



Centro de Investigación en Computación del IPN

Java - curso básico

Dra. Erandi Castillo Montiel

Ejercicio

Tipo ensalada

☐ Chica (\$35.0)

☐ Mediana (\$55.0)

☐ Grande (\$65.0)

Complemento

☐ Crocante (\$5.0)

☐ Aderezo (\$10.0)

☐ Queso parmesano (\$15.0)

Calcular Limpiar

Subtotal:

Impuesto (%10):

Total:

Salir

Implementación

```
private void calcularBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    double total, subtotal=0;  
    final double impuesto=0.10;  
    double tax;  
    //para los radio button  
    if(chicaRB.isSelected()){  
        subtotal=subtotal+35.0;  
    }else if (medianaRB.isSelected()){  
        subtotal=subtotal+55.0;  
    }else if (grandeRB.isSelected()){  
        subtotal=subtotal+65.0;  
    }  
    //para los check box  
    if(aderezoCB.isSelected()){  
        subtotal=subtotal+5.0;  
    }  
    if(crocanteCB.isSelected()){  
        subtotal=subtotal+10.0;  
    }  
    if(quesoCB.isSelected()){  
        subtotal=subtotal+15.0;  
    }  
    subTotalText.setText(Double.toString(subtotal));  
  
    tax=subtotal*impuesto;  
    total=tax+subtotal;  
    impuestoText.setText(Double.toString(tax));  
    totalText.setText(Double.toString(total));  
}
```

Implementación

```
private void limpiarBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    aderezoCB.setSelected(false);  
    crocanteCB.setSelected(false);  
    quesoCB.setSelected(false);  
    chicaRB.setSelected(false);  
    grandeRB.setSelected(false);  
    medianaRB.setSelected(false);  
    impuestoText.setText(" ");  
    totalText.setText(" ");  
    subTotalText.setText(" "); }  

```

```
private void salirBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    System.exit(0);  
}
```

Tarea



Implementación

Cambiar icono de la pantalla
Agregar fondo a la ventana

```
import javax.swing.ImageIcon;
import javax.swing.JLabel;
import javax.swing.JLayeredPane;
import javax.swing.JPanel;

public class RestauranteUI extends javax.swing.JFrame {

    public RestauranteUI() {
        initComponents();
        setLocationRelativeTo(null);
        setResizable(false);
        setTitle("Restaurante");
        //cambia el icono de la ventana
        this.setIconImage(new ImageIcon(getClass().getResource("/imagenes/icono.png")).getImage());
        ((JPanel) getContentPane()).setOpaque(false);
        //cambia el fondo de la ventana
        ImageIcon uno = new ImageIcon(this.getClass().getResource("/imagenes/restaurante.jpg"));
        JLabel fondo = new JLabel();
        fondo.setIcon(uno);
        getLayeredPane().add(fondo, JLayeredPane.FRAME_CONTENT_LAYER);
        fondo.setBounds(0,0,uno.getIconWidth(),uno.getIconHeight());
    }
}
```

Funcionalidad a los botones

```
private void acercaDeBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    AcercaDeUI aForm = new AcercaDeUI();  
    aForm.setVisible(true);  
    dispose();  
}
```

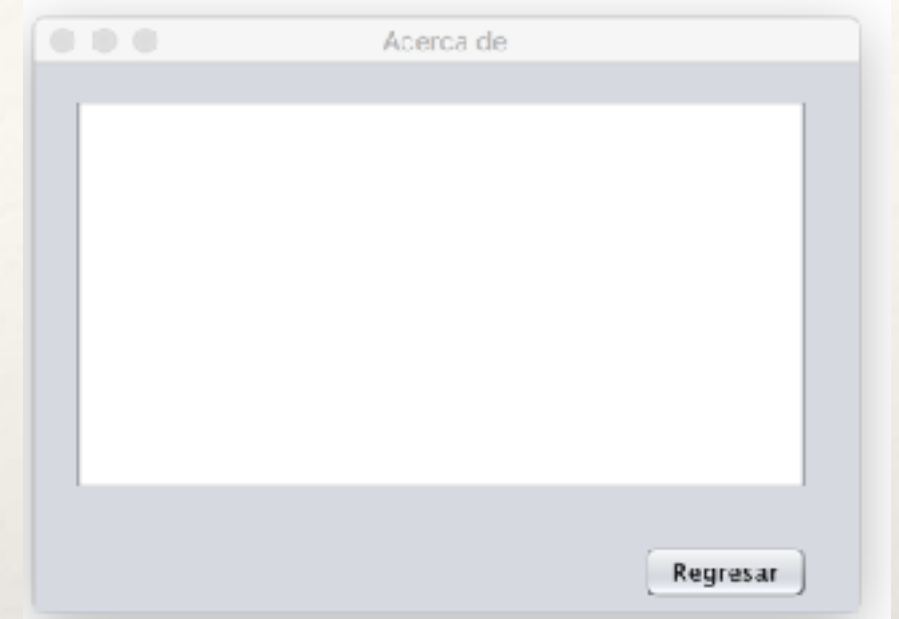
```
private void valorBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    OpcionesUI oForm = new OpcionesUI();  
    oForm.setVisible(true);  
    dispose();  
}
```

```
private void expertosBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    ExpertosUI eForm = new ExpertosUI();  
    eForm.setVisible(true);  
    dispose();  
}
```

```
private void salirBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    dispose();  
}
```

Acerca de

- ❖ Leer de un archivo
- ❖ Cambiar el fondo
- ❖ Funcionalidad al botón regresar



```
public AcercaDeUI() {  
    initComponents();  
    setLocationRelativeTo(null);  
    setResizable(false);  
    setTitle("Restaurante");  
    this.setIconImage(new ImageIcon(getClass().getResource("/imagenes/icono.png")).getImage());  
    ((JPanel) getContentPane()).setOpaque(false);  
    ImageIcon uno = new ImageIcon(this.getClass().getResource("/imagenes/opcionesRestaurante2.jpg"));  
    JLabel fondo = new JLabel();  
    fondo.setIcon(uno);  
    getLayeredPane().add(fondo, JLayeredPane.FRAME_CONTENT_LAYER);  
    fondo.setBounds(0, 0, uno.getIconWidth(), uno.getIconHeight());  
    String text = leerAcercaDe();  
    areaText.setText(text);  
}
```



```

public String leerAcercaDe() {
    File archivo = null;
    FileReader fr = null;
    BufferedReader br = null;
    String result = "";
    try {
        // Apertura del fichero y creacion de BufferedReader para poder
        // hacer una lectura comoda (disponer del metodo readLine()).
        archivo = new File("src/informacionRestaurante/acercaDe.txt");
        fr = new FileReader(archivo);
        br = new BufferedReader(fr);

        // Lectura del fichero
        String linea = "";
        while ((linea = br.readLine()) != null) {
            System.out.println("datos: " + linea);
            result = linea;
        }
        // return linea;
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        // En el finally cerramos el fichero, para asegurarnos
        // que se cierra tanto si todo va bien como si salta
        // una excepcion.
        try {
            if (null != fr) {
                fr.close();
            }
        } catch (Exception e2) {
            e2.printStackTrace();
        }
    }
    return result;
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    RestauranteUI rForm = new RestauranteUI();
    rForm.setVisible(true);
    dispose();
}

```

Consejo de los expertos

- ❖ Leer de dos archivos al accionar el botón buscar
- ❖ Cambiar el fondo



```

public ExpertosUI() {
    initComponents();
    setLocationRelativeTo(null);
    setResizable(false);
    setTitle("Restaurante");
    this.setIconImage(new ImageIcon(getClass().getResource("/imagenes/icono.png")).getImage());
    ((JPanel) getContentPane()).setOpaque(false);
    ImageIcon uno = new ImageIcon(this.getClass().getResource("/imagenes/opcionesRestaurante2.jpg"));
    JLabel fondo = new JLabel();
    fondo.setIcon(uno);
    getLayeredPane().add(fondo, JLayeredPane.FRAME_CONTENT_LAYER);
    fondo.setBounds(0, 0, uno.getIconWidth(), uno.getIconHeight());
}

```

```

private void regresarBtnActionPerformed(java.awt.event.ActionEvent evt) {
    RestauranteUI rForm = new RestauranteUI();
    rForm.setVisible(true);
    dispose();
}

private void buscarBtnActionPerformed(java.awt.event.ActionEvent evt) {
    String nombreArchivo="";
    String text;
    if(saludableRB.isSelected()){
        nombreArchivo="saludable";
    }else{
        nombreArchivo="noSaludable";
    }
    text=leerAcercaDe(nombreArchivo);
    areaText.setText(text);
}

```

```
public String leerAcercaDe(String nombreArchivo){

    File archivo = null;
    FileReader fr = null;
    BufferedReader br = null;
    String result="";
    try {
        archivo = new File ("src/informacionRestaurante/"+nombreArchivo+".txt");
        fr = new FileReader (archivo);
        br = new BufferedReader(fr);

        // Lectura del fichero
        String linea="";
        while((linea=br.readLine())!=null){
            System.out.println("datos: "+linea);
            result=result + linea+"\n";
        }

    }
    catch(Exception e){
        e.printStackTrace();
    }finally{
        try{
            if( null != fr ){
                fr.close();
            }
        }catch (Exception e2){
            e2.printStackTrace();
        }
    }
    return result;
}
```

Calcular valor calórico

❖ Crear una clase Menu



```
public OpcionesUI() {  
    initComponents();  
    setLocationRelativeTo(null);  
    setResizable(false);  
    setTitle("Restaurante");  
    this.setIconImage(new ImageIcon(getClass().getResource("/imagenes/icono.png")).getImage());  
    ((JPanel) getContentPane()).setOpaque(false);  
    ImageIcon uno = new ImageIcon(this.getClass().getResource("/imagenes/opcionesRestaurante2.jpg"));  
    JLabel fondo = new JLabel();  
    fondo.setIcon(uno);  
    getLayeredPane().add(fondo, JLayeredPane.FRAME_CONTENT_LAYER);  
    fondo.setBounds(0, 0, uno.getIconWidth(), uno.getIconHeight());  
}
```



```
private void regresarBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    RestauranteUI rForm = new RestauranteUI();  
    rForm.setVisible(true);  
    dispose();  
}  
  
private void calculaBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    double resultado;  
    Menu client = new Menu(entradaCB.getSelectedIndex(), principalCB.getSelectedIndex(),  
                           postreCB.getSelectedIndex());  
    resultado=client.calculaCalorias();  
  
    valorText.setText(String.valueOf(resultado));  
}
```

```

public class Menu {
    public int entrada;
    public int principal;
    public int postre;

    public Menu (int entrada, int principal, int postre)
    {
        this.entrada=entrada;
        this.principal=principal;
        this.postre=postre;
    }

    public double calculaCalorias(){
        double ent,prin,pos,resultado =0;
        ent=calculaCaloriasEntrada();
        prin=calculaCaloriasPrincipal();
        pos=calculaCaloriasPostre();
        resultado=ent+pos+prin;

        return resultado;
    }

    public double calculaCaloriasEntrada (){
        double caloriasEntrada=0.0;
        switch(entrada){
            case 0:
                caloriasEntrada=500;
                break;
            case 1:
                caloriasEntrada=100;
                break;
            case 2:
                caloriasEntrada=600;
                break;
        }
        return caloriasEntrada;
    }
}

```

```

    public double calculaCaloriasPostre (){
        double caloriasPostre=2.0;
        switch(postre){
            case 0:
                caloriasPostre=300; //flan
                break;
            case 1:
                caloriasPostre=50; //manzana
                break;
            case 2:
                caloriasPostre=900; //pastel
                break;
        }
        return caloriasPostre;
    }

    public double calculaCaloriasPrincipal (){
        double caloriasPrincipal=0.0;
        switch(principal){
            case 0:
                caloriasPrincipal=80; //filete de res
                break;
            case 1:
                caloriasPrincipal=500; //pollo
                break;
            case 2:
                caloriasPrincipal=1200; //cabrito
                break;
        }
        return caloriasPrincipal;
    }
}

```