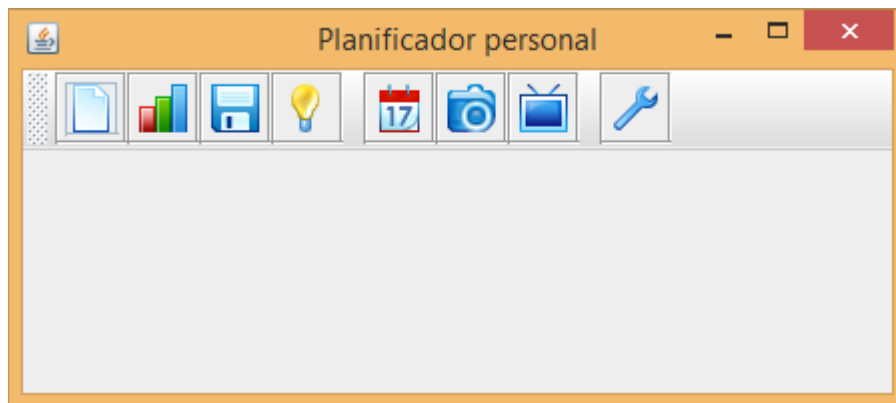


Actividades de Interfaces gráficas de usuario 3

Controles avanzados

1.- Descarga del Moodle la colección de iconos "Iconos toolbar 24x24" y crea una ventana con una barra de herramientas idéntica a la siguiente:



```
public class BarraHerramientas extends JFrame {
    public BarraHerramientas() {
        super("Planificador personal");
        setSize(450, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        BorderLayout borde = new BorderLayout();
        setLayout(borde);

        // Construimos cada botón de la barra
        Imagen img1 = new Imagen("iconos/toolbar/1.png");
        JButton btnBoton1 = new JButton(img1);
        btnBoton1.setToolTipText("Nueva tarea");
        Imagen img2 = new Imagen("iconos/toolbar/11.png");
        JButton btnBoton2 = new JButton(img2);
        Imagen img3 = new Imagen("iconos/toolbar/22.png");
        JButton btnBoton3 = new JButton(img3);
        Imagen img4 = new Imagen("iconos/toolbar/3.png");
        JButton btnBoton4 = new JButton(img4);
        Imagen img5 = new Imagen("iconos/toolbar/44.png");
        JButton btnBoton5 = new JButton(img5);
        Imagen img6 = new Imagen("iconos/toolbar/66.png");
        JButton btnBoton6 = new JButton(img6);
        Imagen img7 = new Imagen("iconos/toolbar/67.png");
        JButton btnBoton7 = new JButton(img7);
        Imagen img8 = new Imagen("iconos/toolbar/73.png");
        JButton btnBoton8 = new JButton(img8);

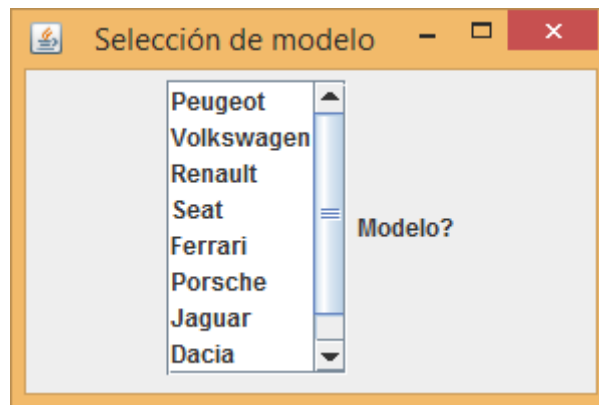
        // Creamos la barra y le añadimos los botones
        JToolBar barra = new JToolBar();
        barra.add(btnBoton1);
        barra.add(btnBoton2);
        barra.add(btnBoton3);
        barra.add(btnBoton4);
        barra.addSeparator();
    }
}
```

```
        barra.add(btnBoton5);
        barra.add(btnBoton6);
        barra.add(btnBoton7);
        barra.addSeparator();
        barra.add(btnBoton8);

        // Agregar la barra de herramientas y el text area al marco.
        add("North", barra);
        setVisible(true);
    }

    public static void main(String[] arguments) {
        BarraHerramientas herramientas = new BarraHerramientas();
    }
}
```

2.- Construye la siguiente interfaz de usuario:



```
public class Lista extends JFrame {

    public Lista() {
        super("Selección de modelo");
        setSize(300, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());

        String categorias[] = { "Peugeot", "Volkswagen", "Renault", "Seat",
                                "Ferrari", "Porsche", "Jaguar", "Dacia", "Volvo" };

        JList lista = new JList(categorias);
        JScrollPane panelScroll = new JScrollPane(lista);

        JLabel lblEtiqueta = new JLabel("Modelo?");

        add(panelScroll);
        add(lblEtiqueta);

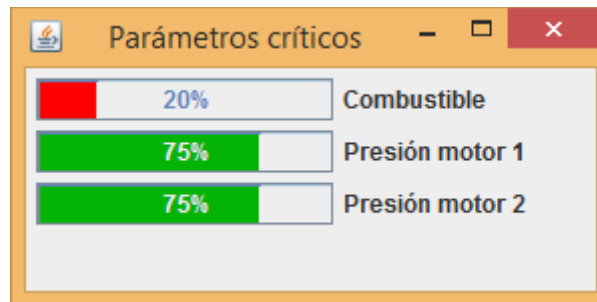
        setVisible(true);
    }

    public static void main(String[] args) {
        new Lista();
    }
}
```

3.- Diseña una interfaz de usuario que permita representar varios parámetros de un avión mediante barras de progreso. Puedes utilizar:

```
setLayout(new FlowLayout(FlowLayout.LEFT));
```

si quieres que los controles queden alineados a la izquierda.



```
public class BarraProgreso extends JFrame {

    public BarraProgreso() {
        super("Parámetros críticos");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 150);
        setLayout(new FlowLayout(FlowLayout.LEFT));

        JProgressBar prgCombustible = new JProgressBar(0, 2000);
        prgCombustible.setValue(400);
        prgCombustible.setStringPainted(true);
        prgCombustible.setForeground(Color.RED);

        JProgressBar prgPresionMotor1 = new JProgressBar(0, 600);
        prgPresionMotor1.setValue(450);
        prgPresionMotor1.setStringPainted(true);
        prgPresionMotor1.setForeground(new Color(0x04B404));

        JProgressBar prgPresionMotor2 = new JProgressBar(0, 600);
        prgPresionMotor2.setValue(450);
        prgPresionMotor2.setStringPainted(true);
        prgPresionMotor2.setForeground(new Color(0x04B404));

        JLabel lblCombustible = new JLabel("Combustible");
        JLabel lblPresionMotor1 = new JLabel("Presión motor 1");
        JLabel lblPresionMotor2 = new JLabel("Presión motor 2");

        add(prgCombustible);
        add(lblCombustible);

        add(prgPresionMotor1);
        add(lblPresionMotor1);

        add(prgPresionMotor2);
        add(lblPresionMotor2);

        setVisible(true);
    }

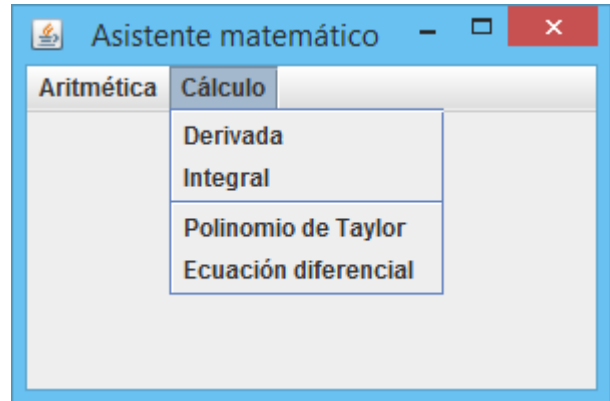
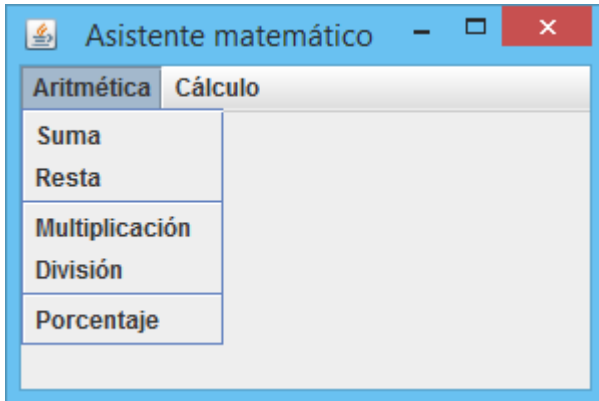
    public static void main(String[] arguments) {
        BarraProgreso marco = new BarraProgreso();
    }
}
```

```

    }
}

```

4.- Crea una ventana con los siguientes menús:



```

class Menu extends JFrame {
    public Menu() {
        super("Asistente matemático");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 200);

        setLayout(new FlowLayout());

        // Creamos los elementos del menú Aritmética.
        JMenuItem itemSuma = new JMenuItem("Suma");
        JMenuItem itemResta = new JMenuItem("Resta");
        JMenuItem itemMultiplicar = new JMenuItem("Multiplicación");
        JMenuItem itemDividir = new JMenuItem("División");
        JMenuItem itemPorcentaje = new JMenuItem("Porcentaje");

        // Creamos el menú Aritmética y le añadimos sus elementos.
        JMenu menuAritmetica = new JMenu("Aritmética");
        menuAritmetica.add(itemSuma);
        menuAritmetica.add(itemResta);
        menuAritmetica.addSeparator();
        menuAritmetica.add(itemMultiplicar);
        menuAritmetica.add(itemDividir);
        menuAritmetica.addSeparator();
        menuAritmetica.add(itemPorcentaje);

        // Creamos los elementos del menú Cálculo.
        JMenuItem itemDerivada = new JMenuItem("Derivada");
        JMenuItem itemIntegral = new JMenuItem("Integral");
        JMenuItem itemTaylor = new JMenuItem("Polinomio de Taylor");
        JMenuItem itemDiferencial = new JMenuItem("Ecuación diferencial");

        // Creamos el menú Cálculo y le añadimos sus elementos.
        JMenu menuCalculo = new JMenu("Cálculo");
        menuCalculo.add(itemDerivada);
        menuCalculo.add(itemIntegral);
        menuCalculo.addSeparator(); // Línea horizontal para separar
        menuCalculo.add(itemTaylor);
        menuCalculo.add(itemDiferencial);
    }
}

```

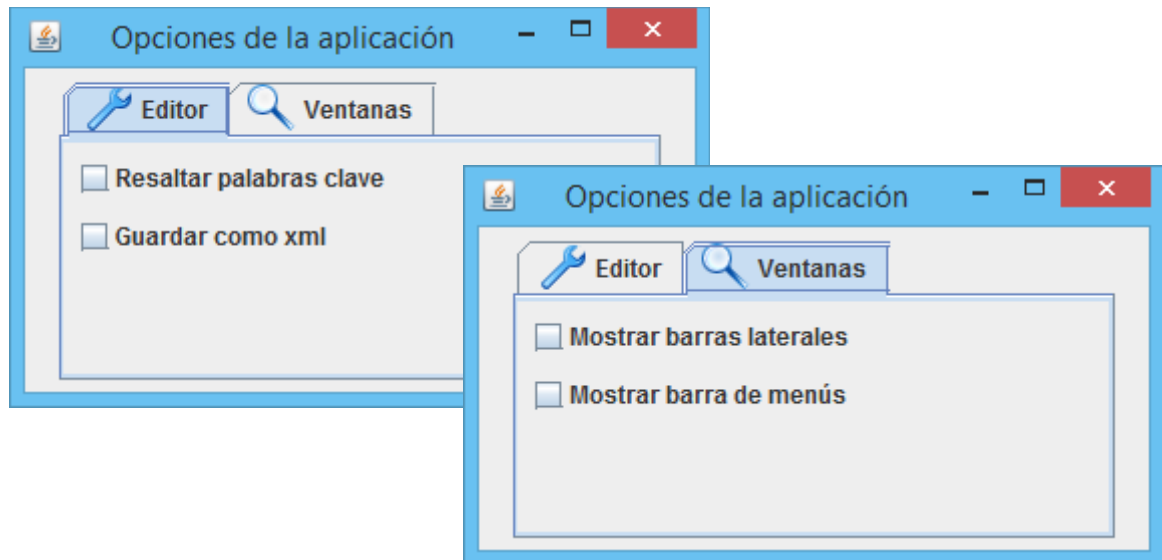
```
// Creamos la barra de menú y le añadimos el menú Archivo.
JMenuBar barraMenu = new JMenuBar();
barraMenu.add(menuAritmetica);
barraMenu.add(menuCalculo);

// Establecemos barraMenu como barra de nuestra ventana.
setJMenuBar(barraMenu);

setVisible(true);
}

public static void main(String[] args) {
    new Menu();
}
}
```

5.- Utiliza la colección "Iconos toolbar 24x24" del Moodle para realizar la siguiente interfaz de usuario:



```
public class PanelesCarpetas extends JFrame {
    public PanelesCarpetas() {

        super("Opciones de la aplicación");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(350, 200);

        setLayout(new FlowLayout());

        JPanel panelEditor = new JPanel();
        panelEditor.setLayout(new FlowLayout(FlowLayout.LEFT));
        ImageIcon iconoEditor = new ImageIcon("iconos/toolbar/73.png");

        JCheckBox chkResaltar = new JCheckBox("Resaltar palabras clave");
        JCheckBox chkGuardar = new JCheckBox("Guardar como xml");
        panelEditor.add(chkResaltar);
        panelEditor.add(chkGuardar);

        JPanel panelVentanas = new JPanel();
        panelVentanas.setLayout(new FlowLayout(FlowLayout.LEFT));
        ImageIcon iconoVentanas = new ImageIcon("iconos/toolbar/65.png");

        JCheckBox chkPanelLateral = new JCheckBox("Mostrar barras laterales");
        JCheckBox chkMenu = new JCheckBox("Mostrar barra de menús");
        panelVentanas.add(chkPanelLateral);
        panelVentanas.add(chkMenu);

        JTabbedPane panelon = new JTabbedPane();
        panelon.addTab("Editor", iconoEditor, panelEditor);
        panelon.addTab("Ventanas", iconoVentanas, panelVentanas);

        panelon.setPreferredSize(new Dimension(300, 150));

        add(panelon);
        setVisible(true);
    }
}
```

8

```
    public static void main(String[] args) {  
        new PanelesCarpetas();  
    }  
}
```