Fundamentos Teóricos-Prácticas 1 y 2

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

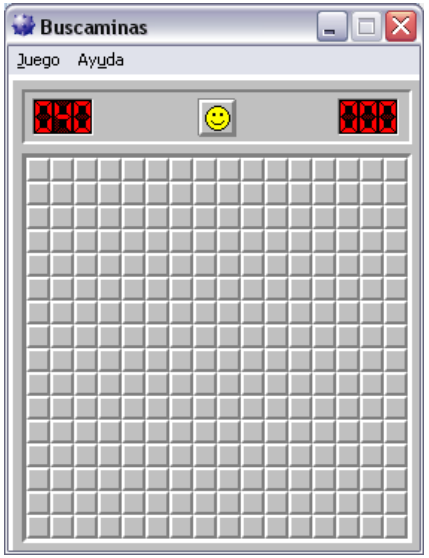
Principales tipos de contenedores

- **Marco (JFrame)**
- Cuadro de Diálogo (JDialog)
- **Panel (JPanel)**
- Panel de Scroll (JScrollPane)
- Panel de Pestañas (JTabbedPane)

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

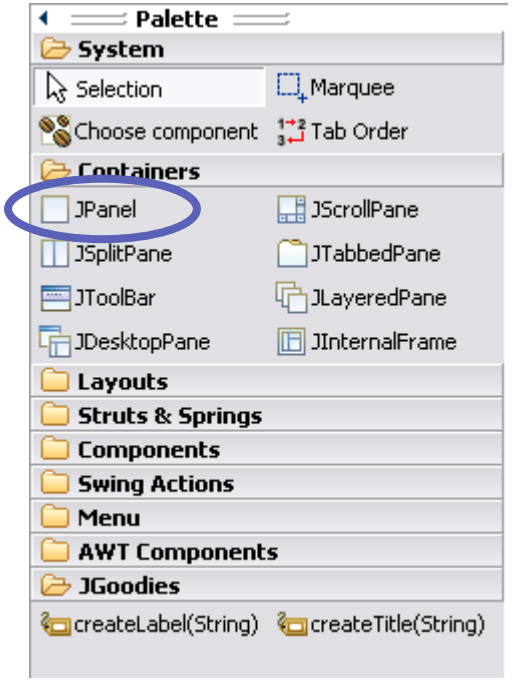
Marco (JFrame)

- Ventana que no está contenida dentro de otra ventana. Representa la ventana principal de una aplicación con IGU.
- Tiene borde, título, menú de control, botones para maximizar y minimizar, controles para redimensionar.
- Puede contener barra de menús.
- Los componentes que formarán parte de la ventana han de situarse sobre un panel, no directamente sobre el JFrame.



Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Panel (JPanel)



- Es un contenedor que agrupa componentes dentro de una ventana u otro panel
- Los 'layouts managers' o gestores de contenido permiten posicionar visualmente los componentes dentro de ellos
- Se suelen emplear también para poner bordes a grupos de componentes (ej. Radio Botones)
- Algunos Métodos:
  - getComponentCount () : número de componentes en el panel
  - getComponents() : devuelve un array con referencias a los componentes que contiene el panel

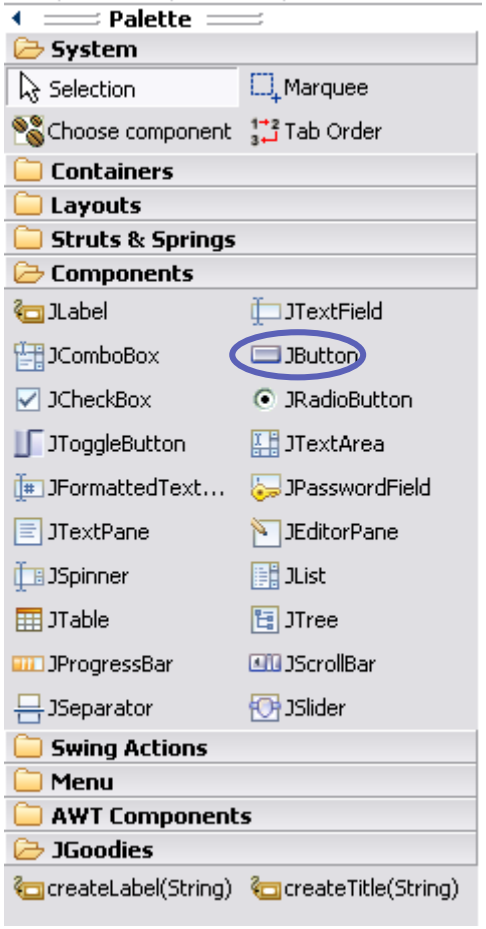
Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Componentes básicos

- Botones
  - Comando (*Command*)
  - Conmutación (*Toggle*)
  - Cajas de chequeo (*Check Boxes*)
  - Radio (*Radio Buttons*)
- Combo Box

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Botones de comando (JButton) (I)



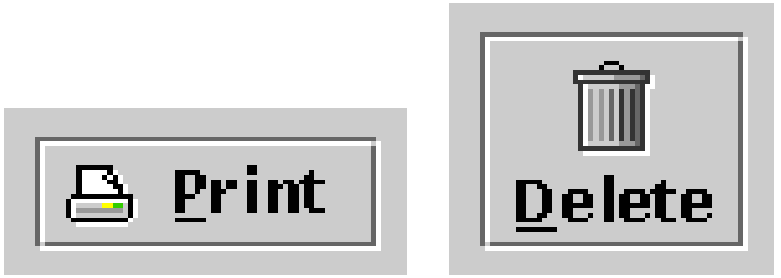
- Botón que puede contener texto, gráficos o ambos.
- Generalmente se emplea una única palabra para identificar la acción que representa el botón.
- Los botones que llevan texto deben tener asignado un mnemónico, **salvo que sean los botones por defecto y de cancelación.**

- Para los que no llevan texto, conviene asociarles tooltips que describan su nombre o función.
  - Proporcionan información (en forma de descripción corta) acerca de un componente cuando el usuario se detiene sobre él
  - Deben estar activos por defecto, pero hay que proporcionar al usuario una manera de desactivarlos, por ejemplo, presentando una opción (*checkbox*) en un cuadro de diálogo de propiedades o preferencias

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Botones de comando (II)

- Los botones que sólo contienen texto, éste debe estar centrado en el botón
- Los botones que contienen texto y gráficos:
  - El texto debe ir colocado después (a la derecha) o debajo del gráfico
  - Siguen siendo necesarios los mnemónicos en el texto (excepto en los botones por defecto y de cancelación)

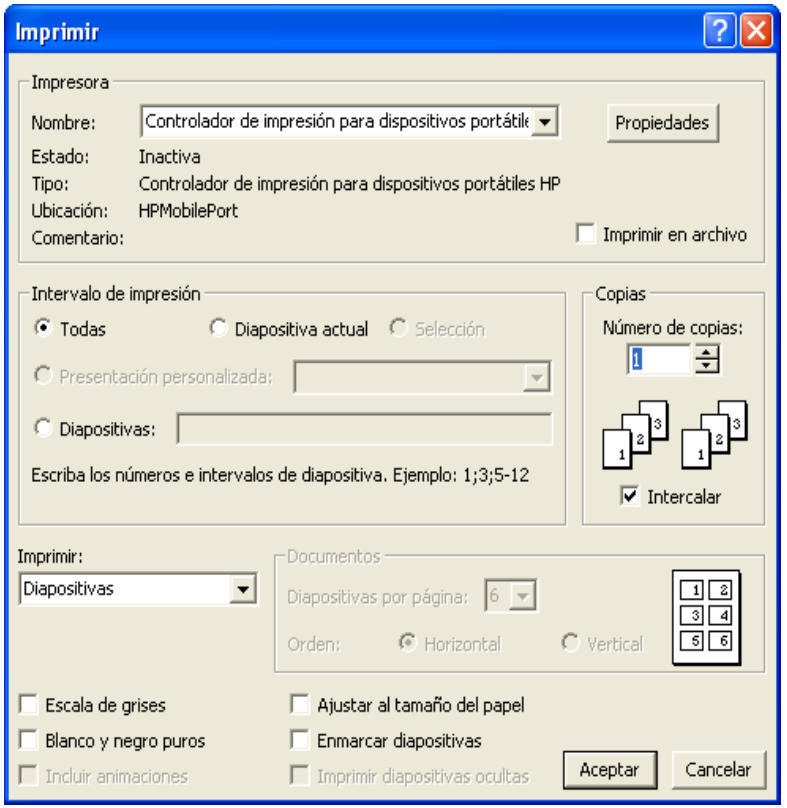


Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Botones de comando (III)

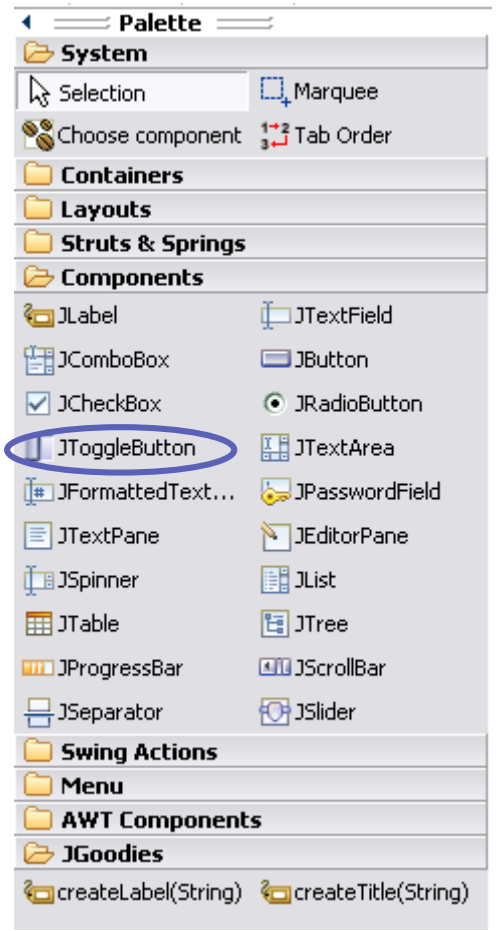
- Si el usuario debe visualizar un cuadro de diálogo **para finalizar la especificación de una acción** iniciada con un botón de comando se añaden... después del texto del botón.

Se necesita más información para completar la ejecución



Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Botones de conmutación (JToggleButton)

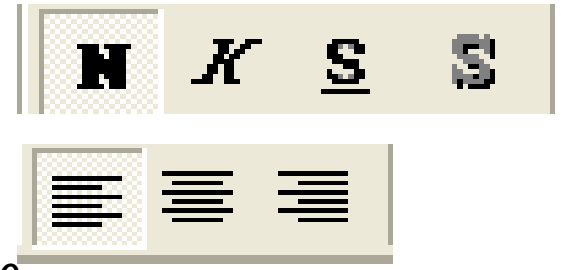


- Un botón de conmutación representa dos estados: on y off.
- Al igual que un botón de comando puede incluir texto y gráficos.
- El gráfico y el texto han de ser los mismos independientemente de que el botón esté on u off.
- Estos botones pueden emplearse para representar opciones independientes (como checkboxes) y opciones exclusivas (como radio botones).

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Botones de conmutación (II)

- Pueden representar:
  - Opciones independientes:
    - Se comportan como *checkboxes*.
  - Opciones exclusivas:
    - Se comportan como *botones de radio*.

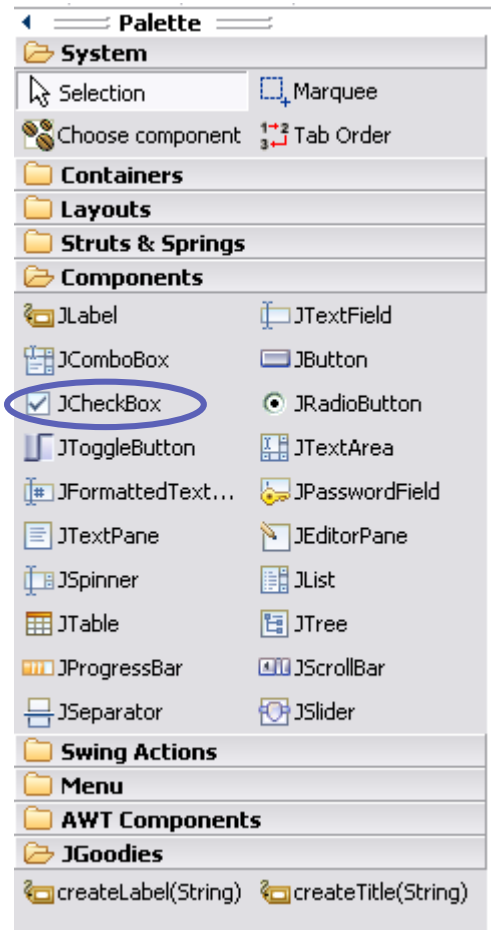


  - En ambos casos los botones de conmutación suelen emplearse en las barras de herramientas (*toolbars) y* los *checkboxes* y *radiobotones* en cuadros de diálogo.

- Propiedad selected:
  - *true* si el botón está ON
  - *false* si el botón está OFF

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

CheckBox (JCheckBox)

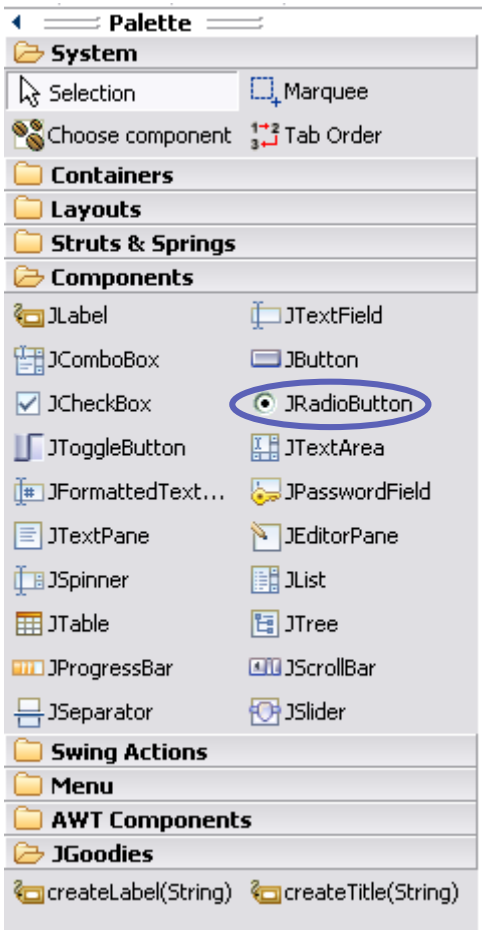


- Es un control que representa también dos estados: on y off

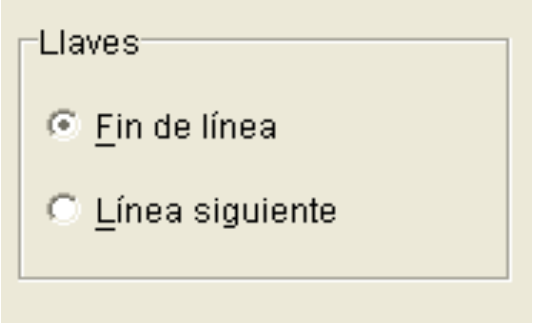
- De la misma forma que en ceso de los botones de conmutación, la propiedad *selected* indica si el componente está o no seleccionado.

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Radio Botones (JRadioButton)



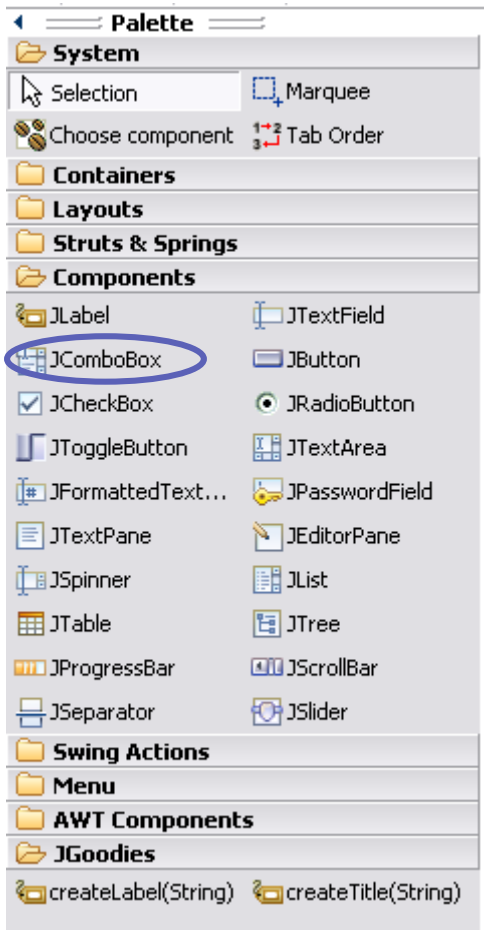
- Permiten seleccionar una única opción dentro de un conjunto de opciones relacionadas
- En java: añadir los radio botones al mismo grupo (ButtonGroup)
- Aunque los radio botones y los botones de conmutación agrupados tienen la misma función conviene emplear los radio botones en cuadros de diálogo.
- Los botones de radio (al igual que los checkboxes) suelen aparecer visualmente agrupados, con una leyenda que indica a que hacen referencia. Una forma para conseguir esta agrupación en Swing es mediante un panel al que se le indica un borde y un título.



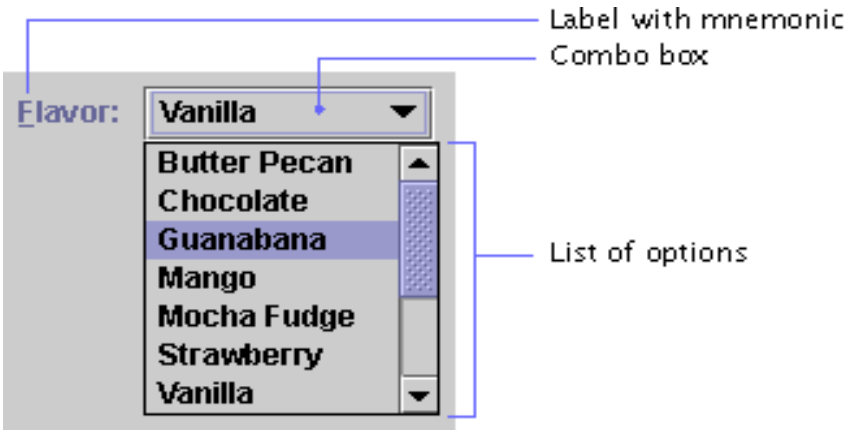
- Propiedades y métodos relevantes son similares al CheckBox.

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Combo Boxes (JComboBox)



- Es un componente con una flecha que al hacer click sobre ella nos permite seleccionar entre un conjunto de opciones mutuamente exclusivas



- Hay que emplear capitalización para el texto de los ítems que aparecen en el combo box.
- Los elementos han de aparecer ordenados.
- Hay que facilitar el acceso por teclado, proporcionando etiquetas con mnemotécnicos.

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Combo Boxes (II)

- **No Editables**
  - A veces llamados List Boxes.
  - Muestran una lista de la que el usuario puede elegir un elemento.
  - Se suelen emplear en vez de un grupo de botones de radio cuando:
    - El espacio en la aplicación es limitado y/o
    - El número de opciones posibles es grande
- **Editables**
  - El usuario puede teclear, seleccionar o editar texto.
  - Se suelen emplear para **ahorrar tiempo** al usuario permitiéndole teclear directamente un valor (además de, por supuesto, seleccionarlo de la lista)

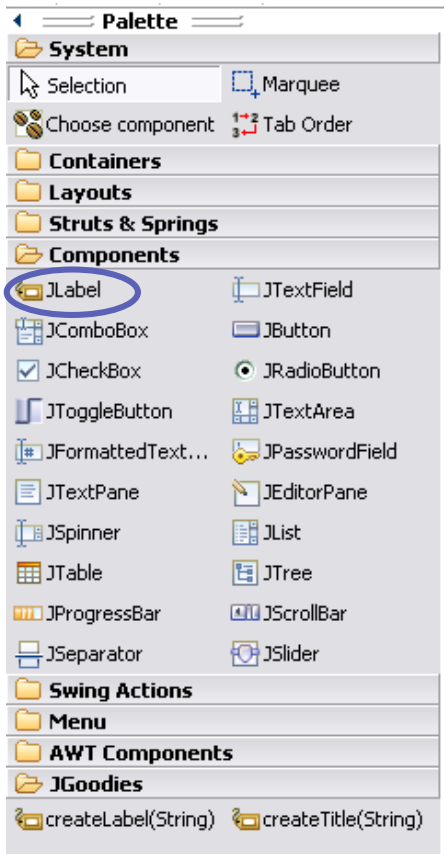
Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Componentes para Texto

- Permiten a los usuarios ver y editar texto en una aplicación
- Algunos componentes:
  - **Etiquetas (*JLabel*)**
  - **Campos de texto (*JText Field*)**
  - **Campos para contraseña (*JPassword Field*)**
  - Areas de texto (*JText Area*)

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

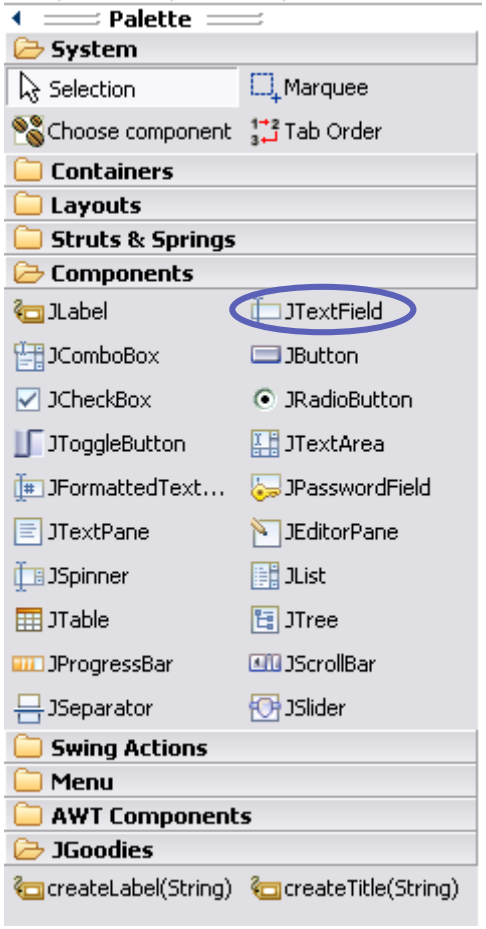
Label (JLabel)



- Muestra texto, gráficos o ambos, pero de sólo lectura
- No puede ser seleccionada por el usuario
- El texto que contienen ha de ser breve y la terminología empleada ha de ser familiar para los usuarios
- Se pueden emplear mnemotécnicos en las etiquetas (*displayedMnemonic*). Cuando se activa el mnemónico, se sitúa el foco en el componente que describe la etiqueta (*labelFor*).
- La etiqueta ha de estar inactiva cuando el componente que describe esté inactivo
- Las etiquetas siempre deben ir antes o encima del componente que describen. Para los lenguajes que leen de izquierda a derecha, antes es a la izquierda del componente.
- Hay que emplear la capitalización en el texto de la etiqueta y colocar : al final del texto.
- Tiene dos funciones en una aplicación:
  - Identificar componentes
  - Comunicar el estado o dar instrucciones a los usuarios

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

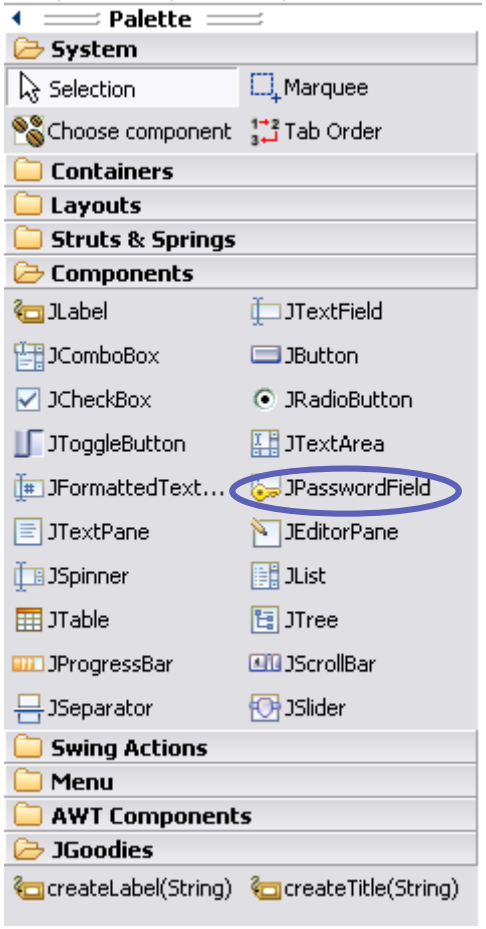
Text Field (JTextField)



- Muestra una línea de texto. Puede ser
  - Editable: los usuarios pueden editar o escribir una línea de texto simple.
  - No editable. Los usuarios pueden seleccionar y copiar el texto, pero no pueden cambiarlo. El texto únicamente puede ser modificado por la aplicación.
- Para asociarle un mnemónico debe asociársele una etiqueta.
- Realizar acciones cuando el usuario:
  - Teclee enter
  - Mueva el foco fuera de este campo
  - ..

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente

Password Field (JPasswordField)



- Es un Text Field editable que muestra unos caracteres enmascarados en lugar de los caracteres que teclea el usuario.
- Proporciona algunas de las capacidades de edición de un Text Field pero no las operaciones de cortar y copiar.

Ana Belén Martínez Prieto
Mª del Carmen Suárez Torrente