



TECNOLOGICO NACIONAL DE MEXICO

**Instituto Tecnológico de la Laguna**

Ingeniería en Sistemas Computacionales

**TOPICOS AVANZADOS DE PROGRAMACION**

PERIODO: Ene - Jun / 2020

GRUPO: "B" 17 – 18 Hrs

PRACTICA No. U1P02

## CONVERSION DE UNIDADES CON METODOS ESTATICOS

ALUMNO:

18131209 José Misael Adame Sandoval

PROFESOR:

Ing. Luis Fernando Gil Vázquez

Torreón, Coah. a 17 de Febrero de 2020

# Ejercicio 1

Desarrollar una clase Java llamada Conversiones con métodos estáticos que permitan la conversión entre diferentes sistemas de unidades:

- De °C a °F
- De °F a °C
- De yardas a metros
- De metros a yardas
- De libras a kilos
- De kilos a libras
- De decimal a binario
- De binario a decimal

La clase debe declarar constantes con el factor de conversión (equivalencia) correspondiente en yardas-metros y libras-kilogramos.

Desarrollar una aplicación con interfaz gráfica de usuario para probar los métodos de la clase Conversiones.

NOTA: Incluir el botón Acerca de..., además diseñar una GUI clara y fácil de entender.

## Análisis

Se utilizarán métodos estáticos para realizar las operaciones de conversión de °C a °F, °F a °C, yardas a metros, metros a yardas, libras a kilos, kilos a libras, decimal a binario y binario a decimal.

La fórmula para obtener la conversión de °C a °F es:

$$^{\circ}\text{F} = \frac{9}{5} ^{\circ}\text{C} + 32$$

$^{\circ}\text{C} \neq \text{""}$

La fórmula para obtener la conversión de °F a °C es:

$$^{\circ}\text{C} = \frac{5}{9} ( ^{\circ}\text{F} - 32 )$$

$^{\circ}\text{F} \neq \text{""}$

La equivalencia de yardas a metros y viceversa es:

$$1 \text{ yd} = 0.9144 \text{ m}$$

```
yd, m ≠ ""
Si yd > 0 ? Realiza la conversión de yd a m
Si m > 0 ? Realiza la conversión de m a yd
```

La equivalencia de libras a metros y viceversa es:

$$1 \text{ lb} = 0.453 \text{ kg}$$

```
lb, kg ≠ ""
Si lb > 0 ? Realiza la conversión de lb a kg
Si kg > 0 ? Realiza la conversión de kg a lb
```

La conversión de decimal a binario se realiza de la siguiente forma:

```

num_dec ≠ ""
num_dec > 0

mientras num_dec > 0
    num_dec mod 2 = modulo
    binario = modulo + "" + binario // Se va agregando los residuos a un String
    num_dec = -----
                2

```

La conversión de binario a decimal se realiza de la siguiente forma

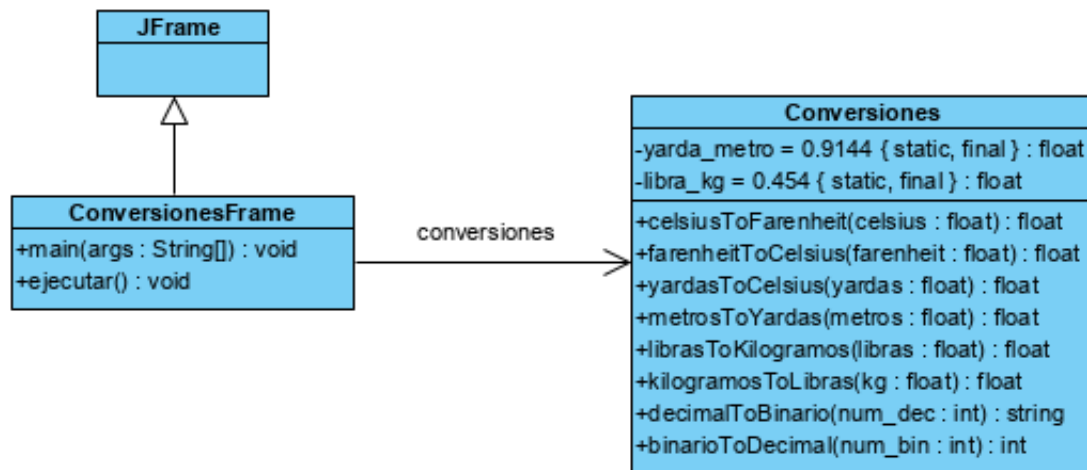
```

num_bin ≠ "",
base = 0
num_bin == 1 && num_bin == 0

mientras num_bin > 0
    num_bin mod 10 = ult_digito
    decimal = decimal + (ult_digito * 2base)
    num_bin = -----
                10
    base = base + 1

```

## Diseño



## Código

### Conversiones.java

```

/*-----
*:
*:          INSTITUTO TECNOLÓGICO DE LA LAGUNA
*:          INGENIERÍA EN SISTEMAS COMPUTACIONALES
*:          TÓPICOS AVANZADOS DE PROGRAMACIÓN "B"
*:
*:          SEMESTRE: ENE-JUN/2020      HORA: 17-18 HRS
*:
*:          Clase que realiza cálculos de conversiones
*:
*: Archivo      : Conversiones.java
*: Autor        : José Misael Adame Sandoval      18131209
*: Fecha        : 15/Feb/2020
*: Compilador   : JAVA J2SE v1.8.0
*: Descripción  : Clase que contiene métodos de cálculo estáticos para
*:                realizar las siguientes conversiones
*:                1. De °C a °F
*:                2. De °F a °C
*:                3. De yardas a metros
*:                4. De metros a yardas
*:                5. De libras a kilos
*:                6. De kilos a libras
*:                7. De decimal a binario
*:                8. De binario a decimal
*:
*: Última modif:
*: Fecha        Modificó          Motivo
*:=====
*: 16/feb/2020 Misael Adame      Agregar prólogo.
*:-----*/

package conversiones;

public class Conversiones {

    //-----

    // Atributos
    private static final float yarda_metro = 0.9144F;
    private static final float libra_kg = 0.454F;

    //-----

    // Métodos de cálculo
    public static float celsiusToFahrenheit ( float celsius ) {
        return ( celsius * 9 / 5 ) + 32;
    }

    //-----

    public static float fahrenheitToCelsius ( float fahrenheit ) {
        return ( fahrenheit - 32 ) * 5 / 9 ;
    }

    //-----

    public static float yardasToMetros ( float yardas ) {
        return yardas * yarda_metro;
    }

    //-----

    public static float metrosToYardas ( float metros ) {
        return metros / yarda_metro;
    }

    //-----

    public static float librasToKilogramos ( float libras ) {

```

```

        return libras * libra_kg;
    }

    //-----

    public static float kilogramosToLibras ( float kg ) {
        return kg / libra_kg;
    }

    //-----

    public static String decimalToBinario ( int num_dec ) {
        String binario = "";
        int modulo = 0;
        while ( num_dec > 0 ) {
            modulo = num_dec % 2;
            binario = modulo + "" + binario;
            num_dec = num_dec / 2;
        }
        return binario;
    }

    //-----

    public static int binarioToDecimal ( int num_bin ) {
        int decimal = 0, base = 0;
        while( num_bin > 0 ) {
            int ult_digito = num_bin % 10;
            decimal += ult_digito * Math.pow(2, base);
            num_bin = num_bin / 10;
            base++;
        }
        return decimal;
    }
}

```

## ConversionesFrame.java

```

/*-----
*:
*:          INSTITUTO TECNOLÓGICO DE LA LAGUNA
*:          INGENIERÍA EN SISTEMAS COMPUTACIONALES
*:          TEMAS AVANZADOS DE PROGRAMACIÓN "B"
*:
*:          SEMESTRE: ENE-JUN/2020      HORA: 17-18 HRS
*:
*:          Aplicación GUI para probar la clase Conversiones
*:
*: Archivo      : ConversionesFrame.java
*: Autor        : José Misael Adame Sandoval      18131209
*: Fecha        : 15/Feb/2020
*: Compilador   : JAVA J2SE v1.8.0
*: Descripción  : Aplicación que accede a la clase Conversiones e invoca a cada
*:                uno de sus métodos para comprobar su funcionamiento
*:
*: Última modif:
*: Fecha        Modificó          Motivo
*:=====
*: 16/feb/2020 Misael Adame      Agregar prólogo y realizar validaciones.
*:-----*/

package conversiones;

import javax.swing.ImageIcon;
import javax.swing.JOptionPane;

public class ConversionesFrame extends javax.swing.JFrame {

    //-----

    public ConversionesFrame() {
        initComponents();
    }
}

```

```

//-----

@SuppressWarnings("unchecked")
Generated Code

//-----

private void jButtonCalcularActionPerformed(java.awt.event.ActionEvent evt) {
    if ( validarDatos () ) {
        if ( ! ( jCboBoxConversion.getSelectedIndex () == 6 ||
                jCboBoxConversion.getSelectedIndex () == 7 ) ) {
            float num = Float.parseFloat ( jtxtDato.getText () );
            switch ( jCboBoxConversion.getSelectedIndex () ) {
                case 0:
                    jLabelResult.setText ( Float.toString ( Conversiones.celsiusToFahrenheit ( num ) ) + "
°F" );
                    break;
                case 1:
                    jLabelResult.setText ( Float.toString ( Conversiones.fahrenheitToCelsius ( num ) ) + "
°C" );
                    break;
                case 2:
                    jLabelResult.setText ( Float.toString ( Conversiones.yardasToMetros ( num ) ) + " m"
);
                    break;
                case 3:
                    jLabelResult.setText ( Float.toString ( Conversiones.metrosToYardas ( num ) ) + " yd"
);
                    break;
                case 4:
                    jLabelResult.setText ( Float.toString ( Conversiones.librasToKilogramos ( num ) ) + "
kg" );
                    break;
                default:
                    jLabelResult.setText ( Float.toString ( Conversiones.kilogramosToLibras ( num ) ) + "
lb");
                    break;
            }
        } else {
            int num = Integer.parseInt( jtxtDato.getText () );
            if ( jCboBoxConversion.getSelectedIndex() == 6 )
                jLabelResult.setText ( Conversiones.decimalToBinario (num) ) ;
            else
                jLabelResult.setText ( Integer.toString ( Conversiones.binarioToDecimal (num) ) ) ;
        }
    }
}

//-----

private void jButtonAcercaDeActionPerformed(java.awt.event.ActionEvent evt) {
    ImageIcon tec = new ImageIcon ( "ITL.png" );
    JOptionPane.showMessageDialog (
        this,
        "Tecnológico Nacional de México\n" +
        "Instituto Tecnológico de La Laguna\n\n" +
        "ConversionesApp v1.0 \n\n" +
        "Autor: José Misael Adame Sandoval 18131209",
        "Acerca de",
        JOptionPane.INFORMATION_MESSAGE,
        tec
    );
}

//-----

public boolean validarDatos () {
    try {
        switch ( jCboBoxConversion.getSelectedIndex () ) {
            case 2:
            case 3:
            case 4:
            case 5:

```

```

        return validarDistanciasPesos ();
    case 6:
    case 7:
        return validarSistemasNumericos ();
    default:
        float num = Float.parseFloat ( jtxfDato.getText () );
        return true;
    }
} catch ( NumberFormatException ex ) {
    switch ( jCboBoxConversion.getSelectedIndex () ) {
        case 2:
        case 3:
        case 4:
        case 5:
            JOptionPane.showMessageDialog (
                this,
                "Debe proporcionar un valor numérico positivo",
                "Error",
                JOptionPane.ERROR_MESSAGE
            );
            return false;
        case 6:
        case 7:
            JOptionPane.showMessageDialog (
                this,
                "Debe proporcionar un valor numérico entero positivo",
                "Error",
                JOptionPane.ERROR_MESSAGE
            );
            return false;
        default:
            JOptionPane.showMessageDialog (
                this,
                "Debe proporcionar un valor numérico",
                "Error",
                JOptionPane.ERROR_MESSAGE
            );
            return false;
    }
}
}

//-----

public boolean validarDistanciasPesos () {
    float num = Float.parseFloat ( jtxfDato.getText () );
    if ( num < 0 ) {
        JOptionPane.showMessageDialog (
            this,
            "Debe proporcionar un valor numérico positivo",
            "Error",
            JOptionPane.ERROR_MESSAGE
        );
        return false;
    }
    else
        return true;
}

//-----

public boolean validarSistemasNumericos () {
    int num = Integer.parseInt ( jtxfDato.getText () );
    if ( num < 0 ) {
        JOptionPane.showMessageDialog (
            this,
            "Debe proporcionar un valor numérico entero positivo",
            "Error",
            JOptionPane.ERROR_MESSAGE
        );
        return false;
    }
    else {
        if ( jCboBoxConversion.getSelectedIndex () == 7 ) {
            String num_bin = jtxfDato.getText () ;
            for ( int i = 0; i < num_bin.length (); i++ ) {

```

```

        char num_pos = num_bin.charAt ( i );
        if ( num_pos != '1' && num_pos != '0' ) {
            JOptionPane.showMessageDialog (
                this,
                "Debe proporcionar un valor numérico binario",
                "Error",
                JOptionPane.ERROR_MESSAGE
            );
            return false;
        }
    }
    return true;
} else
    return true;
}
}

//-----

public static void ejecutar() {
    java.awt.EventQueue.invokeLater ( new Runnable () {
        public void run () {
            new ConversionesFrame ().setVisible ( true );
        }
    });
}

//-----

public static void main ( String args [] ) {
    ejecutar ();
}

//-----

// Variables declaration - do not modify
private javax.swing.JButton jButtonAcercaDe;
private javax.swing.JComboBox<String> jCboBoxConversion;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel3;
private javax.swing.JButton jButtonCalcular;
private javax.swing.JLabel jLabelResult;
private javax.swing.JTextField jtxtDato;
// End of variables declaration
}

```

## ConversionesTest.java

```

/*-----
*:
*:          INSTITUTO TECNOLÓGICO DE LA LAGUNA
*:          INGENIERÍA EN SISTEMAS COMPUTACIONALES
*:          TEMAS AVANZADOS DE PROGRAMACIÓN "B"
*:
*:          SEMESTRE: ENE-JUN/2020      HORA: 17-18 HRS
*:
*:          Clase JUnit que realiza pruebas de los métodos
*:
*: Archivo      : ConversionesTest.java
*: Autor       : José Misael Adame Sandoval      18131209
*: Fecha       : 15/Feb/2020
*: Compilador  : JAVA J2SE v1.8.0
*: Descripción : Clase que contiene métodos de prueba (testing) e invoca
*:              a cada uno de los métodos de la clase Conversiones para
*:              comprobar su funcionamiento asignando un valor y su
*:              resultado esperado.
*:
*:
*: Última modif:
*: Fecha      Modificó      Motivo
*:-----
*: 17/feb/2020 Misael Adame      Agregar prólogo

```



```
:*-----*/

package conversionestest;

import conversiones.Conversiones;
import org.junit.After;
import org.junit.AfterClass;
import static org.junit.Assert.assertEquals;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;

public class ConversionesTest {

    //-----

    public ConversionesTest() {
    }

    //-----

    @BeforeClass
    public static void setUpClass() {
    }

    //-----

    @AfterClass
    public static void tearDownClass() {
    }

    //-----

    @Before
    public void setUp() {
    }

    //-----

    @After
    public void tearDown() {
    }

    //-----

    @Test
    public void celsiusToFahrenheitTest () {
        assertEquals ("38 °C a °F: " , 100.4,
            Conversiones.celsiusToFahrenheit ( 38 ), 0.0005 );
    }

    //-----

    @Test
    public void fahrenheitToCelsiusTest () {
        assertEquals ("73.4 °F a °C: " , 23.0,
            Conversiones.fahrenheitToCelsius ( 73.4f ), 0.0005 );
    }

    //-----

    @Test
    public void yardasToMetrosTest () {
        assertEquals ("8 Yardas a Metros: " , 7.3152,
            Conversiones.yardasToMetros ( 8 ), 0.00005 );
    }

    //-----

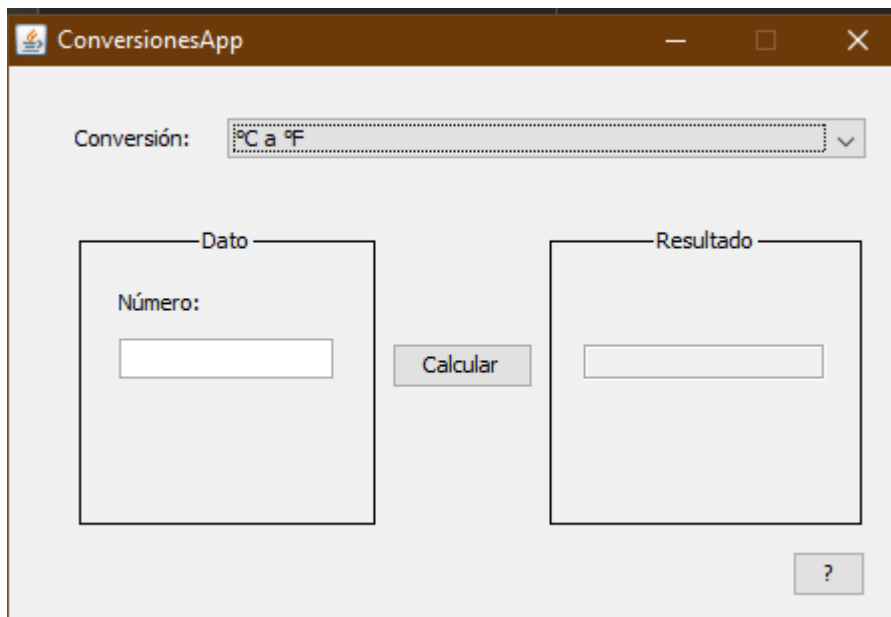
    @Test
    public void metrosToYardasTest () {
        assertEquals ("15 Metros a : " , 16.4042,
            Conversiones.metrosToYardas ( 15 ), 0.00005 );
    }
}
```

```
//-----  
  
@Test  
public void librasToKilogramosTest () {  
    assertEquals ("0.5 libras a Kilogramos: " , 0.227,  
        Conversiones.librasToKilogramos ( 0.5f ), 0.0005 );  
}  
  
//-----  
  
@Test  
public void kilogramosToMetrosTest () {  
    assertEquals ("22 Kilogramos a Libras: " , 48.45815,  
        Conversiones.kilogramosToLibras ( 22 ), 0.000005 );  
}  
  
//-----  
  
@Test  
public void decimalToBinarioTest () {  
    assertEquals ("29 en Decimal a Binario: " , "11101",  
        Conversiones.decimalToBinario ( 29 ) );  
}  
  
//-----  
  
@Test  
public void binarioToDecimalTest () {  
    assertEquals ("10101 en Binario a Decimal: " , 21,  
        Conversiones.binarioToDecimal ( 10101 ) );  
}  
}
```

## Prueba de Ejecución

Al ejecutar la aplicación es necesario capturar al menos un valor numérico

Esta es la interfaz inicial



Ejecutándose con °C a °F

The screenshot shows a window titled "ConversionesApp" with a standard Windows title bar (minimize, maximize, close buttons). Inside the window, there is a label "Conversión:" followed by a dropdown menu currently set to "°C a °F". Below this, there are two main sections: "Dato" (Data) and "Resultado" (Result). The "Dato" section contains a label "Número:" and a text input field containing the value "38". The "Resultado" section contains a text output field displaying "100.4 °F". Between these two sections is a button labeled "Calcular". At the bottom right of the window, there is a small button with a question mark "?".

Ejecutándose con °F a °C

The screenshot shows the same "ConversionesApp" window, but the dropdown menu is now set to "°F a °C". The "Dato" section's input field now contains "73.4". The "Resultado" section's output field now displays "23.0 °C". The "Calcular" button and the question mark button remain in the same positions.

Ejecutándose Yardas a Metros

The screenshot shows a window titled "ConversionesApp" with a dark red title bar. Inside, there is a dropdown menu labeled "Conversión:" with "Yardas a Metros" selected. Below this, there are two main sections: "Dato" and "Resultado". The "Dato" section has a label "Número:" and a text input field containing the value "8". The "Resultado" section has a text input field containing the value "7.3152 m". Between these two sections is a button labeled "Calcular" with a blue dashed border. At the bottom right of the window is a small button with a question mark "?".

Ejecutándose Metros a Yardas

The screenshot shows the same "ConversionesApp" window, but the dropdown menu "Conversión:" now has "Metros a Yardas" selected. The "Dato" section's text input field now contains the value "15". The "Resultado" section's text input field now contains the value "16.4042 yd". The "Calcular" button and the question mark button remain in the same positions.

Ejecutándose Libras a Kilogramos

The screenshot shows a window titled "ConversionesApp" with a standard Windows title bar. Inside the window, there is a dropdown menu labeled "Conversión:" with "Libras a Kilogramos" selected. Below this, there are two main sections: "Dato" and "Resultado". The "Dato" section has a label "Número:" and a text input field containing "0.5". The "Resultado" section has a text input field containing "0.227 kg". A "Calcular" button is positioned between the two sections. At the bottom right of the window, there is a small button with a question mark.

Ejecutándose Kilogramos a Libras

The screenshot shows the same "ConversionesApp" window, but the dropdown menu "Conversión:" now has "Kilogramos a Libras" selected. The "Dato" section's "Número:" input field now contains "22". The "Resultado" section's input field now contains "48.45815 lb". The "Calcular" button and the question mark button at the bottom right remain in the same positions.

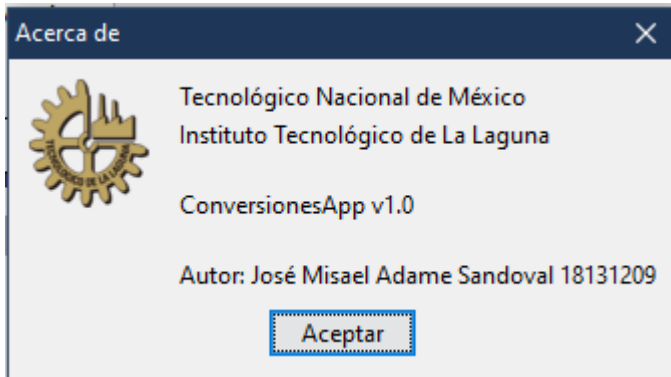
Ejecutándose Decimal a Binario

The screenshot shows a window titled 'ConversionesApp'. At the top, there is a dropdown menu labeled 'Conversión:' with 'Decimal a Binario' selected. Below this, there are two main sections: 'Dato' and 'Resultado'. In the 'Dato' section, there is a label 'Número:' followed by a text input field containing the value '29'. In the 'Resultado' section, there is a text input field containing the value '11101'. Between these two sections is a button labeled 'Calcular'. At the bottom right of the window, there is a small button with a question mark '?'.

Ejecutándose Binario a Decimal

The screenshot shows the same 'ConversionesApp' window, but with 'Binario a Decimal' selected in the 'Conversión:' dropdown menu. In the 'Dato' section, the 'Número:' input field now contains the binary value '10101'. In the 'Resultado' section, the output field contains the decimal value '21'. The 'Calcular' button remains between the sections, and the question mark button is still at the bottom right.

Al dar clic en el botón ¿ (Acerca de)



Al ejecutar la clase ConversionesTest fue exitoso en todos los casos de prueba

