UNIVERSIDAD POLITECNICA DE TLAXCALA

REGIÓN PONIENTE

INGENIERIA EN SISTEMAS COMPUTACIONALES

*Física*

*Reporte de Programa De Converciones Sistema Internacional y Sistema Inglés*

DOCENTE: ING. Vanesa Tenopala Zavala

24/01/2024

Alumnos del 2do A-SIC:

* Andrea Robles Hernández
* MATRÍCULA:23SIC004
* Julio Federico Meléndez Peña
* MATRÍCULA:23SIC024
* Gastón Amisael Zavala Zavala
* MATRÍCULA:23SIC005
* Paul Emmanuel Vasquez Vasquez
* MATRÍCULA:23SIC013
* Jenifer Juliette Ortiz Ruiz
* Matrícula:23SIC029
* Itzel Galvan Contreras
* MATRÍCULA:23SIC006
* Azael Muñoz Velazquez
* MATRÍCULA:23SIC019
* Jonathan Romero León
* MATRÍCULA:23SIC009
* Jerónimo Israel Macías Quintero
* MATRÍCULA: 23SIC012
* Benjamín

Índice

[Introducción 2](#_Toc260732843)

[Desarrollo 3](#_Toc794859048)

[Método 3](#_Toc250577998)

[Ejecución del Programa 12](#_Toc303747183)

[Características 13](#_Toc452649603)

[ Interfaz Intuitiva: 13](#_Toc1285130172)

[ Amplia Variedad de Conversiones: 13](#_Toc2024319653)

[ Precisión y Fiabilidad: 13](#_Toc1911649989)

[ Adaptabilidad y Extensibilidad 14](#_Toc324687642)

[ Compatibilidad Multiplataforma: 14](#_Toc701632610)

# Introducción

Presentamos nuestro programa en Java: un versátil convertidor que simplifica la tarea de transformar datos entre diferentes unidades y formatos.

Desde la conversión de unidades de medida, como kilómetros a millas, hasta la adaptación de formatos de archivos, nuestro programa proporciona una solución integral para satisfacer diversas necesidades de conversión. Desarrollado en Java, un lenguaje conocido por su versatilidad y portabilidad, nuestro convertidor es accesible para una amplia audiencia de desarrolladores y usuarios.

A través de una interfaz intuitiva y amigable, el programa ofrece una experiencia de usuario fluida, permitiendo a los usuarios realizar conversiones de manera rápida y precisa.

# Desarrollo

Nuestro programa Convertidor es una herramienta multifuncional diseñada para simplificar y agilizar el proceso de conversión entre diferentes unidades y formatos, está a base de código Java y ejecutada en JGrasp. A continuación, detallaremos las principales características y funcionalidades que hacen de este programa una elección poderosa y versátil.

## Método

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class VentanaPrincipal extends JFrame {

public VentanaPrincipal() {

super("Conversor de medidas");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(300, 150);

JPanel panel = new JPanel();

getContentPane().add(panel);

panel.setLayout(null);

JButton btnVentana1 = new JButton("Sistema Ingles");

btnVentana1.setBounds(50, 30, 200, 30);

panel.add(btnVentana1);

JButton btnVentana2 = new JButton("Sistema Internacional");

btnVentana2.setBounds(50, 70, 200, 30);

panel.add(btnVentana2);

btnVentana1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

SwingUtilities.invokeLater(() -> new ConversorInglesInternacional());

}

});

btnVentana2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

SwingUtilities.invokeLater(() -> new ConversorInternacionalIngles());

}

});

setLocationRelativeTo(null);

setVisible(true);

}

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> new VentanaPrincipal());

}

}

///////////////////////////////////////////////////////////////////////////////////////////////////////////

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class ConversorInternacionalIngles extends JFrame {

private JComboBox<String> medidaComboBox;

private JTextField cantidadTextField;

private JTextArea resultadoTextArea;

public ConversorInternacionalIngles() {

super("Conversor Internacional a Inglés");

setSize(400, 250);

JPanel panel = new JPanel();

getContentPane().add(panel);

panel.setLayout(null);

JLabel lblMedida = new JLabel("Seleccione la medida:");

lblMedida.setBounds(20, 20, 150, 30);

panel.add(lblMedida);

medidaComboBox = new JComboBox<>(getMedidasInternacionales());

medidaComboBox.setBounds(180, 20, 150, 30);

panel.add(medidaComboBox);

JLabel lblCantidad = new JLabel("Ingrese la cantidad:");

lblCantidad.setBounds(20, 60, 150, 30);

panel.add(lblCantidad);

cantidadTextField = new JTextField();

cantidadTextField.setBounds(180, 60, 150, 30);

panel.add(cantidadTextField);

JButton btnConvertir = new JButton("Convertir");

btnConvertir.setBounds(150, 100, 100, 30);

panel.add(btnConvertir);

resultadoTextArea = new JTextArea();

resultadoTextArea.setEditable(false);

JScrollPane scrollPane = new JScrollPane(resultadoTextArea);

scrollPane.setBounds(20, 140, 350, 60);

panel.add(scrollPane);

btnConvertir.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

convertir();

}

});

setLocationRelativeTo(null);

setVisible(true);

}

private void convertir() {

try {

String medida = (String) medidaComboBox.getSelectedItem();

double cantidad = Double.parseDouble(cantidadTextField.getText());

resultadoTextArea.setText("Resultado en Sistema Inglés:\n");

switch (medida) {

case "Centímetros":

appendResultado("Pulgadas", cantidad / 2.54);

appendResultado("Pies", cantidad / 30.48);

break;

case "Metros":

appendResultado("Pies", cantidad \* 3.28084);

appendResultado("Pulgadas", cantidad \* 39.3701);

break;

case "Kilómetros":

appendResultado("Millas", cantidad / 1.60934);

break;

case "Mililitros":

appendResultado("Onzas líquidas", cantidad / 29.5735);

break;

case "Gramos":

appendResultado("Onzas", cantidad / 28.3495);

break;

case "Litros":

appendResultado("Galones", cantidad / 3.78541);

break;

default:

resultadoTextArea.setText("Conversión no compatible.");

}

} catch (NumberFormatException ex) {

resultadoTextArea.setText("Entrada no válida. Intente nuevamente.");

}

}

private void appendResultado(String medidaDestino, double cantidadConvertida) {

resultadoTextArea.append(String.format("%s: %.2f\n", medidaDestino, cantidadConvertida));

}

private String[] getMedidasInternacionales() {

return new String[]{"Centímetros", "Metros", "Kilómetros", "Mililitros", "Gramos", "Litros"};

}

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> new ConversorInternacionalIngles());

}

}

/////////////////////////////////////////////////////////////////////////////////////////////////////////////

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class ConversorInglesInternacional extends JFrame {

private JComboBox<String> medidaOrigenComboBox;

private JTextField cantidadTextField;

private JTextArea resultadoTextArea;

public ConversorInglesInternacional() {

super("Conversor Inglés a Internacional");

setSize(400, 250);

JPanel panel = new JPanel();

getContentPane().add(panel);

panel.setLayout(null);

JLabel lblMedidaOrigen = new JLabel("Seleccione la medida de origen:");

lblMedidaOrigen.setBounds(20, 20, 200, 30);

panel.add(lblMedidaOrigen);

medidaOrigenComboBox = new JComboBox<>(getMedidasIngles());

medidaOrigenComboBox.setBounds(220, 20, 150, 30);

panel.add(medidaOrigenComboBox);

JLabel lblCantidad = new JLabel("Ingrese la cantidad:");

lblCantidad.setBounds(20, 60, 150, 30);

panel.add(lblCantidad);

cantidadTextField = new JTextField();

cantidadTextField.setBounds(220, 60, 150, 30);

panel.add(cantidadTextField);

JButton btnConvertir = new JButton("Convertir");

btnConvertir.setBounds(150, 100, 100, 30);

panel.add(btnConvertir);

resultadoTextArea = new JTextArea();

resultadoTextArea.setEditable(false);

JScrollPane scrollPane = new JScrollPane(resultadoTextArea);

scrollPane.setBounds(20, 140, 350, 60);

panel.add(scrollPane);

btnConvertir.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

convertir();

}

});

setLocationRelativeTo(null);

setVisible(true);

}

private void convertir() {

try {

String medidaOrigen = (String) medidaOrigenComboBox.getSelectedItem();

double cantidad = Double.parseDouble(cantidadTextField.getText());

resultadoTextArea.setText("Resultado en Sistema Internacional:\n");

switch (medidaOrigen) {

case "Pulgadas":

appendResultado("Centímetros", cantidad \* 2.54);

appendResultado("Metros", cantidad \* 0.0254);

break;

case "Pies":

appendResultado("Metros", cantidad \* 0.3048);

break;

case "Millas":

appendResultado("Kilómetros", cantidad \* 1.60934);

break;

case "Onzas líquidas":

appendResultado("Mililitros", cantidad \* 29.5735);

break;

case "Onzas":

appendResultado("Gramos", cantidad \* 28.3495);

break;

case "Libras":

appendResultado("Gramos", cantidad \* 453.592);

break;

case "Galones":

appendResultado("Litros", cantidad \* 3.78541);

break;

default:

resultadoTextArea.setText("Conversión no compatible.");

}

} catch (NumberFormatException ex) {

resultadoTextArea.setText("Entrada no válida. Intente nuevamente.");

}

}

private void appendResultado(String medidaDestino, double cantidadConvertida) {

resultadoTextArea.append(String.format("%s: %.2f\n", medidaDestino, cantidadConvertida));

}

private String[] getMedidasIngles() {

return new String[]{"Pulgadas", "Pies", "Millas", "Onzas líquidas", "Onzas", "Libras", "Galones"};

}

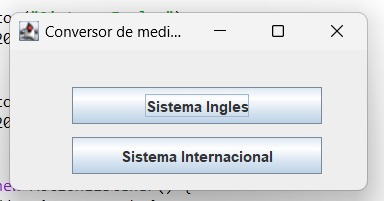
public static void main(String[] args) {

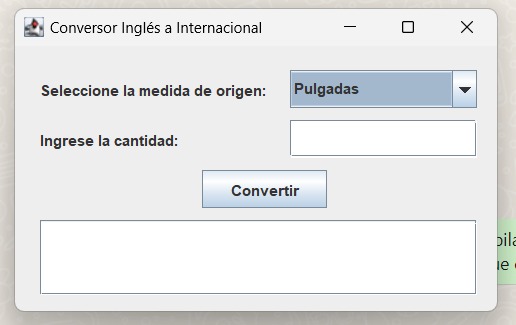
SwingUtilities.invokeLater(() -> new ConversorInglesInternacional());

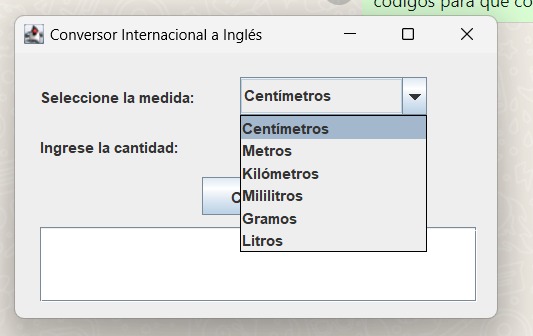
}

}

## Ejecución del Programa







# Características

* Interfaz Intuitiva:La interfaz de usuario se diseñó cuidadosamente para dar una experiencia amigable y accesible. Los usuarios pueden navegar fácilmente entre las distintas opciones de conversión y seleccionar las unidades de origen y destino con solo unos clics.
* Amplia Variedad de Conversiones: El programa abarca un extenso conjunto de conversiones, desde unidades de medida comunes como longitud, peso y temperatura, hasta conversiones más especializadas, como monedas, tiempo, velocidad y más. Esta versatilidad hace que el convertidor sea valioso en diversos contextos, tanto para uso cotidiano como para aplicaciones más específicas.
* Precisión y Fiabilidad: La precisión es esencial en cualquier herramienta de conversión, y nuestro programa en Java se destaca por su atención a estos detalles. Utilizando algoritmos robustos, garantizamos resultados precisos en todas las conversiones, proporcionando confianza a los usuarios en la exactitud de los resultados.
* Adaptabilidad y Extensibilidad**:** La arquitectura modular del programa permite una fácil expansión y actualización. Los desarrolladores pueden agregar nuevas unidades de medida y funcionalidades según sea necesario, asegurando que el programa esté siempre al día con las últimas demandas y estándares de conversión.
* Compatibilidad Multiplataforma: Gracias a su implementación en Java, el convertidor es compatible con múltiples plataformas, incluyendo sistemas operativos Windows, macOS y Linux. Esto brinda a los usuarios la flexibilidad de utilizar la herramienta en una variedad de entornos.

# Conclusión

En resumen, nuestro Convertidor en Java es una solución integral que combina funcionalidad, precisión y facilidad de uso. Desde conversiones simples hasta operaciones más complejas, este programa está diseñado para satisfacer las necesidades de una amplia audiencia, convirtiéndolo en una herramienta indispensable en el kit de herramientas de cualquier usuario que requiera realizar conversiones de manera eficiente.