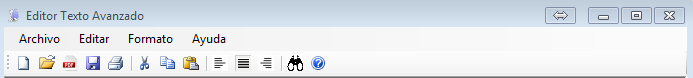
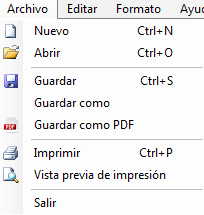
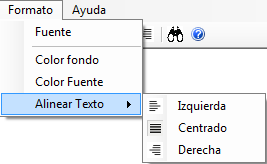
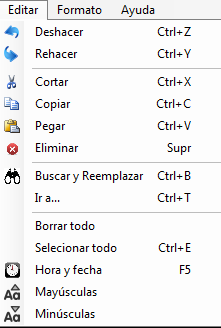
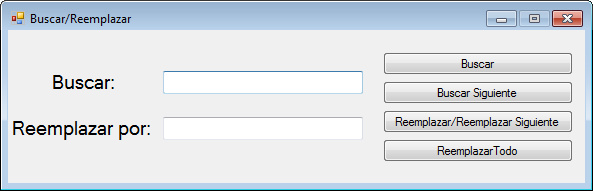
TRABAJO 2- Editor de Texto Avanzado

**DESCRIPCIÓN**

La siguiente aplicación trata sobre un editor avanzado de texto con el cual se pueden realizar las operaciones básicas de creación de ficheros, lectura y edición de los mismos.







**TABLA DE PROPIEDADES Y VALORES**

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | Form1 |
| MainMenuStrip | MenuStrip1 |
| Text | Editor Texto Avanzado |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | MenuStrip1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | ArchivoToolStripMenuItem |
| Text | Archivo |

Misma nomenclatura para los menús de Editar, Formato y Ayuda.

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | NuevoToolStripMenuItem |
| Text | Nuevo |
| ShourtcutKeys | Ctrl+N |

Misma nomenclatura para el resto de elementos del menú Archivo, Editar, formato y Ayuda.

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | BarraHerramientas |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | NumeroLineaColumna |
| Dock | Bottom |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | lblColumna |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | lblLinea |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | txtContenido |
| AcceptsTab | True |
| Dock | Fill |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | PrintDialog1 |
| Document | PrintDocument1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | ColorDialog1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | FontDialog1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | SaveFileDialog1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | SaveFilePDF |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | PrintDocument1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | PrintPreviewDialog1 |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | OpenFileDialog1 |

Formulario frmBusqueda:

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | frmBusqueda |
| AutoSizeMode | GrowAndShrink |
| Text | Buscar/Reemplazar |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | btnBuscar |
| Text | Buscar |

Mismas propiedades para el resto de botones cambiando el nombre del botón respectivamente para Buscar, Buscar Siguiente, reemplazar y Reemplazar todos.

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | lblBuscar |
| Text | Buscar: |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | lblReemplazar |
| Text | Reemplazar por:: |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | txtBuscar |

|  |  |
| --- | --- |
| PROPIEDAD | VALOR |
| Name | txtReemplazar |

**CÓDIGO FUENTE DE LA APLICACIÓN**

Imports System.IO

Imports System.Drawing.Printing

Imports iTextSharp

Imports iTextSharp.text

Imports iTextSharp.text.pdf

Public Class Form1

Dim cadena As String 'variable que guarda la ruta de acceso el nombre de archivo y la extensión

Dim ruta As String 'Almacenamos ruta documento.

Dim esGuardado As Boolean = False 'Variable que nos permite saber si un documento ha sido guardado o no.

Dim fecha, dia As String 'Guardamos la fecha y hora actual del sistemas

Dim posicion, linea, col As Integer 'Guardamos posicion, número de línea y columna.

Private Sub abrirSubMenu\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles abrirSubMenu.Click, AbrirToolStripButton.Click

OpenFileDialog1.Filter = "Text (\*.txt)|\*.txt||\*.rtf|HTML(\*.html\*)|\*.html|PHP(\*.php\*)|\*.php\*|Rich Text Files|\*.rtf|All files(\*.\*)|\*.\*"

If OpenFileDialog1.ShowDialog() = Windows.Forms.DialogResult.OK Then

OpenFileDialog1.FileName = OpenFileDialog1.FileName 'Abrimos dialogo de apertura de ficheros

cadena = OpenFileDialog1.FileName 'Guardamos el nombre del archivo en la variable cadena

txtContenido.LoadFile(OpenFileDialog1.FileName, RichTextBoxStreamType.RichText) 'Cargamos el arvhivo

End If

ruta = Path.GetFullPath(OpenFileDialog1.FileName) + " -Editor Texto Avanzado" 'Guardamos en ruta la ruta completa junto con el nombre y la extensión del fichero

Me.Text = ruta 'Mostramos como título del formulario la ruta del fichero

End Sub

Private Sub NuevoToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles NuevoToolStripMenuItem.Click, NuevoToolStripButton.Click

txtContenido.Clear() 'Limpia el contenido del RichTextBox

End Sub

Private Sub GuardarToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles GuardarToolStripMenuItem.Click, GuardarToolStripButton.Click

SaveFileDialog1.Filter = "Text (\*.txt)|\*.txt|Rich Text Files|\*.rtf|HTML(\*.html\*)|\*.html|PHP(\*.php\*)|\*.php\*|All files(\*.\*)|\*.\*" 'Diferentes formatos con los que podemos guardar el documento

SaveFileDialog1.DefaultExt = "Rich Text Files|\*.rtf"

If cadena = "" Then 'Si el documento no existía lo guardamos como (ruta,nombre y extensión)

GuardarcomoToolStripMenuItem\_Click(Nothing, Nothing)

Else

txtContenido.SaveFile(OpenFileDialog1.FileName, RichTextBoxStreamType.RichText) 'Guardamos el contenido del RichTextBox con formato

End If

esGuardado = True 'Ponemos la variable a true ya que hemos guardado el documento

End Sub

Private Sub GuardarcomoToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles GuardarcomoToolStripMenuItem.Click

SaveFileDialog1.Filter = "Text (\*.txt)|\*.txt|Rich Text Files|\*.rtf|HTML(\*.html\*)|\*.html|PHP(\*.php\*)|\*.php\*|All files(\*.\*)|\*.\*" 'Diferentes formatos con los que podemos guardar el documento

SaveFileDialog1.DefaultExt = "Rich Text Files|\*.rtf" 'Formato de guardado por defecto

If SaveFileDialog1.ShowDialog() = Windows.Forms.DialogResult.OK Then

SaveFileDialog1.FileName = SaveFileDialog1.FileName 'Mostramos diálogo de guardado

txtContenido.SaveFile(SaveFileDialog1.FileName, RichTextBoxStreamType.RichText) 'Guardamos el contenido del RichTextBox con formato

End If

esGuardado = True 'Ponemos la variable a true ya que hemos guardado el documento

End Sub

Private Sub DeshacerToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles DeshacerToolStripMenuItem.Click

txtContenido.Undo() 'Deshace las acciones

End Sub

Private Sub RehacerToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles RehacerToolStripMenuItem.Click

txtContenido.Redo() 'Rehace las acciones

End Sub

Private Sub CortarToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CortarToolStripMenuItem.Click, CortarToolStripButton.Click

txtContenido.Cut() 'Opción de cortar texto seleccionado.

End Sub

Private Sub CopiarToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CopiarToolStripMenuItem.Click, CopiarToolStripButton.Click

txtContenido.Copy() 'Opción de copiado texto seleccionado

End Sub

Private Sub PegarToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PegarToolStripMenuItem.Click, PegarToolStripButton.Click

txtContenido.Paste() 'Opción de pegado texto seleccionado

End Sub

Private Sub SeleccionartodoToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

txtContenido.SelectAll() 'Seleccionamos todo el contenido del RichTextBox

End Sub

Private Sub BuscarYReemplazarToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BuscarYReemplazarToolStripMenuItem.Click, btnBuscar.Click

frmBusqueda.Show() 'Muestra el formulario de búsqueda/reemplado de palabras.

End Sub

Private Sub ColorFuenteToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ColorFuenteToolStripMenuItem.Click 'Cambiamos el color texto.

If txtContenido.Text = String.Empty Then

MsgBox("No hay texto para cambiar el color.Introduza texto y después proceda a cambiar su color.", 48, "Atención")

Else

If ColorDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then

txtContenido.SelectionColor = ColorDialog1.Color

End If

End If

End Sub

Private Sub FuenteToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles FuenteToolStripMenuItem.Click 'Cambiamos la fuente del texto.

If txtContenido.Text = String.Empty Then

MsgBox("No hay texto para cambiar la fuente.Introduza texto y después proceda a cambiar su fuente.", 48, "Atención")

Else

If FontDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then

txtContenido.SelectionFont = FontDialog1.Font

End If

End If

End Sub

Private Sub ColorFondoToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ColorFondoToolStripMenuItem.Click 'Cambiamos el color de fondo.

If ColorDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then

txtContenido.BackColor = ColorDialog1.Color

End If

End Sub

Private Sub AcercadeToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles AcercadeToolStripMenuItem.Click, AyudaToolStripButton.Click

MsgBox("Editor de texto avanzado desarrollado para DE", 64, "Acerca de este editor...")

End Sub

Private Sub ReemplazarToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs)

frmBusqueda.Show() 'Muestra el formulario de búsqueda/reemplado de palabras.

End Sub

Private Sub SelecionarTodoToolStripMenuItem\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles SelecionarTodoToolStripMenuItem.Click

txtContenido.SelectAll() 'Selecciona todo el contenido del RichTextBox

End Sub

Private Sub MayúsculasToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles MayúsculasToolStripMenuItem.Click

txtContenido.SelectedText = txtContenido.SelectedText.ToUpper 'Convertimos el texto seleccionado a mayúsculas

End Sub

Private Sub MinúsculasToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles MinúsculasToolStripMenuItem.Click

txtContenido.SelectedText = txtContenido.SelectedText.ToLower 'Convertimos el texto seleccionado a minúsculas

End Sub

Private Sub SalirToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles SalirToolStripMenuItem.Click 'Controlamos la salida de la aplicación teniendo en cuenta si el documento ha sido guardado o no.

If esGuardado Then 'Si el documento ha sido guardado, salimos de la aplicación

End

Else

Dim result As Integer = MsgBox("¿Quieres guardar el documento?", MsgBoxStyle.YesNo, Title:="Guardar documento")

If result = MsgBoxResult.Yes Then

If cadena = "" Then 'Si no existe el documento lo guardamos especificando ruta , nombre y extensión

GuardarcomoToolStripMenuItem\_Click(Nothing, Nothing)

Else

GuardarToolStripMenuItem\_Click(Nothing, Nothing) 'Si ya existía el documento guardamos los cambios y salimos de la aplicación

End If

End

ElseIf result = MsgBoxResult.No Then

End

End If

End

End If

esGuardado = True

End Sub

Private Sub ImprimirToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles ImprimirToolStripMenuItem.Click, ImprimirToolStripButton.Click 'Imprime el documento actual.

If PrintDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then

PrintDocument1.PrinterSettings = PrintDialog1.PrinterSettings

PrintDocument1.Print()

End If

End Sub

Private Sub PrintDocument1\_PrintPage(sender As System.Object, e As System.Drawing.Printing.PrintPageEventArgs) Handles PrintDocument1.PrintPage 'Imprime el documento actual.

e.Graphics.DrawString(txtContenido.Text, Font, Brushes.Black, 10, 10)

End Sub

Private Sub EliminarToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles EliminarToolStripMenuItem.Click 'Elimina texto

txtContenido.SelectedText = "" 'Elimina el texto que esta seleccionado solamente.

End Sub

Private Sub HoraFechaToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles HoraFechaToolStripMenuItem.Click 'Muestra la hora y fecha actual del sistema.

fecha = Date.Now.ToString("T")

dia = Date.Today.ToString("D")

txtContenido.Text = dia + " " + fecha

End Sub

Private Sub Form1\_FormClosing(sender As Object, e As System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing

If esGuardado Then 'Si el documento ha sido guardado, salimos de la aplicación

End

Else

Dim result As Integer = MsgBox("¿Quieres guardar el documento?", MsgBoxStyle.YesNo, Title:="Guardar documento")

If result = MsgBoxResult.Yes Then

If cadena = "" Then 'Si no existe el documento lo guardamos especificando ruta , nombre y extensión

GuardarcomoToolStripMenuItem\_Click(Nothing, Nothing)

Else

GuardarToolStripMenuItem\_Click(Nothing, Nothing) 'Si ya existía el documento guardamos los cambios y salimos de la aplicación

End If

End

ElseIf result = MsgBoxResult.No Then

End

End If

End

End If

esGuardado = True

End Sub

Private Sub IzquierdaToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles IzquierdaToolStripMenuItem.Click, ToolStripJustificarIzquierda.Click 'Justificamos texto a la isquierda

txtContenido.SelectionAlignment = HorizontalAlignment.Left

IzquierdaToolStripMenuItem.Checked = True

CentradoToolStripMenuItem.Checked = False

DerechaToolStripMenuItem.Checked = False

End Sub

Private Sub CentradoToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles CentradoToolStripMenuItem.Click, ToolStripButtonJustificarCentro.Click 'Justificamos texto al centro

txtContenido.SelectionAlignment = HorizontalAlignment.Center

IzquierdaToolStripMenuItem.Checked = False

CentradoToolStripMenuItem.Checked = True

DerechaToolStripMenuItem.Checked = False

End Sub

Private Sub DerechaToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles DerechaToolStripMenuItem.Click, ToolStripButtonJustificarDerecha.Click 'Justificamos texto a la derecha

txtContenido.SelectionAlignment = HorizontalAlignment.Right

IzquierdaToolStripMenuItem.Checked = False

CentradoToolStripMenuItem.Checked = False

DerechaToolStripMenuItem.Checked = True

End Sub

Private Sub GuardarComoPDFToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles GuardarComoPDFToolStripMenuItem.Click, btnGenerarPDF.Click 'Generamos un fichero PDF

SaveFilePDF.Filter = "Fichero PDF|\*.pdf"

If txtContenido.Text = "" Then

MsgBox("Debe introducir el texto a convertir a PDF.", MsgBoxStyle.Exclamation + MsgBoxStyle.OkOnly)

txtContenido.Focus()

Else

If txtContenido.Text = "" Then

MsgBox("Debe indicar el fichero PDF destino de la conversión del texto.", MsgBoxStyle.Exclamation + MsgBoxStyle.OkOnly)

txtContenido.Focus()

Else

Try

If SaveFilePDF.ShowDialog() = Windows.Forms.DialogResult.OK Then

SaveFilePDF.FileName = SaveFilePDF.FileName 'Mostramos diálogo de guardado

'Creamos el objeto documento PDF

Dim documentoPDF As New Document

PdfWriter.GetInstance(documentoPDF, New FileStream(SaveFilePDF.FileName, FileMode.CreateNew))

documentoPDF.Open()

'Escribimos el texto en el objeto documento PDF

documentoPDF.Add(New Paragraph(txtContenido.Text, FontFactory.GetFont(FontFactory.TIMES, 11, iTextSharp.text.Font.NORMAL)))

'Añadimos los metadatos para el fichero PDF

documentoPDF.AddCreator("Héctor Máximo Enciso Lezcano.")

'Cerramos el objeto documento, guardamos y creamos el PDF

documentoPDF.Close()

'Comprobamos si se ha creado el fichero PDF

If System.IO.File.Exists(SaveFilePDF.FileName) Then

If MsgBox("Texto convertido a fichero PDF correctamente " + "¿desea abrir el fichero PDF resultante?", MsgBoxStyle.Question + MsgBoxStyle.YesNo) = MsgBoxResult.Yes Then

'Abrimos el fichero PDF con la aplicación asociada

System.Diagnostics.Process.Start(SaveFilePDF.FileName)

End If

Else

MsgBox("El fichero PDF no se ha generado, " + "compruebe que tiene permisos en la carpeta de destino.", MsgBoxStyle.Exclamation + MsgBoxStyle.OkOnly)

End If

End If

Catch ex As Exception

MsgBox("Se ha producido un error al intentar convertir el texto a PDF: " + vbCrLf + vbCrLf + ex.Message, MsgBoxStyle.Critical + MsgBoxStyle.OkOnly)

End Try

End If

End If

'esGuardado = True

End Sub

Private Sub txtContenido\_SelectionChanged(sender As Object, e As System.EventArgs) Handles txtContenido.SelectionChanged 'Controlamos numero de linea y columnas en funcion de la posicion del cursor

posicion = txtContenido.SelectionStart 'Guardamos en posición el inicio del texto que tenemos seleccionado

linea = txtContenido.GetLineFromCharIndex(posicion) 'Guardamos en linea el numero de linea a partir de posicion

col = posicion - txtContenido.GetFirstCharIndexOfCurrentLine 'Guardamos en col la resta entre la posicion y el indice del primer caracter de la linea actual

lblLineasTotales.Text = "Líneas totales: " & txtContenido.Lines.Length 'Guardamos el número de lineas totales

lblColumna.Text = "Columna: " & (col + 1) 'Mostramos la columna en la que nos encontramos

lblLinea.Text = "Linea: " & (linea + 1) 'Mostramos la linea en la que nos encontramos

End Sub

Private Sub IrAToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles IrAToolStripMenuItem.Click 'Posicionamos el cursor en una linea concreta

Dim linea As Integer

linea = InputBox("Introduzca número de linea al que quiere ir: ") 'Guardamos numero linea introducido a buscar

Dim c As Long

For c = 0 To Len(txtContenido.Text) - 1 'Recorremos todo el RichTextBox

If txtContenido.GetLineFromCharIndex(c) >= (linea - 1) Then 'Restamos una a la linea introducida pues empieza en cero

txtContenido.SelectionStart = c 'Posicionamos el cursor en la posicion igual a la linea introducida

Exit For

End If

Next c

End Sub

Private Sub VistapreviadeimpresiónToolStripMenuItem\_Click(sender As System.Object, e As System.EventArgs) Handles VistapreviadeimpresiónToolStripMenuItem.Click 'Muestra dialogo de vista previa de impresión

PrintPreviewDialog1.Document = PrintDocument1 'PrintPreviewDialog asociado con PrintDocument.

PrintPreviewDialog1.ShowDialog() 'Abre vista previa de impresión

End Sub

Private Sub BorrarTodoToolStripMenuItem\_Click\_1(sender As System.Object, e As System.EventArgs) Handles BorrarTodoToolStripMenuItem.Click 'Borra contenido del RichTextBox

txtContenido.Clear()

End Sub

End Class

Public Class frmBusqueda

Dim index As Integer = 0 'Guardamos posicion de ocurrencia de cadena

Dim Pos, strSearchTerm, lastIndex, strLastSearch As Integer

Dim PalabraClave As String 'Guardamos texto de la palabra a buscar

Dim TipoBusqueda As Long

Private Sub btnBuscar\_Click(sender As System.Object, e As System.EventArgs) Handles btnBuscar.Click 'Busca ocurrencia de una palabra en el RichTextBox

If Form1.txtContenido.Find(txtBuscar.Text) = -1 Then 'Si no se encuentra el texto a buscar salta un mensaje indicando al usuario que la palabra no ha sido encontrada.

MsgBox("Texto no encontrado.", 48, "Atención")

Else

If txtBuscar.Text.Length > 0 Then 'Si se ha introducido texto en txtBuscar se busca y se superpone el Form1

Form1.txtContenido.Find(txtBuscar.Text)

Form1.Focus()

Else

MsgBox("No ha introducido texto para buscar.", 48, "Atención")

End If

End If

End Sub

Private Sub btnReemplazar\_Click(sender As System.Object, e As System.EventArgs) Handles btnReemplazar.Click 'Reemplaza una a una las ocurrencias de una cadena en el RichTextBox

If txtReemplazar.Text.Length > 0 Then 'Si se ha introducido texto en txtReemplazar

PalabraClave = txtBuscar.Text

If Len(PalabraClave) Then 'Si PalabraClave tiene longitud mayor que 0

Pos = InStr(Pos + 1, Form1.txtContenido.Text, PalabraClave, TipoBusqueda) 'Guardamos en posición la posicion que ocupa PalabraClave en txtContenido

If Pos > 0 Then

Form1.txtContenido.SelectionStart = Pos - 1

Form1.txtContenido.SelectionLength = Len(PalabraClave)

'Selecciona la longitud de la palabra a reemplazar

Form1.txtContenido.Focus() 'Superpone el Form1

Clipboard.Clear() 'Limpia el portapapeles

Clipboard.SetText(txtReemplazar.Text) 'Establece el texto que se encuentra en txtReemplazar

SendKeys.Send("^(V)") 'Envia simulando la pulsación CRTL+V (pegar el texto que se encuentra en el portapapeles)

Else

MsgBox("Texto no encontrado.", 48, "Atención")

End If

End If

Else

MsgBox("No ha introducido texto para reemplazar.", 48, "Atención")

End If

End Sub

Private Sub btnReemplazarTodo\_Click(sender As System.Object, e As System.EventArgs) Handles btnReemplazarTodo.Click 'Reemplazamos todos las ocurrencias de una cadena en el RichTextBox

If Form1.txtContenido.Text = Form1.txtContenido.Text.Replace(txtBuscar.Text, txtReemplazar.Text) = -1 Then

MsgBox("Texto no encontrado.", 48, "Atención")

Else

If txtReemplazar.Text.Length > 0 Then

Form1.txtContenido.Text = Form1.txtContenido.Text.Replace(txtBuscar.Text, txtReemplazar.Text) 'Reemplazamos la cadena vieja(txtBuscar.txt) por la cadena nueva(txtReemplazar.txt)

Form1.Focus()

Else

MsgBox("No ha introducido texto para reemplazar.", 48, "Atención")

End If

End If

End Sub

Private Sub btnBuscarSiguente\_Click(sender As System.Object, e As System.EventArgs) Handles btnBuscarSiguente.Click 'Buscamos la primera ocurrencia de una cadena y las siguiente hasta alcanzar el fin del documento

lastIndex = Form1.txtContenido.Text.LastIndexOf(txtBuscar.Text) 'Indica la posicion de la ultima ocurrencia de una cadena

If lastIndex = 0 Then 'Si la ultima ocurrencia en 0 quiere decir que esta al comienzo del documento

strSearchTerm = 0 'Establecemos a 0

ElseIf lastIndex = -1 Then 'Si no existe la ultima ocurrencia

If Form1.txtContenido.Text = "" Then 'Si el txtContenido está vacio

MsgBox("No se encontró '" & txtBuscar.Text & "'.", 48, "Atención")

Else

If Form1.txtContenido.Find(txtBuscar.Text, 0, Form1.txtContenido.Text.Length, RichTextBoxFinds.None) = -1 Then 'Si no está vacio buscamos de nuevo

MsgBox("No se encontró '" & txtBuscar.Text & "'.", 48, "Atención")

Else

strSearchTerm = Form1.txtContenido.Find(txtBuscar.Text, 0, Form1.txtContenido.Text.Length, RichTextBoxFinds.None) 'Si se encuentra establecemos strSearchTerm con el índice de inicio de la ocurrencia.

End If

End If

Else

strSearchTerm = Form1.txtContenido.Find(txtBuscar.Text, index, lastIndex, RichTextBoxFinds.None) 'Si nada de lo anterior de cumple establecemos strSearchTerm al índice de la ocurrencia

End If

If strSearchTerm = -1 Then

strLastSearch = Form1.txtContenido.Text.LastIndexOf(txtBuscar.Text) 'Si se encuentra pero es la útlima, repetimos el bucle

Form1.Focus()

Form1.txtContenido.SelectionStart = strLastSearch

Form1.txtContenido.SelectionLength = txtBuscar.Text.Length

index = 0

Else 'Si se encuentra y no es la última, establecer index a partir de la declaración Find para evitar la aparición anterior

Form1.Focus()

Form1.txtContenido.SelectionStart = strSearchTerm

Form1.txtContenido.SelectionLength = txtBuscar.Text.Length

index = strSearchTerm + 1

End If

End Sub

End Class