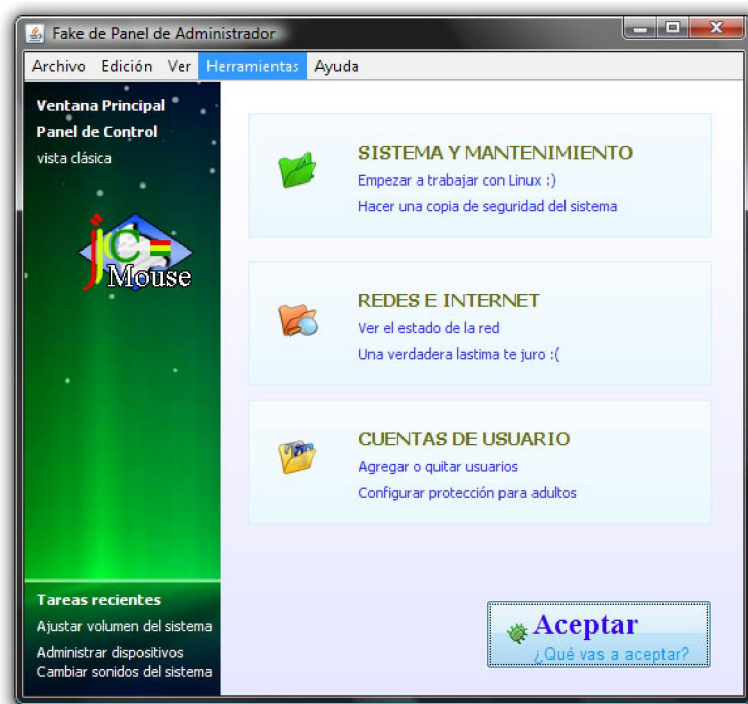
Componente Swing creado por jc Mouse

---

# jcMousePanel

## v. 1.6.9

<http://www.jc-mouse.net/>



## Componente Swing creado por jc Mouse

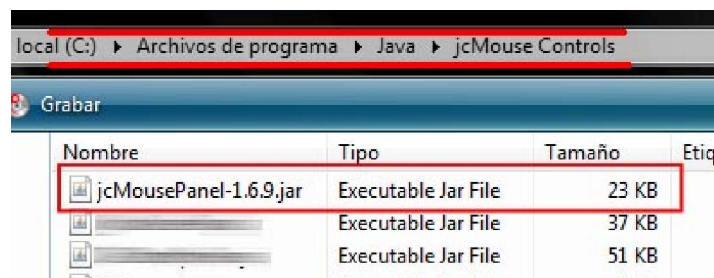
**jcMousePanel** es un componente JPanel java para trabajar con imágenes gráficamente desde un IDE como Netbeans todo esto cómodamente sin tener que escribir líneas de código y poder desarrollar interfaz de usuario más personalizadas y agradables al usuario final

La nueva versión para este año 2012 es jcMousePanel-1.6.9.jar, ahora este componente trae muchas mejoras y más opciones de uso.

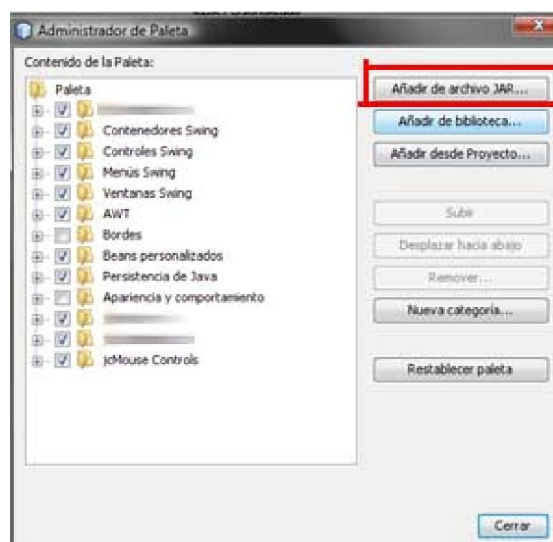
### Instalación.

Su instalación es igual a cualquier otro componente swing, aquí te digo como:

1. Descarga el componente y descomprímelo en tu disco duro, copia el archivo jcMousePanel-1.6.9.jar en la carpeta donde guardas tus componentes, por ejemplo, yo lo guardo en “c:\archivos de programa\java\jcMouse Controls\”.

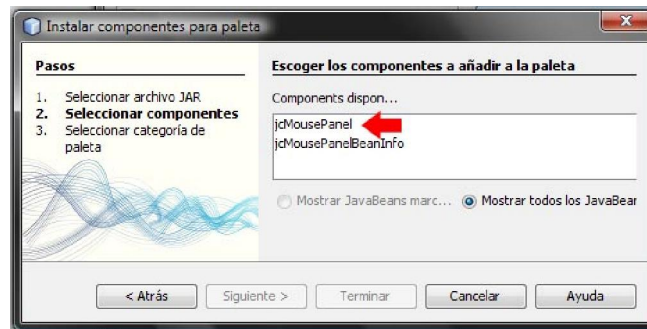


2. Ejecuta Netbeans y abre el “**Administrador de paleta...**”, para eso debes hacer un clic derecho en cualquier lugar de la paleta y buscas en el menú emergente “**Administrador de paleta...**”. En la venta que aparece clic en “**Añadir de Archivo JAR...**”



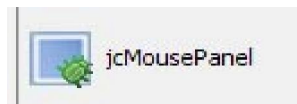
## Componente Swing creado por jc Mouse

3. Buscas el archivo jcMousePanel-1.6.9.jar clic en **"Siguiente"**, aquí importante, te muestra una lista de componentes, selecciona jcMousePanel, presionas siguiente, escoges la carpeta donde estará el componente y clic en **"Terminar"**.



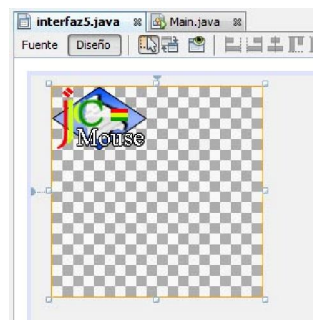
Tu componente está instalado y listo para ser utilizado, re-usado, investigado y (usted me entiende - ;) -)

Podrás identificarlo en tu paleta de componentes por el icono que tiene la forma de un rectángulo de color azul y en la parte inferior derecha un bicho de color verde y claro el nombre "jcMousePanel"



## Propiedades y formas de uso

Para comenzar a utilizar *jcMousePanel* debes arrastrar este a tu JFrame como cualquier otro componente JPanel, pero podras ver que este tiene otra forma (imagen inferior), no te asustes ni te alarmes esa es la interfaz de nuestro componente.



La forma es una cuadrícula de 200x200 y tiene además un icono (nuevo) en la parte superior izquierda. Veamos ahora que podemos hacer con este componente.

## Componente Swing creado por jc Mouse

**Modos de Uso.** Lo primero que debes conocer sobre este componente es que cuenta con 5 modos de uso, cada uno con sus propias propiedades, estas son:

- 1 = Imagen de fondo
- 2 = Transparencia sin imagen ni degradado
- 3 = Degradado 2 colores
- 4 = Imagen y transparencia
- 5 = Degradado y Transparencia

Para cambiar entre modos de uso, en las propiedades del componente busca la opción “modo” y solamente debes escribir un valor numérico entre las 5 opciones que tienes y presionar ENTER para activar el “modo de uso”, por defecto el valor es 1 (Imagen de fondo)



- **setModo(int value)**  
Cambia el modo de uso del componente  
Parámetro de Entrada: value de tipo entero (INT)
- **getModo()**  
Retorna el valor entero del modo de uso que este activo

### Colocar imagen de fondo (Modo=1)

Para colocar una imagen en el componente, se utiliza la propiedad “icon” , esta propiedad es semejante a la propiedad icon de un JLabel.

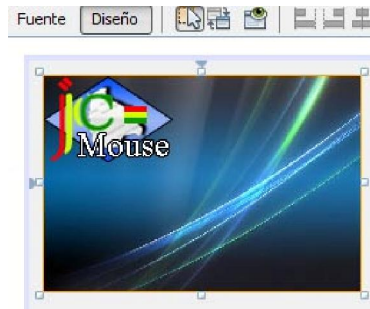


- **setIcon(Icon imagen)**  
Parámetro de entrada: imagen de tipo Icon
- **getIcon()**  
Retorna la imagen que se encuentra en el componente en formato Icon

## Componente Swing creado por jc Mouse

---

La imagen que añada al componente se ajustara al tamaño de este, dependiendo de la memoria que disponga su PC y del tamaño de la imagen que utilice, puede que sienta un poco lento su equipo, recuerde que está trabajando con imágenes directamente desde el editor.



### Logo de Empresa

Como se habrá dado cuenta, el componente tiene un imagen (mi logo) en la parte superior izquierda del componente, este sirve para agregar una imagen independiente de la imagen de fondo del componente. La propiedad para añadir una imagen es "iconLogo", el funcionamiento de este componente es igual a la propiedad "icon".



- **setIconLogo(Icon imagen)**

Parámetro de entrada: imagen de tipo Icon

- **geticonLogo()**

Retorna la imagen almacenada como "logo"

Esta imagen a diferencia de la imagen agregada como fondo, puede ubicarse en cualquier parte del JPanel, para esto cuenta con la propiedad "posicionLogo"



- **setposicionLogo(Point XY)**

Coordenadas de la esquina superior izquierda de la imagen utiliza como "logo"

Parámetro de entrada: XY de tipo Point

- **getposicionLogo()**

Retorna las coordenadas de la imagen utilizada como "logo"

## Componente Swing creado por jc Mouse

Puede ocurrir el caso que no necesite utilizar una segunda imagen en el componente, por este motivo, existe la propiedad “visibleLogo”, esta propiedad que es un Boolean, sirve para ocultar o mostrar la imagen de tipo logo.

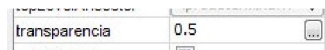


- setVisibleLogo(boolean value)
- getVisibleLogo()

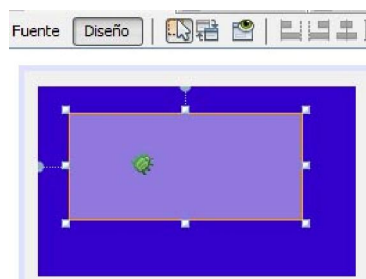


### Transparencia (Modo=2)

Este “modo de uso” le da la propiedad de **transparencia** al componente jpanel, la transparencia esta en un rango de 0 a 1 donde 1 es completamente transparente y 0 es opaco. La propiedad para colocar este valor de transparencia es “transparencia” :-0



- setTransparencia(float value)
- getTransparencia()



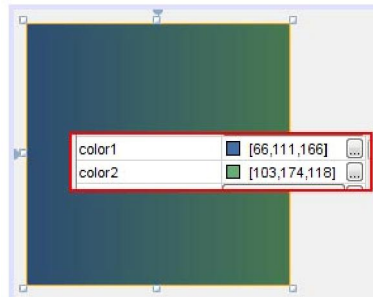
## Componente Swing creado por jc Mouse

---

### Degradado de dos colores (Modo=3)

En este modo de uso puede utilizar dos colores para forma el efecto de degradado dentro el jpanel. Al activar este modo, la imagen de fondo ya no se mostrara en pantalla.

Para colocar los colores se utilizan las propiedades “color1” y “color2”, ambas propiedades reciben como parámetro un valor de tipo Color.



- **setcolor1**(Color value)
- **getcolor1**()
- **getcolor2**(Color value)
- **getcolor2**();

También existe la propiedad “darker”, esta propiedad es un boolean que dependiendo si esta o no activo, oscurece los colores del degradado.



- **setdarker**(boolean value)
- **getdarker**()

Para manipular la posición con la que se pinta el degradado, se utiliza las propiedades “gPosX1”, “gPosY1”, “gPosX2”, “gPosY2” para cada color respectivamente.

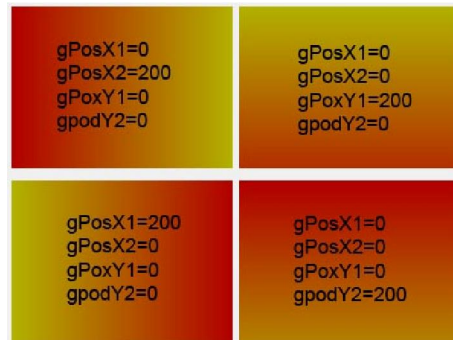
gPosX1	0.0	...
gPosX2	200.0	...
gPosY1	0.0	...
gPosY2	0.0	...

Estas propiedades reciben un valor float que es la posición en la que se quiere pintar un determinado color

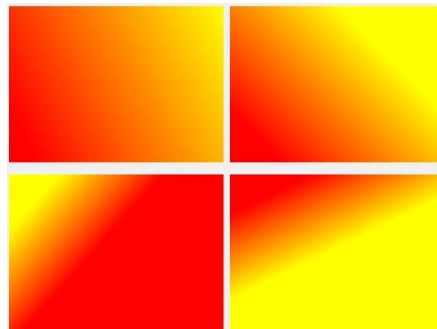
## Componente Swing creado por jc Mouse

---

Ejemplos: Para un `jcMousePanel` de tamaño 200x200 y con colores RGB `Color1=255,0,0` – `Color2=255,255,0` tenemos las siguientes posibilidades.



O porque no también podemos inclinar el degradado y conseguir interesantes efectos.



### Imagen y Transparencia (Modo=4)

No hay mucho que decir sobre este “modo de uso”, es una combinación entre una imagen (`modo=1`) y la transparencia del `jpanel` (`modo=2`) por tanto las propiedades que utiliza son las mismas que estos modos de uso





## Componente Swing creado por jc Mouse

---

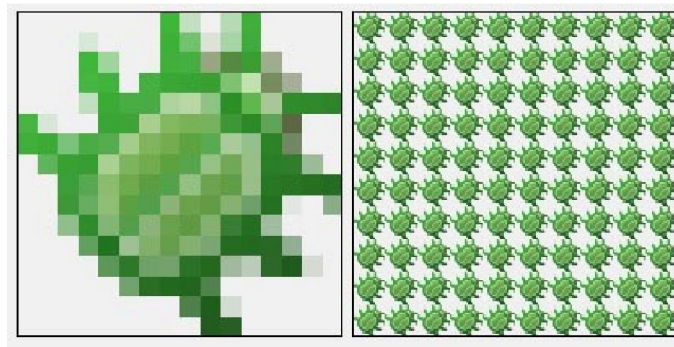
### Degrado y Transparencia (Modo=5)

Al igual que el anterior una combinación de modos (modo 3 y modo 2) con las mismas propiedades que estos modos de uso



### Mosaico de Imágenes.

Al agregar una imagen en el modo 1, esta ocupa todo el espacio disponible por el componente sin fijarse en el tamaño de la imagen añadida, sin embargo, en ocasiones se desea formar un mosaico de la imagen.



Para lograr este efecto, jcMousePanel dispone de una propiedad “mosaico” que activa o desactiva el efecto.

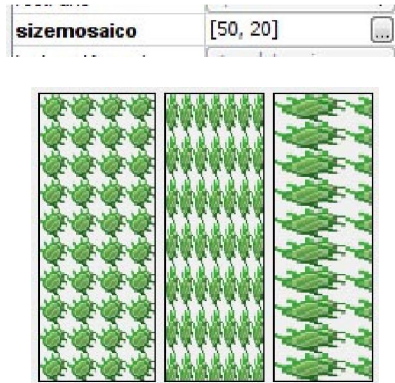


- setmosaico(boolean value)
- getmosaico()

## Componente Swing creado por jc Mouse

---

Además podemos especificar el tamaño del mosaico con la propiedad “sizemosaico” que por defecto tiene un valor de 20x20.



- **setsizemosaico**(Dimension d)
- **getsizemosaico**()

## LEEME

Esta obra está protegida bajo la ley del **PirateRight**, el conocimiento es y debe ser gratuito, este componente swing java se ofrece gratuitamente para su uso y deshuso ;) tanto personal como comercial, al ser un producto personal no se cuenta con los recursos necesarios para su continua mantenimiento (*en otras palabras no tengo plata*) sin embargo se hace un esfuerzo para mantener este y similares componentes actualizados y libre de bugs por lo que un incentivo o un simple gracias valen mucho para el autor y nos animan a seguir trabajando.

Atte Jc-Mouse



*Para los que no entendieron la indirecta más vale una imagen que cien mil palabras*

# ***ENJOY!!!***