

Microsoft Visual Basic

GUÍA DE LABORATORIO N° 4

Objetivos

Luego de completar este laboratorio, el estudiante será capaz de:

- Manipular y realizar operaciones con arreglos.
- Crear una estructura y realizar operaciones con los campos de la misma.
- Implementar arreglos de estructuras.
- Trabajar con listas desplegables y cuadros combinados.

Aplicación N° 1

Elabore una aplicación que permita leer N números de tipo entero, y a continuación los visualice ordenados en forma ascendente o descendente.

The screenshot shows a Windows-style application window titled "Ordenación por burbuja". Inside the window, there are several controls: a text box labeled "Ingrese un nuevo número:" containing the value "14", a button labeled "Añadir", a list box labeled "Lista de números:" containing the values "2", "3", "14", "37", "68", and "97" (with "14" selected), a button labeled "Ordenar", and a button labeled "Salir" with a hand cursor icon. There are also radio buttons for "Orden:" with "Ascendente" selected and "Descendente" unselected.

Para el desarrollo de esta aplicación, proceda a ubicar los siguientes controles en el formulario:

- 3 GroupBox
- 1 caja de texto
- 1 control lista
- 2 botones de opción
- 3 botones de comando

En seguida proceda a establecer las propiedades según se indica:

Form1

Name	FrmBurbuja
BorderStyle	Fixed Dialog
Text	Ordenación por burbuja

GroupBox1

Name	GrpNumero
Text	Ingrese un nuevo número:

GroupBox2

Name	GrpLista
Text	Lista de números:

GroupBox3

Name	GrpOrden
Text	Orden:

Text1

Name	TxtNumero
Text	

ListBox1

Name	LstNumero
List	

Option1

Name	OptAscendente
Text	Ascendente
Checked	True

Option2

Name	OptDescendente
Text	Descendente
Checked	False

Command1

Name	BtnAnnadir
Text	&Añadir

Command2

Name	BtnOrdenar
Text	&Ordenar

Command3

Name	BtnSalir
Text	&Salir
Image	??

Una vez establecidas las propiedades proceda a ingresar el código que se indica a continuación:

```
Private Sub BtnAnadir_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Btn_Anadir.Click
    If IsNumeric(TxtNumero.Text) Then
        LstNumero.Items.Add(TxtNumero.Text)
        TxtNumero.Text = ""
        TxtNumero.Focus()
        TxtNumero.SelectAll()
    Else
        MessageBox.Show("Introduzca un número entero por favor")
        TxtNumero.Focus()
        TxtNumero.SelectAll()
    End If
End Sub
```

```
Private Sub Ordenar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles BtnOrdenar.Click

    Dim i As Integer, j As Integer, N As Integer
    N = LstNumero.Items.Count 'Devuelve numero de elementos en el listbox
    Dim Lista(N - 1) As Integer 'Los elementos de una matriz se empiezan a referenciar
en 0

    If N = 0 Then
        MessageBox.Show("No hay números en la lista")
    Else
        'Pasar datos del listbox a la matriz Lista
        For i = 0 To N - 1
            Lista(i) = LstNumero.Items.Item(i)
        Next
    End If

    '-----
    'Ordenar de menor a mayor
    If OptAscendente.Checked Then
        Dim aux As Integer = 0
        For i = 0 To N - 2
            For j = i + 1 To N - 1
                'Compara cada i con todo el array
                If Lista(i) > Lista(j) Then
                    'Intercambia si primero mayor
                    aux = Lista(i)
                    Lista(i) = Lista(j)
                    Lista(j) = aux
                End If
            Next j
        Next i
    End If
```

```

'-----
'Ordenar de mayor a menor
If OptDescendente.Checked Then
    Dim aux As Integer = 0
    For i = 0 To N - 2

        For j = i + 1 To N - 1
            'Compara cada i con todo el array
            If Lista(i) < Lista(j) Then
                'Intercambia si primero mayor
                aux = Lista(i)
                Lista(i) = Lista(j)
                Lista(j) = aux
            End If
        Next j
    Next i

End If
'-----
'Escribir los numeros ordenados en el listbox
LstNumero.Items.Clear() 'Vacía el listbox
For i = 0 To N - 1
    LstNumero.Items.Add(Lista(i))
Next i
End Sub

```

Aplicación N° 2

Elaborar una aplicación que permita seleccionar un artículo de un cuadro combinado (Combo). Apenas el usuario seleccione un artículo se debe mostrar el precio del mismo, el interés es fijo para esta ocasión.

El diseño de la interfaz debe ser similar a la siguiente figura:

La venta ha realizarse es a plazos, ello condiciona la cuota mensual a pagarse. Cuando se haga click sobre el botón Cuota mensual debe mostrarse un cuadro de diálogo con los datos propuestos:



De manera similar al hacer click sobre el botón Total nos debe mostrar la cantidad total a pagar.



Para el desarrollo de esta aplicación, proceda a ubicar los siguientes controles en el formulario:

- 1 GroupBox
- 3 etiquetas
- 1 cuadro combinado
- 2 cajas de texto
- 3 botones de opción
- 2 botones de comando

En seguida proceda a establecer las propiedades según se indica:

Form1

Name	FrmPedidoPagoAplazado
BorderStyle	Fixed Dialog
Text	Pedido pago aplazado

GroupBox1

Name	GrpPlazo
Text	Plazo:

Label1

Name	LblArticulo
Text	Artículo:

Label2

Name	LblPrecio
Text	Precio €

Label3

Name	LblInteres
Text	Interés:

ComboBox1

Name	CboArticulo
Text	

Text1

Name	TxtPrecio
ReadOnly	True
Text	

Text2

Name	TxtInteres
ReadOnly	True
Text	

Option1

Name	Opt6meses
Text	6 meses
Checked	True

Option2

Name	Opt12meses
Text	12 meses
Checked	False

Option3

Name	Opt24meses
Text	24 meses
Checked	False

Command1

Name	BtnCuotaMensual
Text	&Cuota mensual

Command2

Name	BtnTotal
Text	&Total

Una vez establecidas las propiedades de la interfaz, haga doble click sobre el formulario e ingrese las siguientes declaraciones en la sección General del módulo de formulario:

Private Structure Artículo

```
Dim Nombre As String
Dim Precio As Double
End Structure
Dim A(4) As Artículo, Plazo As Integer
Const Interes = 0.12
```

Recuerde que un dato declarado en la sección General de un módulo o clase puede ser accedido por todos los procedimientos de dicho módulo. Luego, continúe ingresando el código que se muestra a continuación:

```
Private Sub FrmPedidoPagoAplazado_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    A(0).Nombre = "Monitor SAMSUNG Syncmaster 3"
    A(1).Nombre = "Impresora Hewlet Packard Deskjet 930C"
    A(2).Nombre = "Impresora Epson Stylus Color 740"
    A(3).Nombre = "Microprocesador Pentium I 233 MHz"
    A(0).Precio = 150 : A(1).Precio = 275 : A(2).Precio = 145 : A(3).Precio = 80
    'Cargar los datos del array en el combobox
    Dim i As Integer
    For i = 1 To 4
        CboArticulo.Items.Add(A(i - 1).Nombre)
    Next
    'Inicializamos Intere y plazo
    TxtInteres.Text = CStr(Interes) : Plazo = 6
End Sub
```

```
Private Sub CboArticulo_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles CboArticulo.SelectedIndexChanged
    Dim i As Integer
    i = CboArticulo.SelectedIndex 'Indice del elemento seleccionado
    TxtPrecio.Text = CStr(A(i).Precio) 'Escribe el precio del elemento seleccionado
End Sub
```

```

Private Sub BtnCuota_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles BtnCuota.Click
    Dim Total As Double, CuotaMensual As Double, i As Integer
    i = CboArticulo.SelectedIndex 'Indice del elemento seleccionado
    Total = A(i).Precio * (1 + Interes)
    If Opt6meses.Checked Then
        Plazo = 6
    ElseIf Opt12meses.Checked Then
        Plazo = 12
    ElseIf Opt24meses.Checked Then
        Plazo = 24
    End If
    CuotaMensual = Total / Plazo
    MessageBox.Show("Cuota mensual: " & CuotaMensual.ToString("#,##0.00 €"))
End Sub

Private Sub BtnTotal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles BtnTotal.Click
    Dim Total As Double, i As Integer
    i = CboArticulo.SelectedIndex 'Indice del elemento seleccionado
    Total = A(i).Precio * (1 + Interes)
    MessageBox.Show("Total: " & Total.ToString("#,##0.00 €"))
End Sub

```


Modificación de la aplicación 1 para usar procedimientos

Escribir el siguiente procedimiento en el código de la aplicación 1, justo antes de End Class.

```
Sub Ordenar_Ascendente(ByVal vector() As Integer)
    Dim aux As Integer = 0
    Dim N As Integer = vector.Length

    For i = 0 To N - 2
        For j = i + 1 To N - 1
            'Compara cada i con todo el array
            If vector(i) > vector(j) Then
                'Intercambia si primero mayor
                aux = vector(i)
                vector(i) = vector(j)
                vector(j) = aux
            End If
        Next j
    Next i
End Sub
```

Sustituir en el código de la aplicación 1, este código:

```
'Ordenar de menor a mayor
If OptAscendente.Checked Then
    Dim aux As Integer = 0

    For i = 0 To N - 2
        For j = i + 1 To N - 1
            'Compara cada i con todo el array
            If Lista(i) > Lista(j) Then
                'Intercambia si primero mayor
                aux = Lista(i)
                Lista(i) = Lista(j)
                Lista(j) = aux
            End If
        Next j
    Next i
End If

Por este otro:
'Ordenar de menor a mayor
If OptAscendente.Checked Then
    Ordenar_Ascendente(Lista)
End If
```

Comprobar que la aplicación funciona correctamente.

De la misma forma, crear un procedimiento que ordene de forma descendente un vector de números.

Aplicación N° 4

Desarrollar una aplicación que permita realizar consultas acerca de un determinado curso, los cuales se mostraran en una lista. El usuario debe seleccionar un curso y en seguida se debe presentar el Name del profesor encargado del curso (teoría), el Name del jefe de práctica (laboratorio), así como los horarios de teoría y de laboratorio. El diseño de la interfaz deberá ser similar al siguiente:

Consulta de cursos

Profesor de teoría: Castillo Peralta, Carlos

Horario de teoría: Sa 10-13

Jefe de práctica: Córdoba Saavedra, Javier

Horario de laboratorio: Do 08-10

Cursos:

- Microsoft Visual Basic Nivel II
- Microsoft Visual FoxPro Nivel I
- Microsoft Visual FoxPro Nivel II
- Microsoft Visual C++ Nivel I
- Microsoft Visual C++ Nivel II
- Microsoft Visual J++ Nivel I
- Microsoft Visual J++ Nivel II
- Microsoft SQL Server Nivel I
- Microsoft SQL Server Nivel II

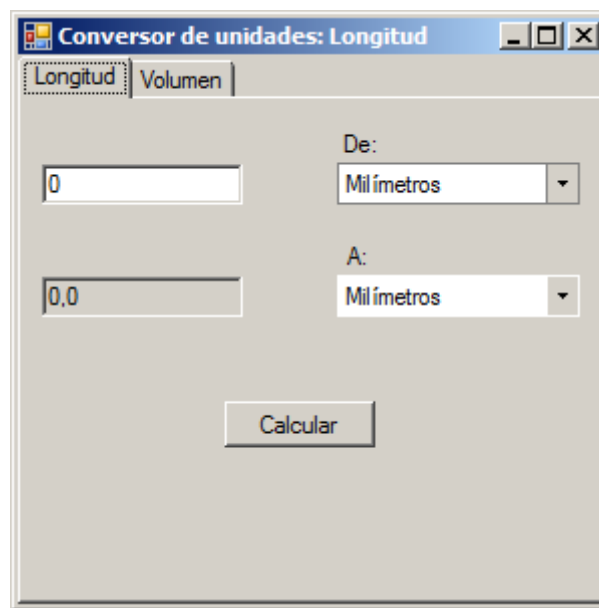
Aplicaciones propuestas:

Implementar una aplicaciones que realice conersiones de unidades.

Por ejemplo de mm a pulgadas, kilos a onzas, etc.

Para ello utilizar una ficha donde se realicen las conversiones de longitud otra para las de volumen y otra para las de área.

La interfaz podría ser la siguiente.



Implementar una aplicación que dadas las temperaturas de un mes Calcule la media y nos la muestre junto con una imagen de sol o nubes según la media sea muy alta o demasiado baja.