# Capítulo 3. Lenguaje de Programación Orientada a Objetos (JAVA)

## Clases, Herencia, Eventos.

Creando una ventana en Java, heredando la clase JFrame

```
import javax.swing.JFrame;
      // Con el extends heredamos la clase JFrame a mi clase MiPrograma
public class MiPrograma extends JFrame
{
      public MiPrograma(){
            super();
      }
      public MiPrograma (String title){
      }

      public static void main (java.lang.String args[]){
            JFrame MiVentana = new JFrame("MI PRIMER VENTANA" );
      //llamamos a la clase y creamos un objeto llamado MiVentana
            MiVentana.setVisible(true);//le decimos al compilador que
      queremos que se vea la ventana
            MiVentana.setSize(600,800);//le damos el tamaño deseado a
      nuestra ventana
            MiVentana.setDefaultCloseOperation(EXIT_ON_CLOSE);//le
      decimos que al dar clic en la X se cierre nuestra ventana
      }
}
```

Ahora agregamos elementos a nuestra clase.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class MenuFrame extends JFrame implements ActionListener {

      /* ------ FIELDS ------ */

      // Menues

      /** File menu. */
      private JMenu fileMenu;
      private JMenu colourMenu;
      private JMenu formatMenu;
      private JMenu editMenu;

      // Other components
```

```
   /** Text Area. */
   private JTextArea textArea;
   /** Array of colour radio button menu items. */
   private JRadioButtonMenuItem colourItems[];
   /** Colours avialble for use with colour radio button menu items. */
   private String colours[]   = {"Blue","Red","Yellow"};
   /** Label for colours group of radio button menu items. */
   private ButtonGroup coloursButtonGroup;

/* ------------------------------------------------- */
   /*                                                 */
   /*                  CONSTRUCTORS                   */
   /*                                                 */
   /* ------------------------------------------------- */

   public MenuFrame() {
   super("JMenu Example");

 // Content pane
   Container container = getContentPane();
   //container.setBackground(Color.pink);

 // Create menus
 createFileMenu();
 createFormatMenu();
 createEditMenu();

 // Create menu bar
 JMenuBar bar = new JMenuBar();
 setJMenuBar(bar);
 bar.add(fileMenu);
 bar.add(colourMenu);

 bar.add(editMenu);
 // Add text area
 textArea = new JTextArea(40, 10);
 textArea.setEditable(false);
   container.add(new JScrollPane(textArea),BorderLayout.CENTER);
 }

   /* CREATE FILE MENU */

   /** Creates file menu */

   private void createFileMenu() {
   // Create file menu
 fileMenu = new JMenu("File");
 fileMenu.setMnemonic('F');

 // Create file menu items
 JMenuItem aboutItem = new JMenuItem("About ...");
 aboutItem.setMnemonic('A');
```

```
aboutItem.setEnabled(true);
aboutItem.addActionListener(this);
JMenuItem exitItem = new JMenuItem("Exit");
exitItem.setMnemonic('x');
exitItem.setEnabled(true);
exitItem.addActionListener(this);

// Add to menu
fileMenu.add(aboutItem);
fileMenu.addSeparator();
fileMenu.add(exitItem);
}

   /* CREATE COLOUR MENU */

   /** Creates colour menu comprising radio menu buttons. */

   private void createColourMenu() {
   // Create colour menu
colourMenu = new JMenu("Colour");
fileMenu.setMnemonic('C');

// Create colour menu radio button items
colourItems     = new JRadioButtonMenuItem[colours.length];
coloursButtonGroup = new ButtonGroup();
for (int index=0;index < colours.length;index++) {
   colourItems[index] = new JRadioButtonMenuItem(colours[index]);
   colourItems[index].addActionListener(this);
   colourMenu.add(colourItems[index]);
   coloursButtonGroup.add(colourItems[index]);
   }

// Select first colour button
colourItems[0].setSelected(true);
}


   /* CREATE FORMAT MENU */

   /** Creates format menu */

   private void createFormatMenu() {
   // Create file menu
formatMenu = new JMenu("Format");
formatMenu.setMnemonic('r');

// Create file menu items
createColourMenu();

// Add items to menu
formatMenu.add(colourMenu);
}
   private void createEditMenu(){
```

```
editMenu = new JMenu("Edit");
editMenu.setMnemonic('e');

JMenuItem activarItem = new JMenuItem("Activar Edición");
activarItem.setEnabled(true);
   activarItem.addActionListener(this);

editMenu.add(activarItem);
}
   /* ------ METHODS ------ */

   /* ACTION PERFORMED */

   /** Item handlers.
   @param event the triggered event. */

   public void actionPerformed(ActionEvent event) {
   if (event.getActionCommand().equals("About ...")) about();
else if (event.getActionCommand().equals("Exit")) exitSystem();
else if (event.getActionCommand().equals("Blue")) changeColour("Blue");
else if (event.getActionCommand().equals("Red")) changeColour("Red");
else if (event.getActionCommand().equals("Yellow"))
                                       changeColour("Yellow");
else if (event.getActionCommand().equals("Activar Edición"))
                                       activarEdicionAreaTexto();
else JOptionPane.showMessageDialog(this,"Error in event handler",
                              "Error: ",JOptionPane.ERROR_MESSAGE);
}

   /* ABOUT */

   /** Outputs JOption pane if about menu item selected. */

   private void about() {
textArea.append("Code example illustrating use of JMenus\n");
}

   /* EXIT */

   /** Exits system */

   private void exitSystem() {
   System.exit(0);
}

   /* CHANGE COLOUR */

   /** Changes colour */

   private void changeColour(String newColour) {
textArea.append("Change colour to " + newColour+ "\n");
}
```

```
      /* EDIT: ACTIVAR EDICION AREA DE TEXTO */
      private void activarEdicionAreaTexto(){
    textArea.setEditable(true);
    }
      }

public class MenuExampleApp1 {

      /* ------ FIELDS ------ */

      // No fields

      /* ------ METHODS ------ */

      /** Main method (to start the "ball rolling"). */

      public static void main(String Args[]) {
      // Creat menu frame
      MenuFrame menuFrame = new MenuFrame();
    menuFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    // Set frame size
    menuFrame.setSize(500,400);

    // Display frame
    menuFrame.setVisible(true);
    }
}
```

Actividad: Incluir una nueva clase u objeto que genere texto y se pueda incluir en una nueva área de texto.

---

Cargar una Imagen con java.

```
import java.awt.*;
import java.awt.event.*;
import java.awt.image.*;
import java.io.*;
import javax.imageio.*;
import javax.swing.*;

/**
 * This class demonstrates how to load an Image from an external file
 */
public class LoadImageApp extends Component {

      BufferedImage[] img = new BufferedImage[5];

      public void paint(Graphics g) {
      g.drawImage(img[0], 0, 0, null);
      g.drawImage(img[1], 0, 0, null);
      g.drawImage(img[0], 110,110, null);
```

```
    }

    public LoadImageApp() {
    try {
           img[1] = ImageIO.read(new File("strawberry.jpg"));
           img[0] = ImageIO.read(new File("manzana.jpg"));
    } catch (IOException e) {
    }

    }

    public Dimension getPreferredSize() {
    if (img == null) {
           return new Dimension(100,100);
    } else {
           return new Dimension(img[0].getWidth(null), img[0].getHeight(null));
    }
    }

    public static void main(String[] args) {

    JFrame f = new JFrame("Load Image Sample");

    f.addWindowListener(new WindowAdapter(){
           public void windowClosing(WindowEvent e) {
                  System.exit(0);
           }
           });
  LoadImageApp X = new LoadImageApp();
    f.add(X);
    f.pack();
    f.setVisible(true);
  X.img[2] = X.img[0];
  try {
        Thread.sleep(2000L);      // one second
  }catch (Exception e) {}
  X.img[0] = X.img[1];
  X.img[1]=X.img[2];
  f.repaint();
    }
}
```

Actividad:

- Escalar la imagen y ponerla en diferente posición con la acción del mouse.
- Cargar una imágen animada (GIF animada).