

Programació Gràfica amb SWING

Fins ara, tots els nostres programes interactuaven amb la consola, bé fora la del nostre entorn de desenvolupament o la del sistema, la qual cosa resulta tediosa i avorrida, en este tema comencem a programar d'una manera més gràfica.

Java proporciona una sèrie de **classes, mètodes i ferramentes per poder** crear finestres, botons, menús i aconseguir crear interfícies gràfiques mitjanament series. Tots estos elements els trobem al paquet **SWING**.

SWING és una biblioteca gràfica de Java. Inclou **widgets** per a Interfícies Gràfiques d'Usuari o **GUI (Graphical UserInterface)**.

Una interfície d'usuari (**GUI**) és el que permet a un usuari comunicar-se amb un programa a través de components visuals, com caixes de text, botons, menús, menús desplegable, taules, etc.

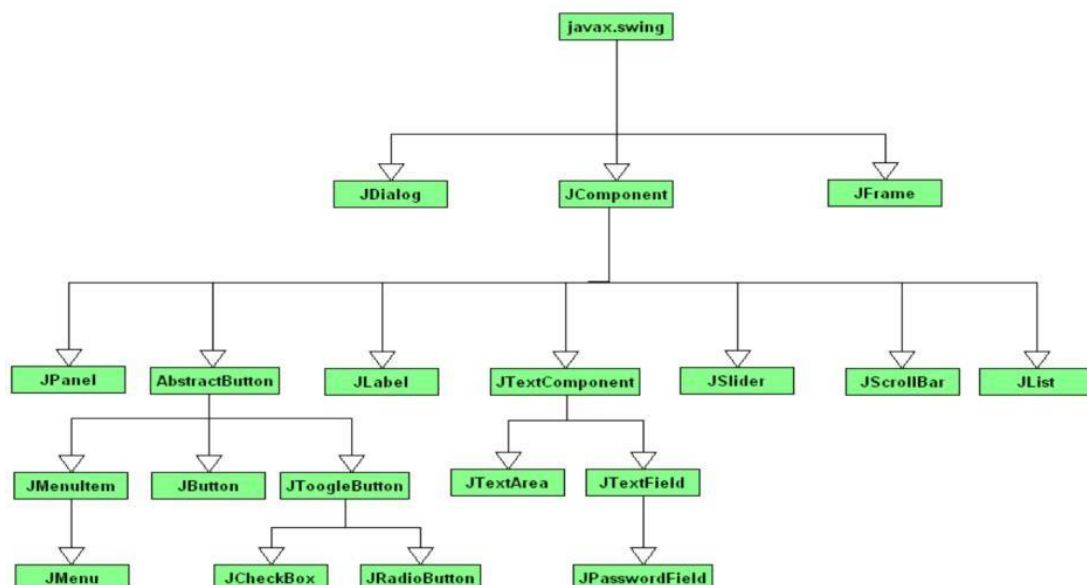
SWING està basat en la **Abstract Window Toolkit (AWT)**, o "kit de herramientas de ventana abstracta", és un kit de gràfics, finestres, interfícies d'usuari, i sistema de finestres independent de la plataforma original de Java.

Classes del paquet SWING

Per a utilitzar les biblioteques que ens proporciona el paquet SWING, hem d'importar-lo: `import javax.swing.*;`

Si accedim a l'API de JAVA (<https://docs.oracle.com/en/java/javase/21/docs/api/index.html>) i busquem dins del mòdul **java.desktop** el **paquet javax.swing**, trobarem que moltes de les classes que ens proporciona comencen per J, com **JButton**, **JLabel**, **JPanel**, **JMenu**, etc., estes classes ens permetran, **mitjançant herència**, crear els diferents components de les interfícies gràfiques.

La jerarquia de classes del paquet SWING és la següent:



- **JComponent**: Superclasse de tots els components de SWING. Tots els components de l'aplicació hereten de JComponent (`javax.swing.JComponent`)
- Cal utilitzar en totes les classes **import** `javax.swing.*`; i **import** `java.awt.*`; que són els paquets que contenen totes les classes i objectes gràfics de java.

Components i Contenidors

Un component és un **objecte** que representa un element de la interfície gràfica i es poden dividir **en dos grups**:

- ▢ **Contenidors**: Son components que poden contindre altres components. Trobem dos tipus:
 - **D'alt nivell**: Ens proporcionen un espai dins la pantalla per a tots els elements de la interfície gràfica (per exemple **JFrame** i **JDialog**,).
 - **Intermedis**: Ens permeten agrupar i organitzar els components d'una **GUI** dins d'un contenidor **d'alt nivell** (per exemple **JPanel**, **JScrollPane**).
- ▢ **Simple o atòmic**: Són els que no estan formats per altres components, com etiquetes, botons, caixes de text, etc. (per exemple **JLabel**, **JButton**, **TextField**, **TextArea**, **CheckBox**, **RadioButton**, **ToggleButton**, **ComboBox**, **Separator**, **Spinner**, **Slider**, **ProgressBar**)

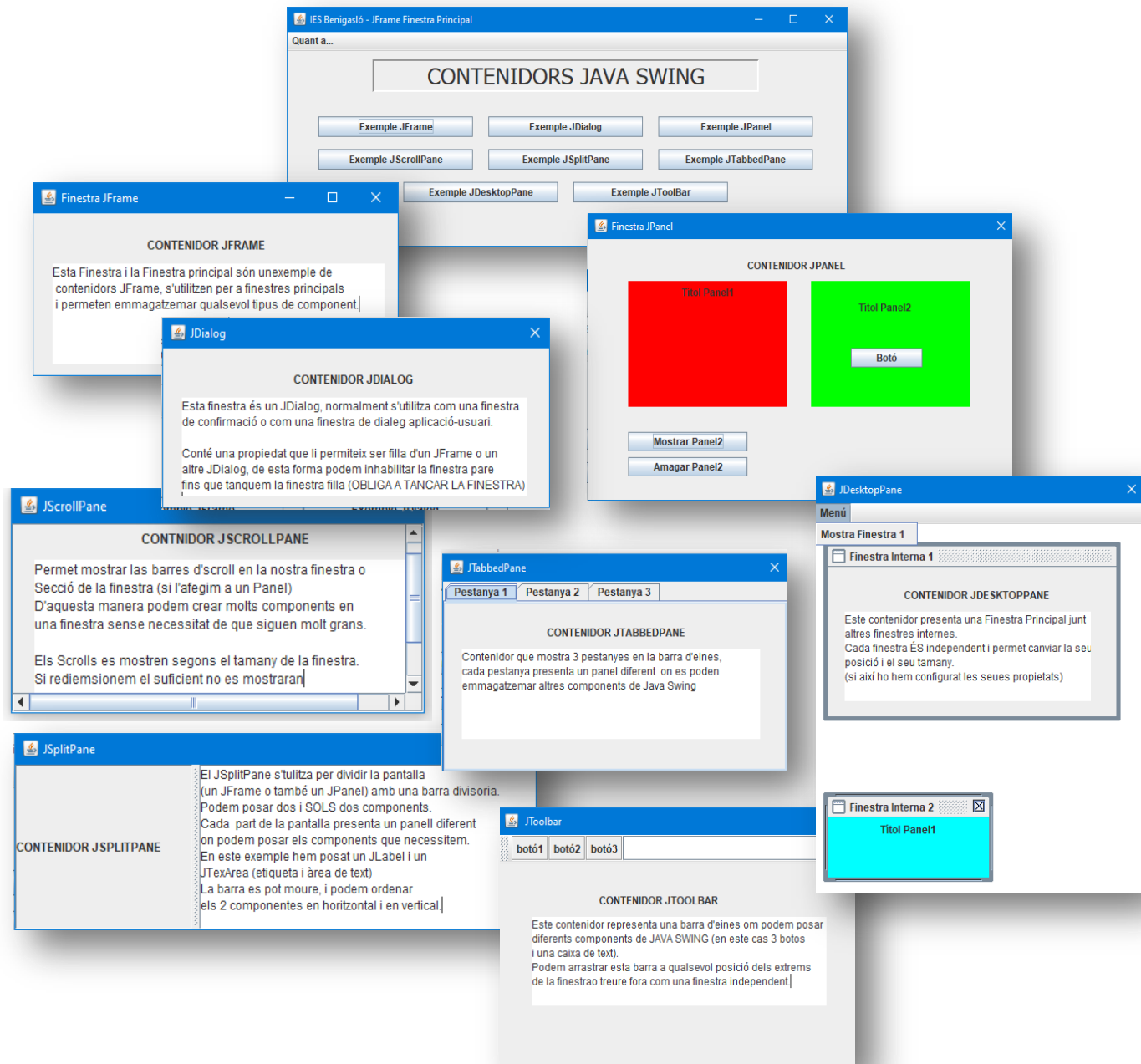
També podem usar classes de suport a l'hora de crear aplicacions com **Graphics**, **Color**, **Font**...

CONTENIDORS

Un contenidor és el tapís on pintarem els nostres components gràfics. A banda dels contenidors principals com **JFrame** i **JDialog** també hi ha altres contenidors inclosos dins dels anteriors (contenidors de components dins de contenidors de components).

- ▢ **JFrame** És la Finestra d'aplicació, el **contenidor principal**
- ▢ **JDialog** Una finestra de tipus **Finestra de diàleg**, també pot ser un contenidor principal.
- ▢ **JPanel** Permet la creació de **panells** independents on **s'emmagatzemen altres components**.
- ▢ **JScrollPane** Permet la vinculació **de barres de desplaçament** en un contenidor.
- ▢ **JSplitPane** Permet la creació d'un **contenidor dividit** en 2 seccions.
- ▢ **JTabbedPane** Permet la **creació de pestanyes**, cada pestanya representa un contenidor independent.
- ▢ **JDesktopPane** - Permet **crear finestres dins d'una finestra** principal
- ▢ **JToolBar** - Permet introduir una **Barra d'eines**

Exemples i Descripcions:

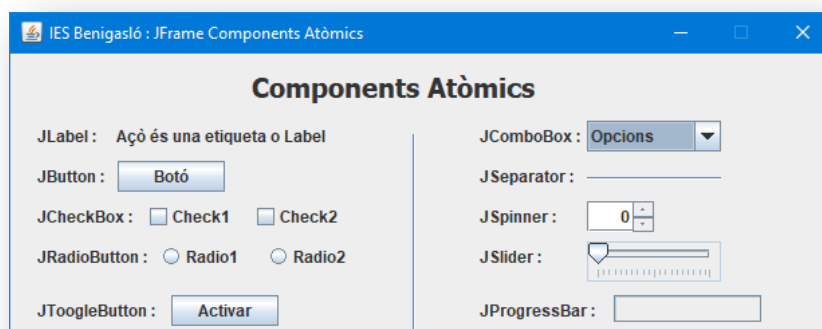


COMPONENTS ATÒMICS

Els components atòmics són els elements que no poden emmagatzemar altres objectes o components gràfics, per exemple, un **JPanel** no és Atòmic, ja que en el podem emmagatzemar **JButtons**, **TextFields** entre d'altres ..

- **JLabel** Permet vincular **Etiquetes**, tant de text com d'imatges.
- **JButton** Permet vincular **Botons** simples.
- **JCheckBox** Són **caselles de verificació**, ideal per a selecció múltiples.
- **JRadioButton** - Permet presentar opcions de selecció similars a les **checkbox**, però sols permet una **selecció única**.
- **JToggleButton** - Botó que al punxar es **quedarà pressionat** fins que s'executi un **altre esdeveniment**.
- **JComboBox** Permet mostrar una **llista d'elements** com un **combo** de selecció.
- **JScrollBar** Permet mostrar una **barra de desplaçament**, molt útil en àrees de text o panells on el contingut és més gran que la mida del component.
- **JSeparator** Permet separar opcions, és una **barra simple**.
- **JSlider** Permet vincular un **botó lliscant** a la nostra finestra.
- **JSpinner** Permet vincular una **caixa de text amb botons integrats** per seleccionar algun valor.
- **JProgressBar** - Estableix una **barra de progrés**.

Exemples

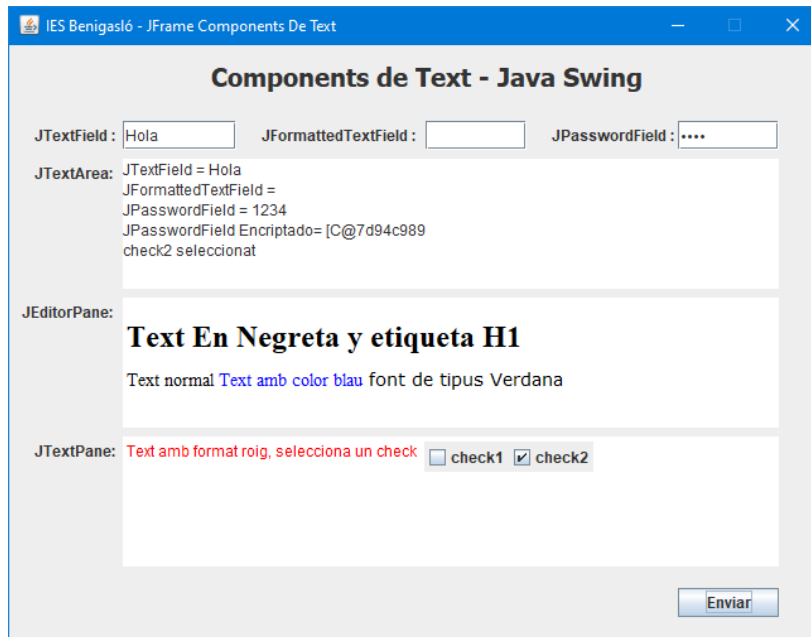


COMPONENTS DE TEXT

Són tots els que ens permeten processar **cadena de text**, sigui com a entrada o eixida d'informació.

- **TextField** Permet introduir un **camp de text simple**.
- **FormattedTextField** Permet introduir un **camp de text amb format**, (si definim que només rep nombres no permetrà lletres ...)
- **PasswordField** Camp de text que **oculta els caràcters** ingressats.
- **TextArea** Permet vincular una **àrea de text** on l'usuari puga **introduir informació** o simplement per presentar cadenes de text.
- **EditorPane** Més complexa que l'anterior. Admet text amb format. Permet combinar fonts, colors i imatges.
- **TextPane** El **més complet**, conté les funcionalitats de l'anterior, amb més possibilitats de formats (fons, colors, etc.), i també admet icones de la classe **ICON** o **qualsevol component** de JAVA com **JButton**, **JLabel**, **JCheckBox**, etc.

Exemples

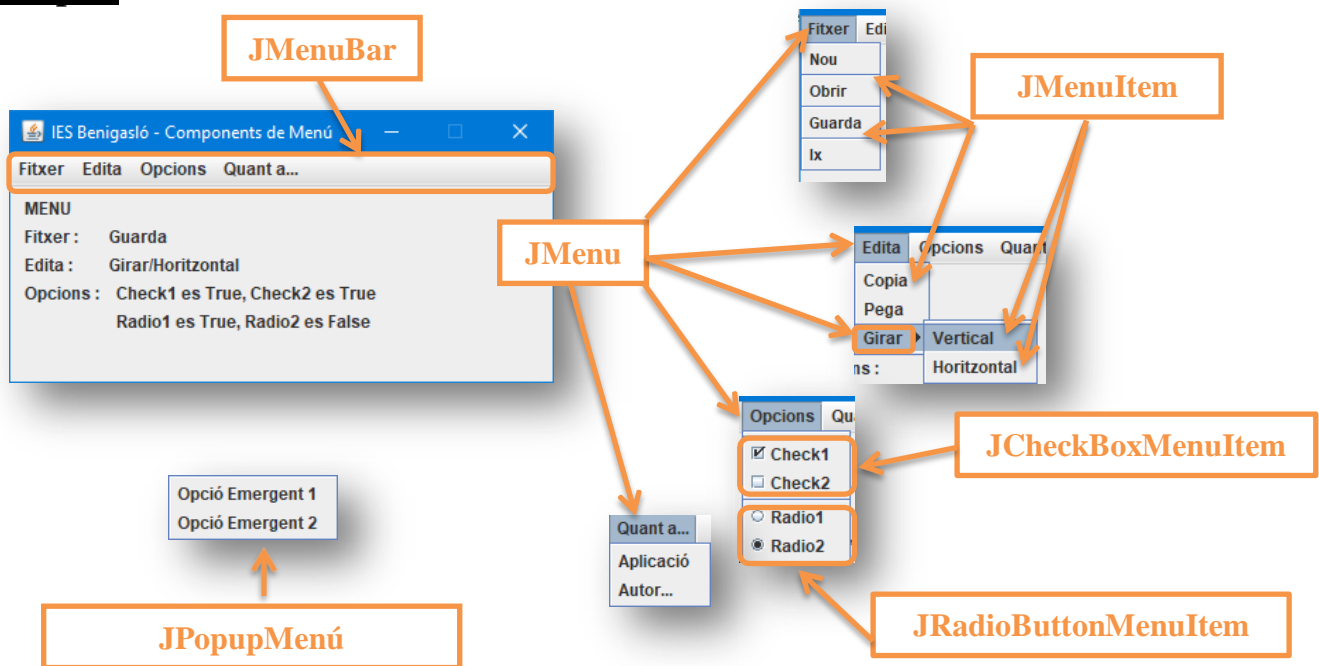


COMPONENTS DE MENÚ

Estos componentes permiten vincular opciones de menú en las nuestras ventanas, tipos menú principal, como ara el conegut Inici, Arxiu, Edició etc.

- **JMenuBar** Permet crear una barra de menús, on situarem els elements **JMenu**.
- **JMenu** Permet vincular botons o enllaços que a ser premuts despleguen un menú principal.
- **JMenuItem** Són els ítems o les opcions de menú.
- **JCheckBoxMenuItem** Element de menú com opcions de checkbox.
- **JRadioButtonMenuItem** Element de menú com botó de selecció.
- **JPopupMenu** Opcions de menú emergents. (El típic amb el botó dret)

Exemples



COMPONENTS AVANÇATS

Estos components compleixen funcions més enfocades a processos específics i complexos, com ara obtenir gran quantitat d'informació d'una base de dades, treball amb nodes, colors, etc..

- **JTable** Permet vincular una taula de dades amb les seves respectives files i columnes.
- **JTree** Carrega un arbre on s'estableix certa jerarquia visual, tipus directori.
- **JList** Permet carregar una llista d'elements, depenent de les propietats pot tenir una llista de selecció múltiple.
- **JFileChooser** És un component que permet la recerca i selecció de fitxers entre d'altres.
- **JColorChooser** Component que permet carregar un panell selector de color
- **JOptionPane** No és una mica complex sinó més un component independent que permet mostrar un quadre de diàleg personalitzable.