

**TECNOLOGICO NACIONAL DE MEXICO
CAMPUS "OAXACA"**

INSTITUTO TECNOLÓGICO DE OAXACA

INGENIERIA EN SISTEMAS COMPUTACIONALES.

NOMBRE DE LA MATERIA: TOPICOS AVANZADOS DE PROGRAMACION.

UNIDAD 1: "INTERFAZ GRAFICA DEL USUARIO".

ACTIVIDAD: EJERCICIOS DE LENGUAJE JAVA 3 Y 4.

GRUPO: ISU

HORA: 09:00-10:00 AM

DOCENTE: M.C. LUIS ALBERTO ALONSO HERNÁNDEZ

ALUMNO: GONZALEZ PASCUAL MELVIN PAUL

FECHA DE ENTREGA: 18 DE OCTUBRE DEL 2020

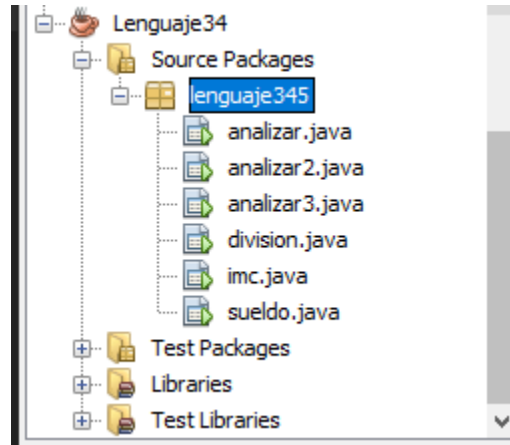
Contenido

Introducción:	3
Ejercicios de Lenguaje Java 1.	4
Ejercicio 1.	4
Ejercicio 2.	6
Ejercicio 3.	8
Ejercicios de Lenguaje Java 4.	11
Ejercicio 1.	11
Ejercicio 2.	13
Ejercicio 3.	15

Introducción:

Se elaborará cada de uno del ejercicio proporcionado de los PDF, lo cual se harán mediante la GUI NetBeans, los cuales se crearon mediante los componentes, se validarán los datos necesarios, así mismo se hará un diseño amigable con el usuario.

Se creará un nuevo proyecto el cual tendrá su propio paquete llamado “lenguaje java”, donde se encontrara los diversos ejercicios realizados.

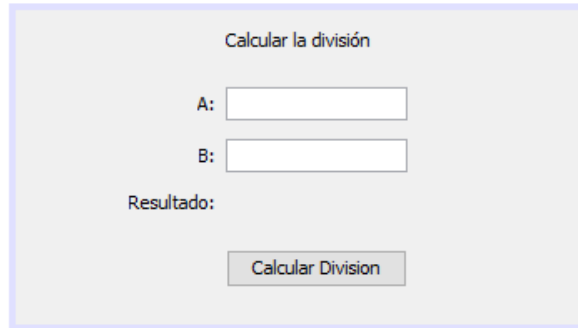


Ejercicios de Lenguaje Java 1.

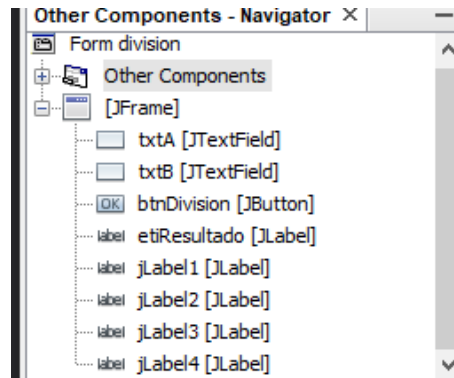
Ejercicio 1.

El siguiente ejercicio es calcular una división, en cual tendrá que ser validado de varias formas, las cual se explicara a continuación.

El cual tendrá de la siguiente forma.



Y los componentes tendrán los siguientes:



Creamos nuestro método que realizara toda la operación la cual es la siguiente.

```
//Metodo para hacer la división.
public void division(){
    String cad1, cad2;
    double a,b,r;
    cad1=txtA.getText();
    cad2=txtB.getText();

    a=Integer.parseInt(cad1);
    b=Integer.parseInt(cad2);
    //Si B es igual a 0, no dejara hacer la operación.
    if (b==0){
        JOptionPane.showMessageDialog(rootPane, "B debe ser mayor a 0");
        txtB.setText("");
    }
    //Si A y B son numeros negativos tampoco se podra realizar la operación
    if (a<0&b<0){
        JOptionPane.showMessageDialog(rootPane, "Solo numeros positivos");
        txtB.setText("");
        txtA.setText("");
    }
    //Si es mayor podra hacer la operación.
    if (a>0&b>0){
        r=a/b;
        etiResultado.setText(""+r);
    }
}
```

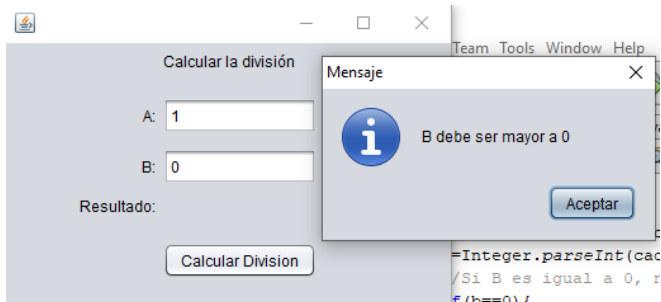
Lo siguiente es crear el método donde validaremos los campos de texto.

```
//Metodo para validar los datos
public void verif(KeyEvent evt){
    char validar = evt.getKeyChar();
    if(Character.isLetter(validar)){
        getToolkit().beep();
        evt.consume();
        JOptionPane.showMessageDialog(rootPane, "Ingrese solo numeros");
    }
}
```

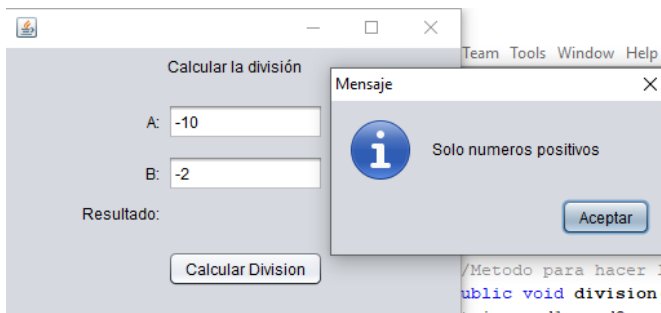
En cada evento correspondiente, le agregaremos los métodos.

```
//Evento para hacer el calculo.
private void btnDivisionActionPerformed(java.awt.event.ActionEvent evt) {
    division();
}
//Evento para validar el campo de texto
private void txtBKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
//Evento para validar el campo de texto
private void txtAKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
```

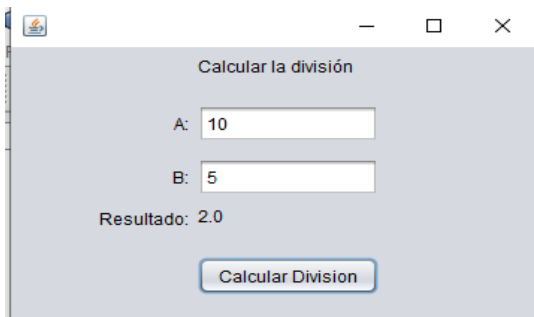
Ahora haremos las pruebas del funcionamiento.



Si el valor de B es 0, nos mostrara el siguiente mensaje.



Si los números son negativos, nos mostrara lo siguiente.



Si los parámetros son los correctos se podrá realizar sin problema, por lo cual es ejercicio se queda completo.

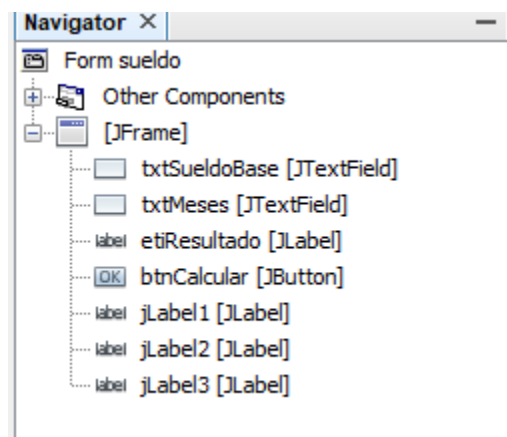
Ejercicio 2.

El siguiente ejercicio es calcular el sueldo de un empleado, además deberá coincidir con la contraseña, para que se pueda estar calculada.

Se usará el siguiente diseño.

Diagrama de interfaz de usuario para el cálculo de sueldo. El diseño incluye tres etiquetas con campos de entrada de texto: "Cobro por mes:", "Meses trabajados:" y "Total a percibir". Debajo de estos campos hay un botón etiquetado "Calcular".

En cual las variables tendrán el siguiente nombre:



Se crearán los métodos para calcular y validar los cuales son los siguiente:

Nuestro método para hacer los cálculos.

```
public void calcular() {  
    String a,b,contrasena;  
    int s,m,r;  
    a=txtSueldoBase.getText();  
    b=txtMeses.getText();  
    s=Integer.parseInt(a);  
    m=Integer.parseInt(b);  
    //Mensaje de salida que pide la contraseña  
    contrasena=JOptionPane.showInputDialog("Ingrese su contraseña");  
    //Si la contraseña es igual, se podra hacer el calculo  
    if(contrasena.equals("hola")) {  
        r=(int) ((s-(0.1*s)) *m);  
        etiResultado.setText(""+r);  
    }  
    //Si la contraseña es incorrecta no se podra calcular.  
    else{  
        JOptionPane.showMessageDialog(rootPane, "Contraseña invalida");  
    }  
}
```

Método para la validación.

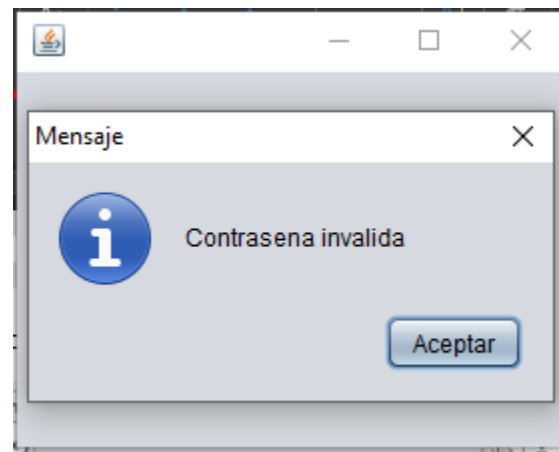
```
//Metodo para la validacion.  
public void verif(KeyEvent evt){  
    char validar = evt.getKeyChar();  
    if(Character.isLetter(validar)){  
        getToolkit().beep();  
        evt.consume();  
        JOptionPane.showMessageDialog(rootPane, "Ingrese solo numeros");  
    }  
}
```

Lo métodos creado se irán a los eventos correspondientes.

```
//Evento para calcular el sueldo  
private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {  
    calcular();  
}  
  
//Evento para la validacion de los datos  
private void txtSueldoBaseKeyTyped(java.awt.event.KeyEvent evt) {  
    verif(evt);  
}  
  
//Evento para la validacion de los datos  
private void txtMesesKeyTyped(java.awt.event.KeyEvent evt) {  
    verif(evt);  
}
```

Lo siguiente será hacer los cálculos de nuestro sueldo.

Al momento de calcular nos pedirá la contraseña, si la contraseña es incorrecta nos mostrara lo siguiente.

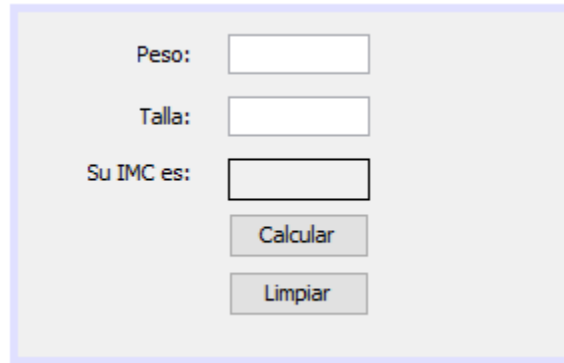


Si nuestra contraseña es correcta nos calculara el total de la siguiente manera.

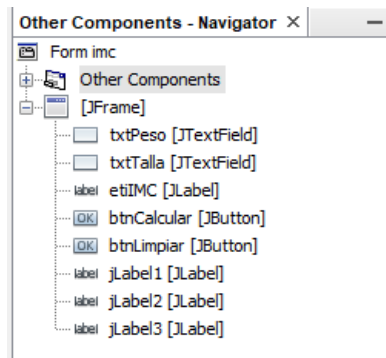
Por lo que podemos concluir que nuestro programa se realizo con éxito.

Ejercicio 3.

En el siguiente ejercicio calcularemos el Índice de Masa Corporal, además de calcular, nos enviara en un mensaje emergente, la conclusión que tiene nuestro cuerpo, por lo cual nuestro diseño será el siguiente.



Las cuales sus variables serán las siguientes.



Se crearon los siguientes métodos.

```
//Metodo que nos retonar el IMC, demas de calcularlo
public double calcularimc(){
    String cad1,cad2;
    double p,t,r;
    cad1=txtPeso.getText();
    cad2=txtTalla.getText();
    p=Double.parseDouble(cad1);
    t=Double.parseDouble(cad2);
    r=(p/Math.pow(t, 2));
    etiIMC.setText(""+r);
    return r;
}
```

```
//Metodo que nos imprime las conclusiones con el IMC obtenido
public void impresion(double r){
    if(r<18) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene anorexia");
    if(r>18&r<20) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene delgadez");
    if(r>20&r<27) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene normalidad");
    if(r>27&r<30) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene obesidad(grado 1)");
    if(r>30&r<35) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene obesidad(grado 2)");
    if(r>35&r<40) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene obesidad(grado 3)");
    if(r>40) JOptionPane.showMessageDialog(null, "Su IMC indica que tiene obesidad mórbida");
}
```



```
//Metodo para borrar y hacer un nuevo calculo.
public void borrar(){
    int resp=JOptionPane.showConfirmDialog(null,"¿Desea borrar los datos?");
    if(JOptionPane.YES_OPTION==resp){
        txtPeso.setText("");
        txtTalla.setText("");
        etiIMC.setText("");
    }
}

//Metodo para validar los campos de texto.
public void verif(KeyEvent evt){
    char validar = evt.getKeyChar();
    if(Character.isLetter(validar)){
        getToolkit().beep();
        evt.consume();
        JOptionPane.showMessageDialog(rootPane, "Ingrese solo numeros");
    }
}
}

/**
```

Los siguiente es agregar los métodos en los eventos necesarios.

```
//Evento que crea el metodo de calcular, ademas de imprimir la conclusion
private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {
    double r=calcularimc();
    impresion(r);
}

//Metodo para borrar los resultados
private void btnLimpiarActionPerformed(java.awt.event.ActionEvent evt) {
    borrar();
}

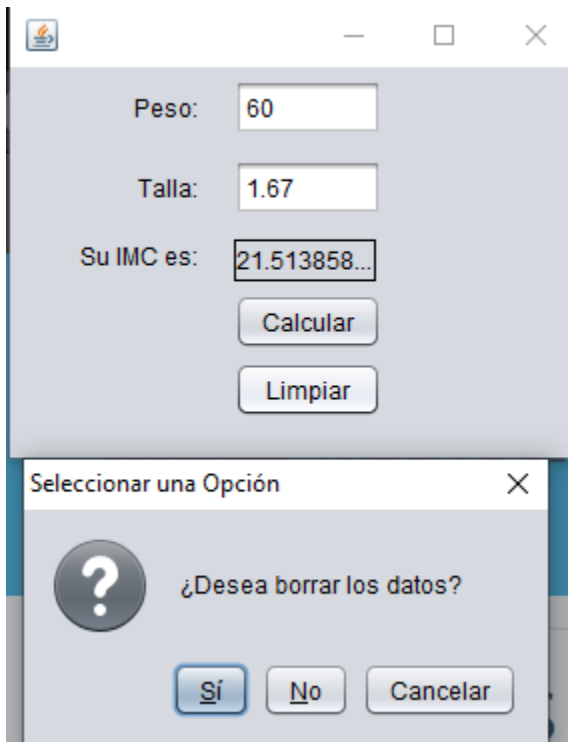
//Metodo para validar el texto
private void txtPesoKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}

//Metodo para validar el texto
private void txtTallaKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
}
```

Lo siguiente es calcular para saber nuestro IMC:

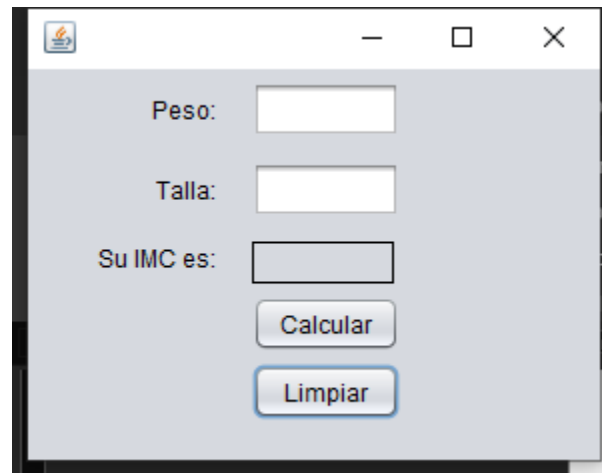
Como se puede apreciar se calcula y nos sale el resultado, ahora usaremos nuestro botón limpiar.

En cual nos pedirá una verificación si se desea eliminar el contenido.



The screenshot shows a BMI calculator application window. It has three input fields: 'Peso:' with the value '60', 'Talla:' with the value '1.67', and 'Su IMC es:' with the calculated value '21.513858...'. Below these fields are two buttons: 'Calcular' and 'Limpiar'. A modal dialog box titled 'Seleccionar una Opción' is open in the foreground. It contains a question mark icon and the text '¿Desea borrar los datos?'. At the bottom of the dialog are three buttons: 'Sí', 'No', and 'Cancelar'.

En cual debemos de confirmar la acción para que sea completado.



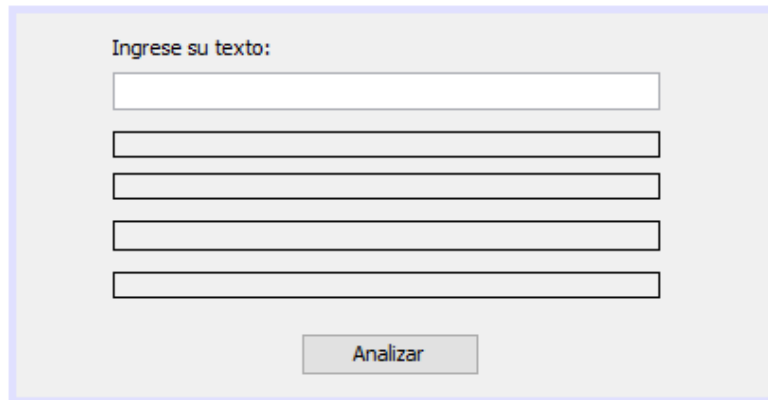
The screenshot shows the same BMI calculator application window, but the input fields are now empty. The 'Peso:' field is empty, the 'Talla:' field is empty, and the 'Su IMC es:' field is empty. The 'Calcular' and 'Limpiar' buttons are still present at the bottom.

Por cual se puede concluir la practica

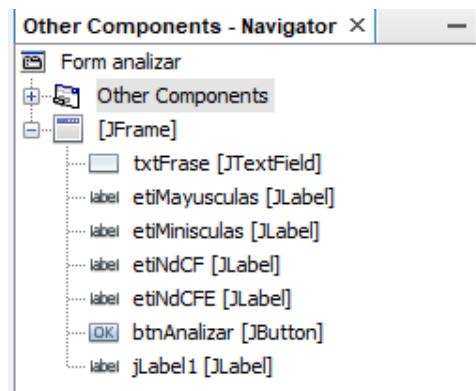
Ejercicios de Lenguaje Java 4.

Ejercicio 1.

El siguiente ejercicio será calcular la cantidad de mayúsculas, minúsculas, el total de palabras, y el total sin espacio, por lo cual se implementó el siguiente diseño.



El cual las variables tendrán los siguientes nombres:



Se crearon los siguientes métodos.

```
//Metodo para calcular las operaciones requeridas.
public void contador(){
    //Condicion que si el cuadro de texto esta vacio, nos regresa un mensaje
    if(txtFrase.getText().length()==0){
        JOptionPane.showMessageDialog(rootPane, "Ingrense Datos");
    }
    else{
        String cadl=txtFrase.getText();
        int m=0;//Se creo dos contadores
        int n=0;
        etiNdCF.setText("Total de letras: "+cadl.length());//Tamaño de la cadena
        for(int x = 0;x<cadl.length();x++){//Ciclo para recorrer todas las palabras
            //Condicion para las palabras Mayusculas
            if(Character.isUpperCase(cadl.charAt(x))){
                m++;
                etiMayusculas.setText("Total de letras con mayusculas: "+m);
            }
            //Condicion para las palabras minusculas
            if(Character.isLowerCase(cadl.charAt(x))){
                n++;
                etiMinusculas.setText("Total de letras con minusculas: "+n);
            }
        }
        cadl=cadl.replaceAll(" ", "");//Se eliminan los espacios
        etiNdCFE.setText("Total de letras sin espacio: "+cadl.length());//El tamaño sin espacios
    }
}
```

```

//Metodo para validar
public void verif(KeyEvent evt){
    char validar = evt.getKeyChar();
    if (Character.isDigit(validar)){
        getToolkit().beep();
        evt.consume();
        JOptionPane.showMessageDialog(rootPane, "Ingrese solo letras");
    }
}

```

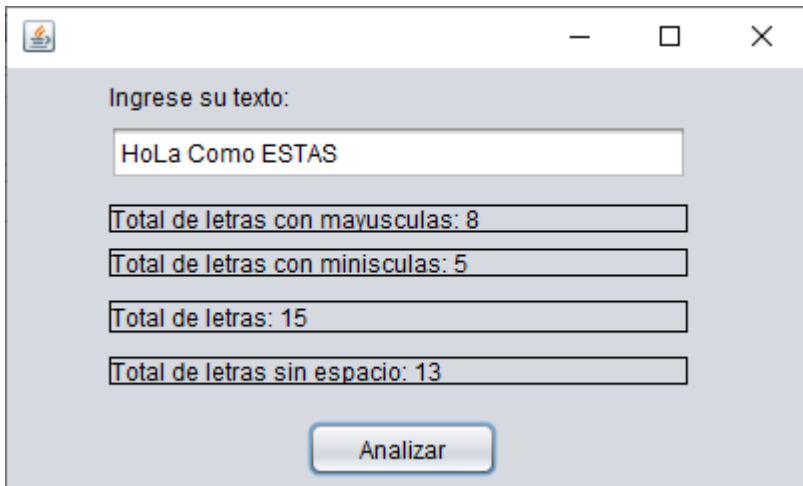
Los siguientes métodos serán implementados en los siguientes eventos.

```

//Evento para hacer el conteo
private void btnAnalizarActionPerformed(java.awt.event.ActionEvent evt) {
    contador();
}
//Metodo para validar.
private void txtFraseKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}

```

Lo siguiente es ejecutar y probar escribiendo una cadena con varias palabras.



Ingrese su texto:

HoLa Como ESTAS

Total de letras con mayusculas: 8

Total de letras con minusculas: 5

Total de letras: 15

Total de letras sin espacio: 13

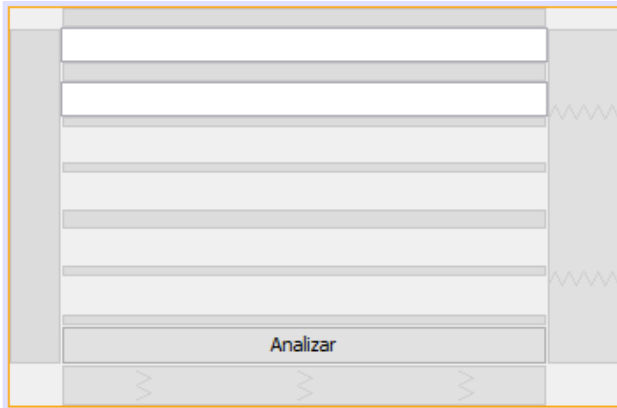
Analizar

Como se puede observar, el programa cumple con lo que se pide.

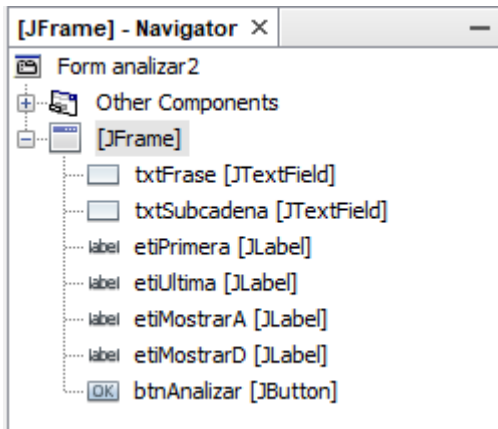
Ejercicio 2.

El siguiente ejercicio buscará la primera y la última aparición de una palabra, además que imprimirá lo que iba a antes y después de cada palabra.

El diseño será el siguiente:



Y los componentes tendrán los siguientes nombres:



Y se crearan los siguientes métodos:

```
//Metodo para hacer las busquedas de las palbaras
public void buscar(){
    //Veirifca que el cuadro de texto tenga los datos
    if(txtFrase.getText().length()==0){
        JOptionPane.showMessageDialog(rootPane, "Ingrense Datos");
    }
    //En caso contrario empezara la busqueda.
    else{
        String cad1=txtFrase.getText();
        String cad2=txtSubcadena.getText();
        //Si lo contiene realizara lo siguiente
        if( cad1.contains(cad2)){
            int x=cad1.indexOf(cad2); //La primera aparicion
            int y=cad1.lastIndexOf(cad2); //La segunda aparicion
            etiPrimera.setText("Encontrado en la posición: "+x);
            etiUltima.setText("Encontrado en la posición: "+y);
            etiMostrarA.setText(cad1.substring(0,cad1.indexOf(cad2))); //Crea la subcadena anterior
            etiMostrarD.setText(cad1.substring(y+cad2.length(),cad1.length())); //Crea la subcadena posterior
        }else{
            //Si no se encuentra lo que se busca, no se realiza nada.
            JOptionPane.showMessageDialog(rootPane, "Palabra no encontrada, no se puede realizar la acción");
        }
    }
}
```

```

//Metodo para la validacion
public void verif(KeyEvent evt){
    char validar = evt.getKeyChar();
    if(Character.isDigit(validar)){
        getToolkit().beep();
        evt.consume();
        JOptionPane.showMessageDialog(rootPane, "Ingrese solo letras");
    }
}

```

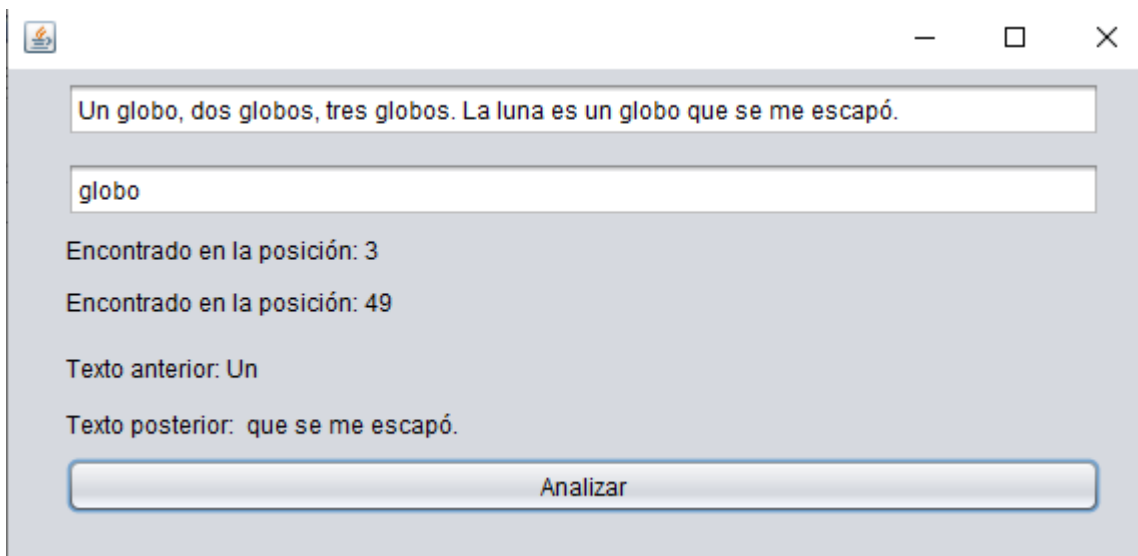
Se agregarán en los siguientes eventos:

```

private void btnAnalizarActionPerformed(java.awt.event.ActionEvent evt) {
    buscar();
    // TODO add your handling code here:
}
//Verifica el teclado.
private void txtFraseKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
//Verifica el teclado.
private void txtSubcadenaKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}

```

Y el funcionamiento es lo siguiente:

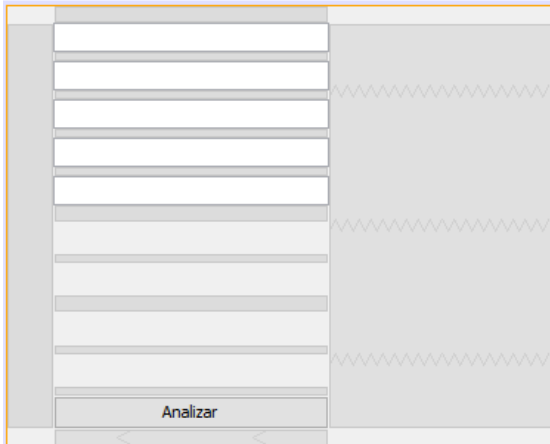


Por lo cual el ejercicio esta completo.

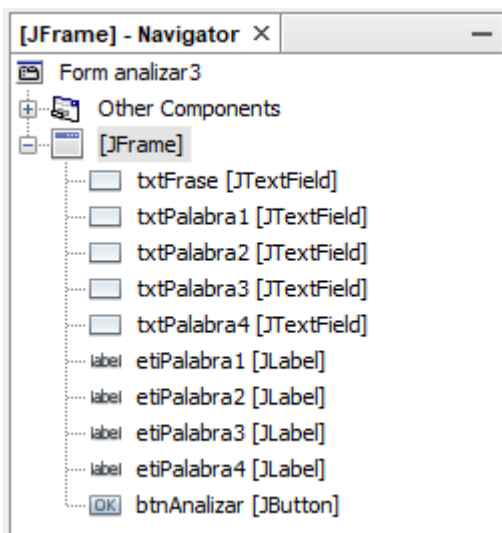
Ejercicio 3.

El siguiente programa será de búsqueda de las palabras, nos indicara si esta al principio, final, si es igual o si lo contiene.

El siguiente diseño será el siguiente:



Y los componentes el siguiente nombre:



Lo siguiente será la creación de los métodos:

```
//Metodo para la busqueda
public void cadenas() {
    //Si esta vacio no se realiza ninguna opcion.
    if (txtFrase.getText().length() == 0) {
        JOptionPane.showMessageDialog(rootPane, "Ingrense Datos");
    } else {
        //Se crean varias cadenas.
        String cad1 = txtFrase.getText();
        String cad2 = txtPalabra1.getText();
        String cad3 = txtPalabra2.getText();
        String cad4 = txtPalabra3.getText();
        String cad5 = txtPalabra4.getText();
        //Condicion para ver si es igual
        if (cad1.equals(cad2)) {
            etiPalabra1.setText("La palabra es igual a la frase");
        } else {
            etiPalabra1.setText("La palabra no es igual");
        }
        //Condicion para ver si es comienzo
        if (cad1.startsWith(cad3)) {
            etiPalabra2.setText("Si comienza");
        } else {
            etiPalabra2.setText("No empieza ");
        }
    }
}
```

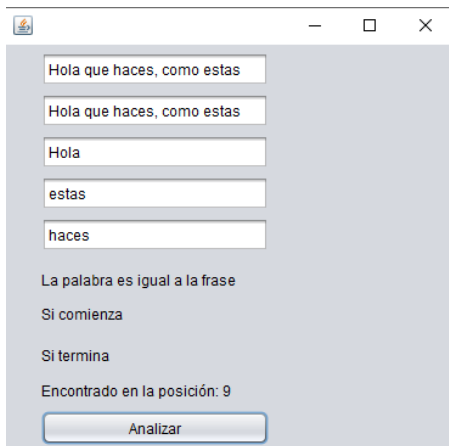
```
//Condicion para ver si es comienza
if(cad1.startsWith(cad3)){
    etiPalabra2.setText("Si comienza");
}else{
    etiPalabra2.setText("No empieza ");
}
//Condicion para ver si termina
if(cad1.endsWith(cad4)){
    etiPalabra3.setText("Si termina");
}else{
    etiPalabra3.setText("No termina ");
}
//Condicion para ver si contiene esa palabra
if( cad1.contains(cad5)){
    etiPalabra4.setText("Encontrado en la posición: "+cad1.indexOf(cad5));
}else{etiPalabra4.setText("No se encontro");}
}
}
```

```
//Metodo para validar el teclado.
public void verif(KeyEvent evt){
    char validar = evt.getKeyChar();
    if(Character.isDigit(validar)){
        getToolkit().beep();
        evt.consume();
        JOptionPane.showMessageDialog(rootPane, "Ingrese solo letras");
    }
}
```

Y será agregado en los siguientes eventos.

```
//Evento del metodo.
private void btnAnalizarActionPerformed(java.awt.event.ActionEvent evt) {
    cadenas();
    // TODO add your handling code here:
}
//Eventos de validacion
private void txtFraseKeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
private void txtPalabra1KeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
private void txtPalabra2KeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
private void txtPalabra3KeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
private void txtPalabra4KeyTyped(java.awt.event.KeyEvent evt) {
    verif(evt);
}
```

El funcionamiento es el siguiente:



Por lo que el programa esta completo, por lo cual ha quedado terminado la practica.