



GUIA DE LABORATORIO 9 - PRESENCIAL

Microsoft Excel 2016 Professional (Avanzado)

Entrega de guía en la Semana: del 05 de Octubre al 09 de Octubre 2020

(Según día de Laboratorio y 24 horas antes de su hora de revisión).

Recordatoria: 1. Podrá Reservar Consulta para Guía de Laboratorio. 2. Deberá conservar Reserva de Revisión de la Guía de Laboratorio 3.

Objetivo:

- Analizar y comprender los problemas presentes y codificarlos en base al diseño de la solución (flujogramas)
- Desarrollar mayor destreza al trabajo con Visual basic para Aplicaciones.
- Crear funciones definidas por el usuario en las macros de Excel.
- Crear macros utilizando formularios y controles en VBA.
- Conocer y trabajar parte de los objetos utilizados dentro de un formulario y sus eventos, programando estos últimos para que interactúen entre sí, así como invocar un formulario a través de otro.

Indicaciones

- Esta guía está diseñada para ser desarrollada desde la semana 7 hasta la semana 9 antes de la sesión de laboratorio, tendrá que entregarse a más tardar 24 horas antes de su hora de laboratorio en base a las indicaciones de entrega arriba señaladas. Recuerde que a partir de la guía 7 debe subir archivo para tener derecho a la revisión en el horario de reserva.
- Para tener derecho a la nota de entrega de la guía de laboratorio 9, es necesario asistir a la Reserva de su laboratorio como se ha establecido en las guías, subir y entregar para revisión su archivo en el enlace correspondiente creado para este fin antes de su laboratorio inscrito.
- La entrega del archivo corresponde al desarrollo de la guía de laboratorio completamente terminada. Si esta es entregada parcialmente, tendrá una nota de entrega parcial. Esto incluye los ejercicios propuestos. Recuerde realizar las entregas de forma gradual conforme va avanzando en el desarrollo de la guía, para que no tenga algún problema de última hora.
- En la sesión del laboratorio de la semana 9, el Instructor revisará frente al estudiante los ejercicios de la guía 9 y hará preguntas al respecto para corroborar que el estudiante la haya realizado y asignado la nota correspondiente a la revisión. Como se ha establecido y realizado en las sesiones anteriores, las notas de las guías se dividen en un 50% para la revisión de la guía y un 50% para la entrega de archivos según criterio de revisión del docente y tipo de entrega parcial o total.

PROCEDIMIENTOS

Práctica de PROCEDIMIENTOS

Abra un libro en Excel y nombre al libro como "**MacrosB.xlsm**" y llame la Hoja1 como "**Function1**" donde se mostrará la tabla resultante. Realice los pasos necesarios para diseñar la función definida por el usuario que nos permite calcular el valor de la función $g(x)$.

$$g(x) = 2x^3 + \ln(x) - \frac{\cos(x)}{e^x} + \sin(x)$$

Tabla Resultante:

FECHANU...		: X ✓ f _x		=g(A9)
	A	B		
1	x	g = 2 * x ^ 3 + Log(x) - Cos(x) / Exp(x) + Sin(x)		
2	0.5	-0.50		
3	1	2.64		
4	1.5	8.14		
5	2	17.66		
6	2.5	32.83		
7	3	55.29		
8	3.5	86.68		
9	4	=g(A9)		

PROCEDIMIENTO

A continuación, se describe en detalle el procedimiento para crear una función en un módulo que nos permitirá calcular el área de un hexágono. Solo tenemos que seguir cada uno de los pasos indicados.

1. En el libro, ingrese al Editor de Visual Basic (alt+F11)
2. Inserte un módulo, seleccionando del menú **Insertar Módulo**
3. En la ventana Código, escriba el nombre a la función como Function y, a continuación, dejando un espacio, el nombre del procedimiento. Para Nuestro caso introduzca en el área de código **Function g.**

- Una vez ingresado el nombre, presione ENTER. El editor de VBA añade unos paréntesis después del nombre inserta y además una nueva línea con las instrucciones End Function.
- A continuación, dentro de los parámetros que agregó VBA, coloca el parámetro x.
- Entre estas dos líneas, escriba el código que sigue, que contiene la fórmula para calcular el valor de la función **g(x)**. Para ello introduzca la siguiente línea:

```
Function g(x)
    g = 2 * x ^ 3 + Log(x) - Cos(x) / Exp(x) + Sin(x)
End Function
```

Nota: La sintaxis de la función g(x) en vba es diferente a la de Excel, La función debe de trabajarse para poderla expresar en la sintaxis correcta para vba.

- En Excel la sintaxis es: $2*B3^3+LN(B3)-COS(B3)/EXP(B3)+SENO(B3)$
- En VBA la sintaxis es $2 * x ^ 3 + Log(x) - Cos(x) / Exp(x) + Sin(x)$

Observe, por ejemplo, que la función logaritmo natural $\ln(x)$, en Excel se escribe LN mientras que en VBA se escribe Log. Ahora Guarde el Libro: " **MacrosB.xlsm** ".

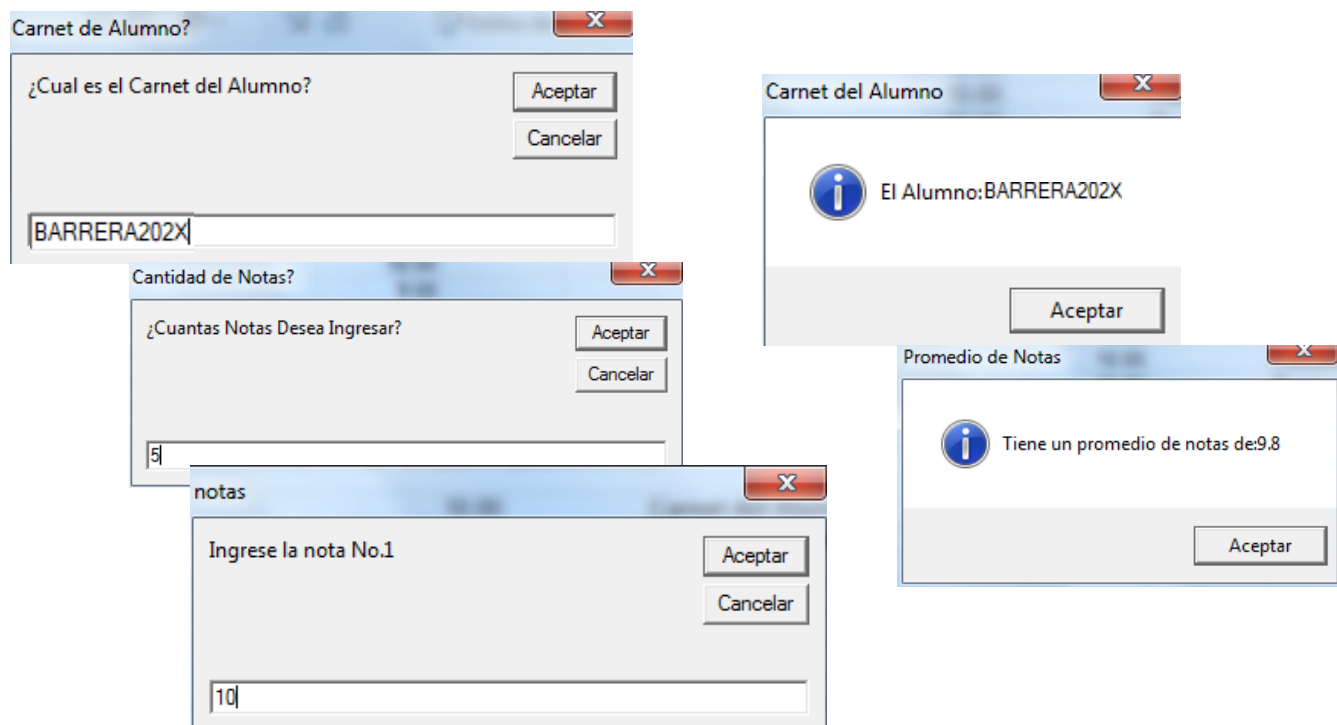
- Las funciones definidas por el usuario las podemos visualizar dando clic en Pestaña **Formulas** y buscamos en categorías de la función la que diga funciones **Definidas por el usuario** y encontraremos la función **g**.
- La función **g** nos pide un parámetro (el valor de x), que para nuestro caso deberá de ubicarse en la celda B2, pulsar "**=g()**" y seleccionar con el puntero la celda A2, luego ENTER y listo. Obtendremos el resultado que se muestra en la imagen anterior. Para completar la tabla resultante basta con copiar y arrastrar desde B2 hasta B9.
- Nuevamente Guarde los cambios efectuados en el libro " **MacrosB.xlsm** ". La hoja es resultante de la muestra de la página 1. Antes de guardar asegúrese de haber nombrado la hoja como " **Function1** ".

FLUJOGRAMAS CON LENGUAJE VISUAL BASIC PARA APLICACIONES (VBA)

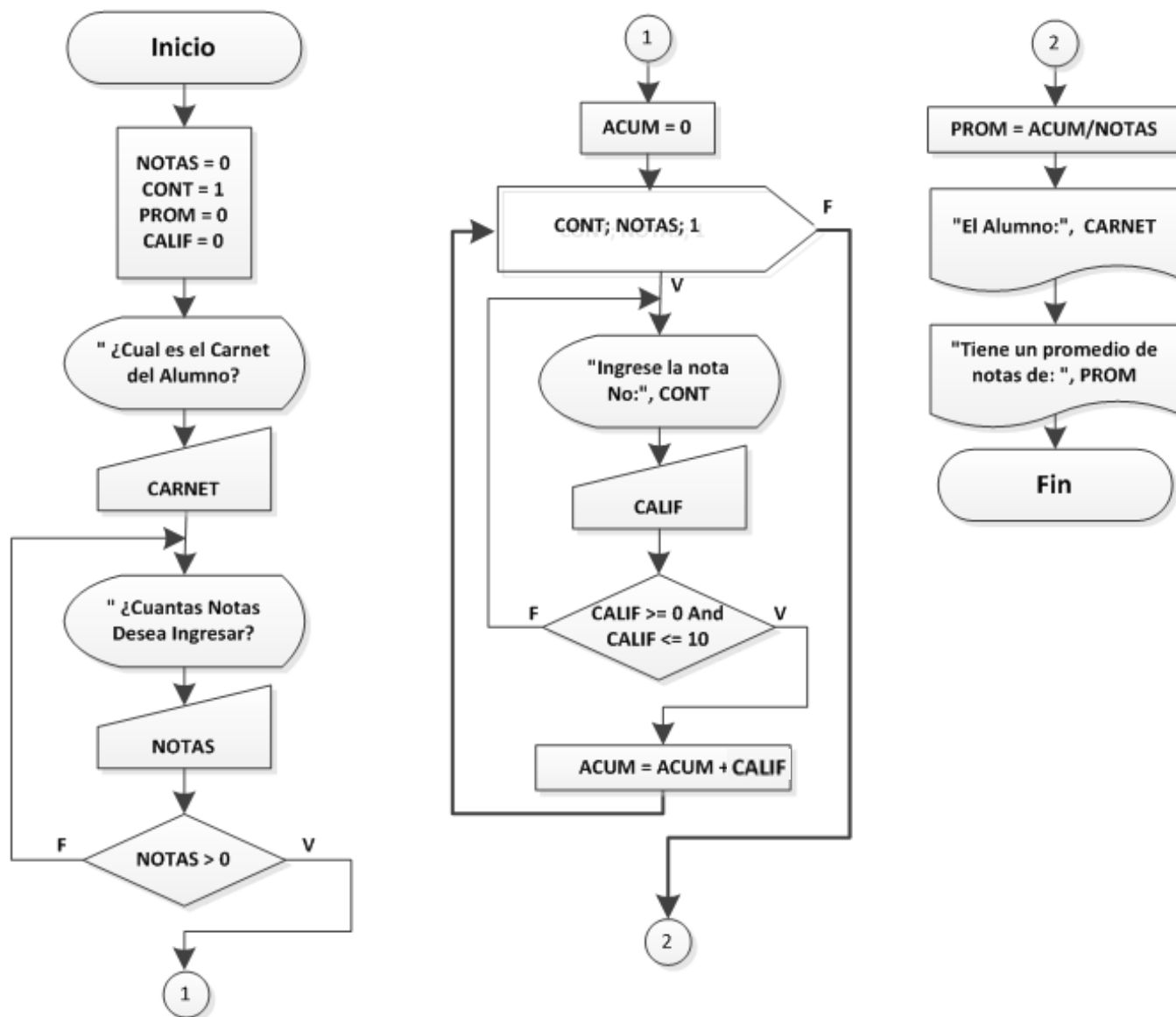
INDICACIONES: Revise la solución del siguiente ejercicio, luego digite el Diseño de Solución, pruébelo y corrobore los resultados. Para ello proceda realizar un **Módulo** que diseñe una solución que permita introducir el Carnet de un alumno y una cantidad de notas desconocida para calcular el promedio de dichas notas. Los resultados se mostrarán en la hoja de Excel denominada " **Notas** " que se encontrará en nuestro libro de trabajo " **BaseGuía9** " como " **Hoja1** " que puede descargar desde el curso de msm115, del aula virtual. El modulo deberá asignar al objeto de **WortArt** que se lee como " **Notas Del Alumno** " ubicado en la hoja de inserción de datos. Además de una tecla rápida " **n** ". Corrobore los resultados con los siguientes Datos de Prueba: **Su_Carnet, 5, 10.0, 10.0, 10.0, 10.0, 9.0**.

	A	B	C	D	E
1	NOTAS DEL ALUMNO				
2	Nota1	10.00	Carnet del Alumno:	BARRERA202X	
3	Nota2	10.00	Cantidad de Notas:	5.00	
4	Nota3	10.00	Promedio:	9.80	
5	Nota4	10.00			
6	Nota5	9.00			
7					

Resultado esperado:



DISEÑO DE LA SOLUCIÓN: Codifique en VBA el siguiente Flujograma



PROCEDIMIENTO

En el **Paso a paso**, que se presenta a continuación, describimos en detalle el procedimiento que debemos realizar para crear la solución en un módulo que nos permitirá calcular el promedio de notas. Solo tenemos que seguir cada uno de los pasos indicados.

1. Abre nuestro libro de Trabajo "**BaseGuía9**" e ingresa a la Hoja1 y renombra a la Hoja como "**Notas**"
2. Configura las celdas donde se mostrarán los datos a ingresar según la muestra de la tabla de la página 2. Es decir, no llenes la tabla con datos, solo aplica negritas y dos decimales donde corresponda.
3. Ingrese al Editor de Visual Basic con la combinación de teclas (Alt+F11)
4. Inserte un módulo, seleccionando del menú **Insertar Módulo**
5. En la ventana Código, escriba al inicio del código el nombre del Procedimiento comenzando con Sub y a continuación, dejando un espacio, el nombre del procedimiento. Para Nuestro caso le asignaremos como nombre "**Promedio_de_Notas**".
6. Una vez ingresado el nombre, presione ENTER. El editor de VBA añade unos **paréntesis** después del nombre insertado y además agrega una nueva línea con las instrucciones **End Sub**.
7. A continuación, dentro de los parámetros que agregó VBA, coloca los parámetros que recibe la función (si existen), para nuestro caso no existe ningún parámetro.
8. Entre estas dos líneas, escriba el código correspondiente al diseño de la solución del problema, el cual sirve para calcular el promedio de las notas de un alumno. Para ello introduzca las siguientes líneas:


LINEAS DE CODIGO:

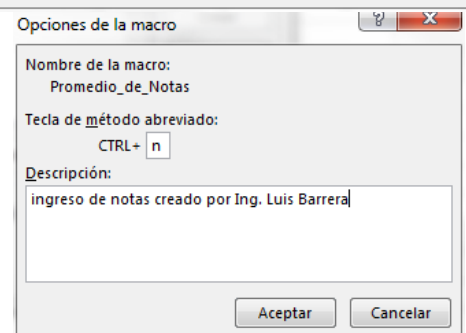
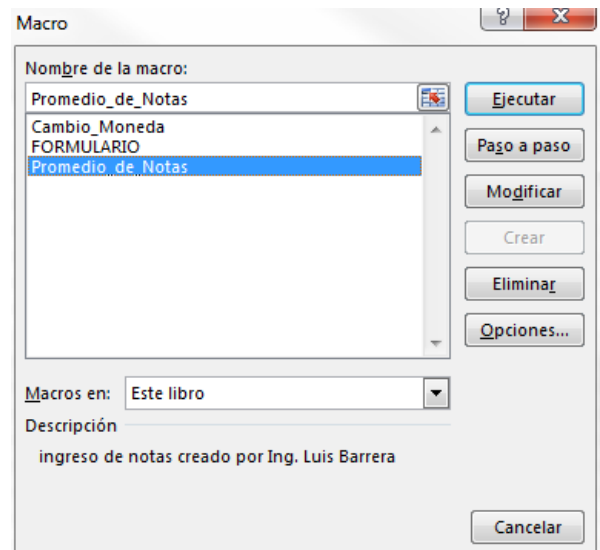
Sub Promedio_de_Notas()

```
'Declaración de variables
Dim Notas As Integer
Dim Acum As Double
Dim Cont As Integer
Dim Calif As Double
Dim Prom As Double
Dim Carnet As String
'
Carnet = InputBox("¿Cual es el Carnet del Alumno?", "Carnet de Alumno?")
'
Do
    ' Cuadro de dialogo que nos indica que tipo de datos hay que introducir
    Notas = InputBox("¿Cuántas Notas Desea Ingresar?", "Cantidad de Notas?")
    Loop Until (Notas > 0)
    Acum = 0 'Inicialización
    For Cont = 1 To Notas
        Do
            Calif = InputBox("Ingrese la nota No." & CStr(Cont), "notas")
            Loop Until (Calif >= 0 And Calif <= 10)
            ThisWorkbook.Sheets("Notas").Select
            ActiveSheet.Range("A" & CStr(Cont + 1)).Value = "Nota " & CStr(Cont)
            ActiveSheet.Range("B" & CStr(Cont + 1)).Value = Calif
            Acum = Acum + Calif
        Next
    Prom = Acum / (Cont - 1)
    MsgBox "El Alumno: " & CStr(Carnet), vbOKOnly + vbInformation, "Carnet del Alumno"
    MsgBox "Tiene un promedio de notas de: " & CStr(Prom), vbOKOnly + vbInformation, "Promedio de Notas"

    ActiveSheet.Range("D2").Value = "Carnet del Alumno: "
    ActiveSheet.Range("D3").Value = "Cantidad de notas: "
    ActiveSheet.Range("D4").Value = "Promedio: "
    ActiveSheet.Range("E2").Value = Carnet
    ActiveSheet.Range("E3").Value = Notas
    ActiveSheet.Range("E4").Value = Prom
```

End Sub

9. Antes de **ejecutar** el procedimiento con el botón  verifique el formato de las celdas de hoja que usted renombro como **"Notas"** (este método se ejecuta al encontrarse en VBA). Donde aparecerá la ventana de ingreso de datos. Además de ello, debemos ubicarnos en la hoja donde deseamos que aparezcan nuestros datos (para nuestro caso sería la Hoja **"Notas"**) o colocar dentro del código la línea de comando `ThisWorkbook.Sheets("Notas").Select` para ir a ubicar los datos en la hoja de Notas. <<Pregunte a su instructor si lo considera necesario, al no comprende parte del código>>.
10. Para ejecutar la macro desde nuestro libro de trabajo, podemos hacerlo aplicando diferentes métodos: a) Debemos presionar la combinación de teclas ALT+F8 para ver las opciones de las macros creadas en nuestro libro, el cual deberá seleccionar la de nuestro interés. Para nuestro caso **"Promedio_de_Notas"** y luego el botón **Ejecutar** de la Ventana de comandos emergente de Macros.
11. **Modificación** del Procedimiento, b) Otro método para ejecutar nuestro Macro es Seleccionando un método abreviado para invocar la macro que contiene nuestro Procedimiento, esto se hará con las teclas combinadas **CTRL + n**. Para realizar este cambio tendrá que presionar la combinación de teclas ALT+F8, seleccionar nuestra macro y a continuación el comando **Opciones...** de la ventana de comandos **Macros**. En Opciones de la macro, deberá agregar la letra **"n"** y una descripción a la macro como la que se muestra en la imagen y luego **Aceptar**.
12. c) **Agregue un método más** para la ejecución del procedimiento **asignando** a un objeto de **WordArt** la invocación de la macro que contiene nuestro Procedimiento que se ubica en la hoja donde se debe de mostrar los datos de nuestro ejercicio. Para nuestro caso el Objeto **"Notas Del Alumno"** tendrá la función de invocar a la macro que contiene el procedimiento, solo basta ubicar el puntero sobre el objeto y pulsa clic y listo. Para la asignación aplica el mismo procedimiento realizado para ejecutar los formularios, **aprendido con los ejercicios de la guía de laboratorio 8**. Ahora ya puede ejecutar macros que contienen procedimientos con cualquiera de los métodos estudiados.



13. Al ejecutar el procedimiento aparecerán los cuadros de texto que se muestran en la página 2. Corrobore los resultados con los siguientes Datos de Prueba: **Su_Carnet, 5, 10.0, 10.0, 10.0, 10.0, 9.0. Aparecen los resultados esperados.**

Si en algún momento aparece la pantalla como la que se muestra, no se preocupe, pueda que haya ejecutado el procedimiento y no haya ingresado ningún dato, para tal caso seleccionar **Finalizar**. Si a la hora de ejecutar el procedimiento existe una interrupción similar a la muestra (al estar probando los datos) donde aparece esa misma pantalla, probablemente exista un error en una línea de código. Para identificarla solo basta con pulsar **Depurar** e inmediatamente le señala con color amarillo la línea donde probablemente se encuentra el error. Al corregirlo vuelva a ejecutar la macro que invoca al procedimiento.

Microsoft Visual Basic

Se ha producido el error '13' en tiempo de ejecución:

No coinciden los tipos

Continuar

Finalizar

Depurar

Ayuda

14. Ahora Guarde el Libro y renómbrelo "**MacrosC.xlsm**", recuerde guárdalo como **un libro de Excel habilitado para Macros**. La hoja es el resultado de la muestra de la página 2, donde se encuentran los datos de pruebas (ingresados) y además los cálculos de los mismos que se encuentra listo para ser impresos y conforme al flujograma codificado de la página 3. Antes de guardar asegúrese de haber nombrado la hoja como "**Notas**". Al finalizar su libro tendrá una extensión **.xlsm**.

Otro ejercicio:

INDICACIONES: Revise la solución del siguiente ejercicio, luego digite el código del Diseño de Solución, pruébelo y corrobore los resultados. Para ello procederemos a realizar un **Módulo** que diseñe una solución que permita ingresar un monto en dólares americanos y permita conocer su equivalente ya sea en: Lempiras, Quetzales, Córdoba oro y Colones Costarricenses. Sabiendo que: Un Dólar equivale a: 17.16 Lempiras, 7.94 Quetzales, 14.90 Córdobas oro y 389.16 Colones Costarricenses. Los resultados se mostrarán en la hoja de Excel denominada "**Convertir**" que se encontrará en nuestro libro de trabajo "**MacrosC**" como "**Hoja2**". El módulo deberá asignar al objeto de **WortArt** que se lee como "**Conversión**" ubicado en la hoja de inserción de datos. Además de una tecla rápida "**r**". Corrobore los resultados con los siguientes Datos de Prueba: **100, r**

Resultado esperado:

	A	B	C	D	E
1	Conversión				
2					
3					
4					
				Monto en Dólares:	\$100.00
				Equivalente a Colones Costarricenses:	38,916.00

US\$ a convertir

Digite el Monto en Dolares a convertir (US\$)

Aceptar

Cancelar

100

Costarricenses



Equivale a \$38,916.00 Colones Costarricenses

Aceptar

Elección de Tipo de Moneda?

Elija a que moneda desea convertir:

(L) Lempiras

(Q) Quetzales

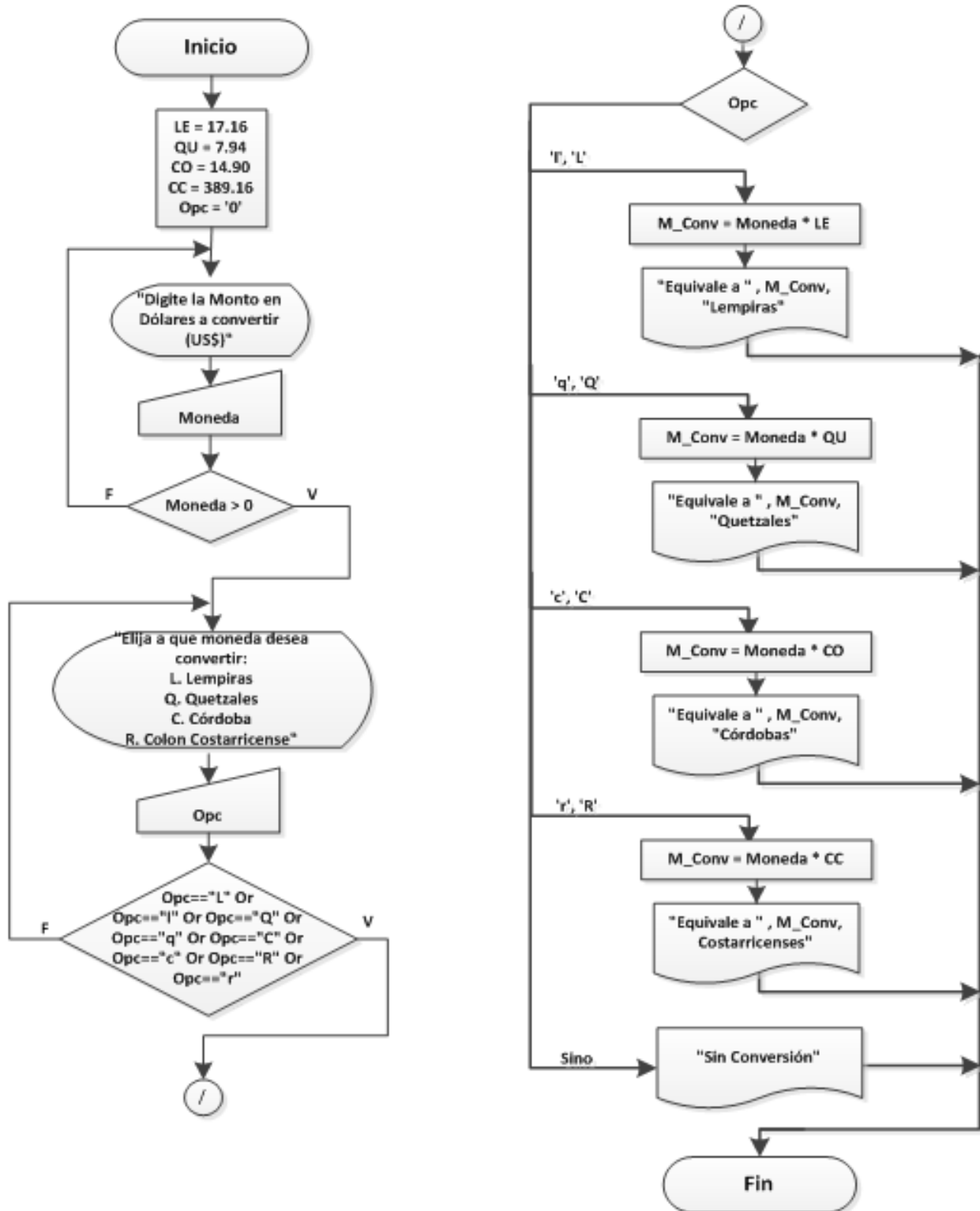
(C) Córdoba

(R) Colon Costarricense

Aceptar

Cancelar

R



PROCEDIMIENTO

En el **Paso a paso**, que se presenta a continuación, describimos en detalle el procedimiento que debemos realizar para crear la solución en un módulo que nos permitirá calcular la conversión de las monedas. Solo tenemos que seguir cada uno de los pasos indicados.

1. Continuando en nuestro libro de Trabajo "**MacrosC**" ingrese a la Hoja2 y renombra a la Hoja como "**Convertir**"
2. Configura las celdas donde se mostrarán los datos a ingresar según la muestra de la tabla de la página 5. Es decir, no llenes la tabla con datos, solo aplica negritas y dos decimales donde corresponda.
3. Ingrese al Editor de Visual Basic con la combinación de teclas (Alt+F11) o desde la cinta de **Desarrollador**
4. Inserte un módulo, seleccionando del menú **Insertar Módulo**
5. En la ventana Código, escriba al inicio del código el nombre del Procedimiento comenzando con **Sub** y a continuación, dejando un espacio, el **nombre del procedimiento**. Para Nuestro caso le asignaremos como nombre "**Cambio_Moneda**".

- Una vez ingresado el nombre, presione **ENTER**. El editor de VBA añade unos **paréntesis** después del nombre insertado y además agrega una nueva línea con las instrucciones **End Sub**.
- A continuación, dentro de los parámetros que agregó VBA, coloca los parámetros que recibe la función (si existen), para nuestro caso no existe ningún parámetro.
- Entre estas dos líneas, escriba el código correspondiente al diseño de la solución del problema, el cual sirve para calcular el valor de cambio entre monedas. Para ello introduzca las siguientes líneas:

LINEAS DE CODIGO:


```

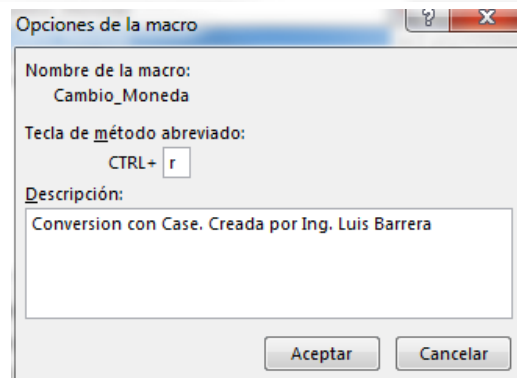
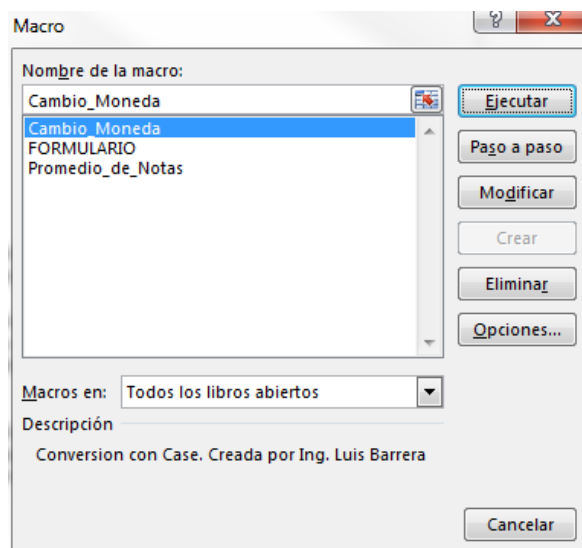
Sub Cambio_Moneda()
'Declaración de Constantes Locales
Const LE As Double = 17.16
Const QU As Double = 7.94
Const CO As Double = 14.9
Const CC As Double = 389.16
'Declaración de Variables Locales
Dim Moneda As Double
Dim M_Conv As Double
Dim Opc As String
Dim Mens As String
'Validación de Moneda
Do
' Cuadro de dialogo que nos indica que tipo de datos hay que introducir
Moneda = InputBox("Digite el Monto en Dólares a convertir (US$)", "US$ a convertir")
Loop Until (Moneda > 0)
'Validación de Opción
Do
Opc = InputBox("Elija a que moneda desea convertir:
                & " (L) Lempiras
                & " (Q) Quetzales
                & " (C) Córdoba
                & " (R) Colon Costarricense ", "Elección de Tipo de Moneda?")
Loop Until Opc = "L" Or Opc = "I" Or Opc = "Q" Or Opc = "q" Or Opc = "C" Or Opc = "c" Or Opc = "R" Or Opc = "r"
'Selecciona el tipo de Conversión en base a una estructura de casos
Select Case Opc
Case "L", "I"
M_Conv = Moneda * LE
MsgBox "Equivale a " & CStr(M_Conv) & " Lempiras", vbOKOnly + vbInformation, "Lempiras"
Mens = " Lempiras: "
Case "Q", "q"
M_Conv = Moneda * QU
MsgBox "Equivale a " & CStr(M_Conv) & " Quetzales", vbOKOnly + vbInformation, "Quetzales"
Mens = " Quetzales: "
Case "C", "c"
M_Conv = Moneda * CO
MsgBox "Equivale a " & CStr(M_Conv) & " Córdoba", vbOKOnly + vbInformation, "Córdoba"
Mens = " Córdoba: "
Case "R", "r"
M_Conv = Moneda * CC
MsgBox "Equivale a " & CStr(M_Conv) & " Colones Costarricenses", vbOKOnly + vbInformation, "Costarricenses"
Mens = " Colones Costarricenses: "
Case Else
MsgBox "Sin Conversión"
Mens = "Sin Conversión"
End Select
'ubicación de los datos
ThisWorkbook.Sheets("Convertir").Select
ActiveSheet.Range("D2").Value = "Monto en Dólares: "
ActiveSheet.Range("D3").Value = " Equivalente a" & Mens
ActiveSheet.Range("E2").Value = Moneda
ActiveSheet.Range("E3").Value = M_Conv
End Sub

```

Investigue que funciones se podría utilizar para:

- Convertir texto en mayúscula o minúscula y hacer más corto el código correspondiente a la solución anterior.
- Convertir los valores número en formato moneda la variable de Tipo de Moneda
- Realice los cambios necesarios para que la estructura del ejercicio anterior aplique una estructura Loop While en las validaciones.

9. Antes de **ejecutar** el procedimiento con el botón  verifique el formato de las celdas de hoja que usted renombro como "**Convertir**" (este método se ejecuta al encontrarse en VBA). Donde aparecerá la ventana de ingreso de datos. Además de ello, debemos ubicarnos en la hoja donde deseamos que aparezcan nuestros datos (para nuestro caso sería la Hoja "**Convertir**").
10. Para ejecutar la macro desde nuestro libro de trabajo, podemos hacerlo aplicando diferentes métodos: a) Debemos presionar la combinación de teclas ALT+F8 para ver las opciones de las macros creadas en nuestro libro, el cual deberá seleccionar la de nuestro interés. Para nuestro caso "**Cambio_Moneda**" y luego el botón **Ejecutar** de la Ventana de comandos emergente de Macros.
11. **Modificación** del Procedimiento, b) Otro método para ejecutar nuestro Macro es Seleccionando un método abreviado para invocar la macro que contiene nuestro Procedimiento, esto se hará con las teclas combinadas **CTRL + r**. Para realizar este cambio tendrá que presionar la combinación de teclas ALT+F8, seleccionar nuestra macro y a continuación el comando **Opciones...** de la ventana de comandos **Macros**. En Opciones de la macro, deberá agregar la letra "**r**" y una descripción a la macro como la que se muestra en la imagen y luego **Aceptar**.
12. c) **Agregue un método más** para la ejecución del procedimiento **asignando** a un objeto de **WordArt** la invocación de la macro que contiene nuestro Procedimiento que se ubica en la hoja donde se debe de mostrar los datos de nuestro ejercicio. Para nuestro caso el Objeto "**Conversión**" tendrá la función de invocar a la macro que contiene el procedimiento, solo basta ubicar el puntero sobre el objeto y pulsa clic y listo.
13. Al ejecutar el procedimiento aparecerán los cuadros de texto que se muestran en la página 5. Corrobore los resultados con los siguientes Datos de Prueba: **100, r**. Aparecen **los resultados esperados**.
14. Ahora Guarde el Libro "**MacrosC.xlsm**", recuerde que debe guardarlo como un **Libro de Excel habilitado para Macros**. La hoja es el resultado de la muestra de la página 5, donde se encuentran los datos de pruebas (ingresados) y además los cálculos de los mismos que se encuentra listo para ser impresos y conforme al flujograma codificado de la página 6. Antes de guardar asegúrese de haber nombrado la hoja como "**Conversión**".



LOS USERFORMS – FORMULARIOS Y MÁS CONTROLES

Ejemplo 4:

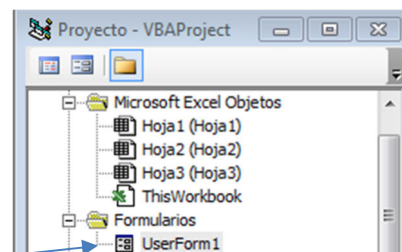
Procederemos a realizar un **Módulo** para cada Ejecutar cada formulario que nos permita conocer más de los objetos de los formularios. Para este caso los resultados se mostrarán en una hoja de Excel denominada "**ResultadoF1**" que se encontrará en nuestro libro de trabajo "**MacrosC.xlsm**" como "**Hoja3**" a la cual deberá renombrar. El formulario deberá asignar al objeto que se lee como "**De Clic Aquí**" ubicado en la hoja de inserción. Recuerde que estos 2 últimos puntos los aprendió en el laboratorio anterior que deberá consultar de no recordarse. Usted deberá analizar qué es lo que hace cada Estructura del código al ejecutar el formulario y complementar la documentación del código para aquellas estructuras principales. <<Pregunte a su instructor si lo considera necesario, al no comprende parte del código>>.

Resultado Esperado: para el caso de la imagen de la muestra se ha seleccionado la opción 2 y opción 7 respectivamente.

	A	B
1		Elección
2	Opción del Frame 1	Opción 2!!!!
3	Opción del Frame 2	Opción 7!!!!

PROCEDIMIENTO

1. Descargue del curso y copie en la unidad de disco C:/ la carpeta de "IMAGENES-GLXX". Descárguela y guárdela, y coloque en sustitución de **XX** su grupo de laboratorio
2. Presione La Teclas **Alt + F11**, para entrar al editor de **Visual Basic**.
3. Activa las siguientes opciones:
 - De clic en el **Menú Ver** y elija la opción **Explorador de Proyectos**
 - De clic en el **Menú ver** y elija la opción **Ventana Propiedades**
4. Del **Menú Insertar** elija la Opción **UserForm**. Esto inserta el Formulario que programaremos con controles.
5. En el **Explorador de Proyecto** se observará que se insertó el **UserForm**.



6. Ahora crearas un formulario con el siguiente aspecto:

3 Marcos Dentro de cada Marco tendrá:

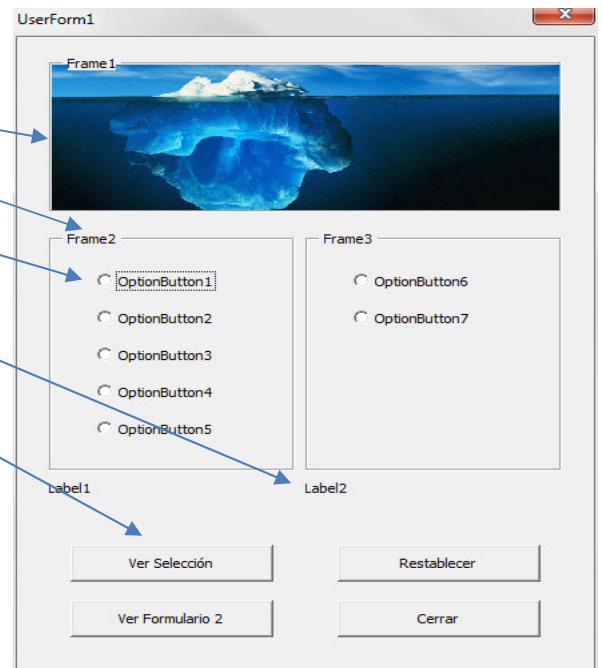
- Frame 1: Posea una Imagen de Fondo
- Frame 2: Posea 5 Botones de Opciones
- Frame 3: Posea 2 Botones de Opciones

2 Etiquetas: Que muestren la elección de Frame 2 y 3

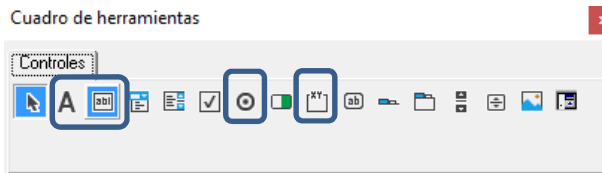
4 Botones de Comandos:

- Primero: Servirá para ver la elección
- Segundo: Restablecer los valores
- Tercero: Mostrar el formulario 2
- Cuarto: Cerrar formularios

Ingrese los objetos en ese orden para que el código se relacione con los objetos ingresados al formulario.



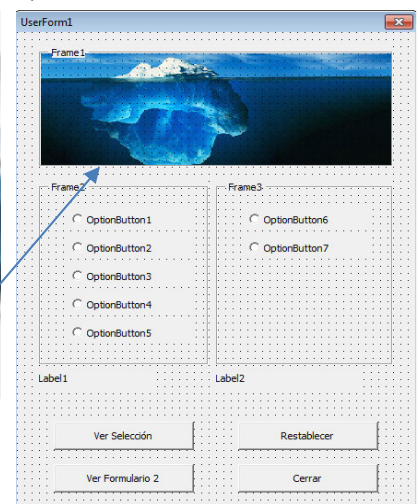
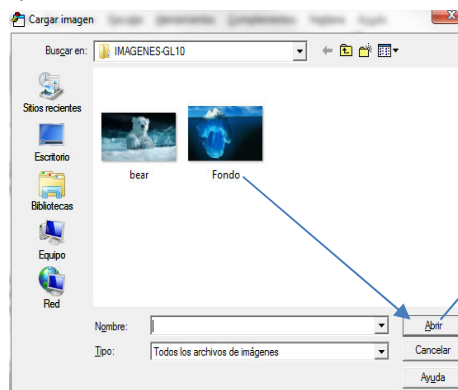
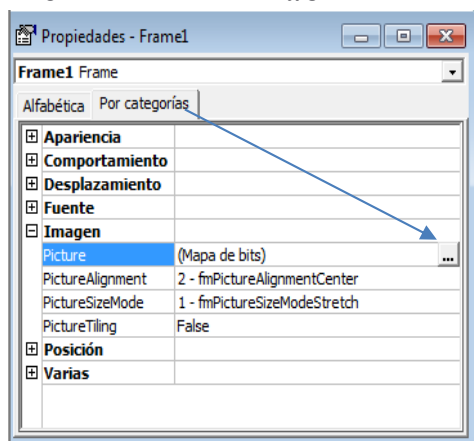
7. Elija del **Cuadro de Herramientas** el Control **Etiqueta** el que tiene la **A** y Arrastre dibujando en el Formulario **USERFORM1** la etiqueta. Quedará el nombre Label1. Si por error da doble clic en la etiqueta y lo manda a la pantalla de programación de la etiqueta, solo de doble clic en **UserForm1** que se encuentra en el **Explorador de Proyecto**. Realice los mismos pasos para el resto de objetos que se mencionan en el paso 5.



8. Las propiedades que cambiaras será únicamente para los siguientes objetos

No	Control	Caption	Name	Propiedad
1	Frame1	Frame1	Frame1	Imagen de fondo y tamaño
2	CommandButton1	Ver Selección	CommandButton1	
3	CommandButton2	Restablecer	CommandButton2	
4	CommandButton3	Ver Formulario 2	CommandButton3	
5	CommandButton4	Cerrar	CommandButton4	

Para cambiar la Propiedad de la Imagen basta con activar la ventana de propiedades (de la vista de Diseño de formularios) y seleccionar el objeto **"Frame1"**. Luego elegir la pestaña **"Por Categorías"**, elegir la categoría **"Imagen"** y de ella seleccionar el botón **"..."** dentro de la propiedad **Picture**. A partir de ahí deberá buscar la carpeta **"IMAGENES-GLXX"** que usted descargo y ubico en la unidad **"C:"**, y seleccionar la imagen con nombre **"Fondo.jpg"**. En cuanto a la propiedad **PictureSizeMode** seleccione en base a lo que se encuentra en la muestra.



9. Digita el código pulsando doble clic al **Botón de Comando "CommandButton1"** y aparecerá la pantalla del Editor de código. Haz lo mismo con el resto de los objetos que acá se mencionan

10. Ahora procedemos a escribir el siguiente código para cada uno de los botones como se muestra a continuación:


LINEAS DE CODIGO:

```
Private Sub CommandButton1_Click()  
    Frame1.Caption = "Frame del grupo 1"  
    Frame2.Caption = "Frame del grupo 2"  
    Frame3.Caption = "Frame del grupo 3"  
    Frame1.Picture = LoadPicture("c:\IMAGENES-GLXX\Fondo.jpg")  
    Frame1.PictureSizeMode = fmPictureSizeModeZoom  
  
    If OptionButton1.Value = True Then  
        Label1.Caption = "Opción 1!!!!"  
        OptionButton1.Caption = "Opción 1!!!!"  
        OptionButton2.Caption = "OptionButton2"  
        OptionButton3.Caption = "OptionButton3"  
        OptionButton4.Caption = "OptionButton4"  
        OptionButton5.Caption = "OptionButton5"  
        OptionButton6.Caption = "OptionButton6"  
        OptionButton7.Caption = "OptionButton7"  
    Else  
        If OptionButton2.Value = True Then  
            Label1.Caption = "Opción 2!!!!"  
            OptionButton2.Caption = "Opción 2!!!!"  
            OptionButton1.Caption = "OptionButton1"  
            OptionButton3.Caption = "OptionButton3"  
            OptionButton4.Caption = "OptionButton4"  
            OptionButton5.Caption = "OptionButton5"  
            OptionButton6.Caption = "OptionButton6"  
            OptionButton7.Caption = "OptionButton7"  
        Else  
            If OptionButton3.Value = True Then  
                OptionButton3.Caption = "Opción 3!!!!"  
                Label1.Caption = "Opción 3!!!!"  
                OptionButton1.Caption = "OptionButton1"  
                OptionButton2.Caption = "OptionButton2"  
                OptionButton4.Caption = "OptionButton4"  
                OptionButton5.Caption = "OptionButton5"  
                OptionButton6.Caption = "OptionButton6"  
                OptionButton7.Caption = "OptionButton7"  
            Else  
                If OptionButton4.Value = True Then  
                    Label1.Caption = "Opción 4!!!!"  
                    OptionButton4.Caption = "Opción 4!!!!"  
                    OptionButton1.Caption = "OptionButton1"  
                    OptionButton2.Caption = "OptionButton2"  
                    OptionButton3.Caption = "OptionButton3"  
                    OptionButton5.Caption = "OptionButton5"  
                    OptionButton6.Caption = "OptionButton6"  
                    OptionButton7.Caption = "OptionButton7"  
                Else  
                    If OptionButton5.Value = True Then  
                        Label1.Caption = "Opción 5!!!!"  
                        OptionButton5.Caption = "Opción 5!!!!"  
                        OptionButton1.Caption = "OptionButton1"  
                        OptionButton2.Caption = "OptionButton2"  
                        OptionButton3.Caption = "OptionButton3"  
                        OptionButton4.Caption = "OptionButton4"  
                        OptionButton6.Caption = "OptionButton6"  
                        OptionButton7.Caption = "OptionButton7"  
                    Else  
                        Label1.Caption = "No ha escogido nada!!!!"  
                    End If  
                End If  
            End If  
        End If  
    End If  
End If
```

```
    If OptionButton6.Value = True Then  
        Label2.Caption = "Opción 6!!!!"  
        OptionButton6.Caption = "Opción 6!!!!"  
        OptionButton2.Caption = "OptionButton2"  
        OptionButton3.Caption = "OptionButton3"  
        OptionButton4.Caption = "OptionButton4"  
        OptionButton5.Caption = "OptionButton5"  
        OptionButton6.Caption = "OptionButton6"  
        OptionButton7.Caption = "OptionButton7"  
    Else  
        If OptionButton7.Value = True Then  
            Label2.Caption = "Opción 7!!!!"  
            OptionButton7.Caption = "Opción 7!!!!"  
            OptionButton2.Caption = "OptionButton2"  
            OptionButton3.Caption = "OptionButton3"  
            OptionButton4.Caption = "OptionButton4"  
            OptionButton5.Caption = "OptionButton5"  
            OptionButton6.Caption = "OptionButton6"  
            OptionButton1.Caption = "OptionButton1"  
        Else  
            Label2.Caption = "No ha escogido nada!!!!"  
        End If  
    End If  
End Sub
```

```
Private Sub CommandButton2_Click()  
    Frame1.Picture = LoadPicture("c:\IMAGENES-GLXX\Fondo.jpg")  
    Frame1.PictureSizeMode = fmPictureSizeModeStretch  
    OptionButton1.Caption = "OptionButton1"  
    OptionButton2.Caption = "OptionButton2"  
    OptionButton3.Caption = "OptionButton3"  
    OptionButton4.Caption = "OptionButton4"  
    OptionButton5.Caption = "OptionButton5"  
    OptionButton6.Caption = "OptionButton6"  
    OptionButton7.Caption = "OptionButton7"  
    Label1.Caption = "Se reestablecio!!!!"  
    Label2.Caption = "Se reestablecio!!!!"  
    OptionButton1.Value = False  
    OptionButton2.Value = False  
    OptionButton3.Value = False  
    OptionButton4.Value = False  
    OptionButton5.Value = False  
    OptionButton6.Value = False  
    OptionButton7.Value = False  
    Frame1.Caption = "Frame1"  
    Frame2.Caption = "Frame2"  
    Frame3.Caption = "Frame3"  
End Sub
```

CommandButton3_Click() y CommandButton4_Click() : deberá de utilizar los comandos vistos en clase para manipular los objetos metodos y eventos de un formulario. Para el caso tendremos que realizar las siguientes acciones:ocultar, mostrar y cerrar el formulario.

11. Probar funcionamiento. Para ello procedemos a presionar el botón Ejecutar **Sub/UserForm** en la barra de herramienta  o presionamos **F5**. Aparecerá el formulario que se muestra en la pagina 9.

12. Crea un Módulo para invocar el formulario. Nombra al módulo como **"FORM1"** que servirá para invocar el formulario actual. En medio de este código deberá de digitar las siguientes 2 líneas:
Load UserForm1
UserForm1.Show
13. Ahora debemos crear un botón de comando desde la hoja para llamar al formulario. Dibujamos el botón según como se explicó en la guía de laboratorio 8 y asigna el módulo creado en el paso 12 de esta guía.
14. Recuerde estar guardando los cambios realizados al Libro **"MacrosC.xlsm"**. Al terminar los pasos, en el libro se debe encontrar la Hoja resultante de la muestra de la página 8, que se encuentra lista para ser utilizado. Antes de guardar asegúrese de haber nombrado la hoja resultante como **"ResultadoF1"**.

Ejemplo 5:

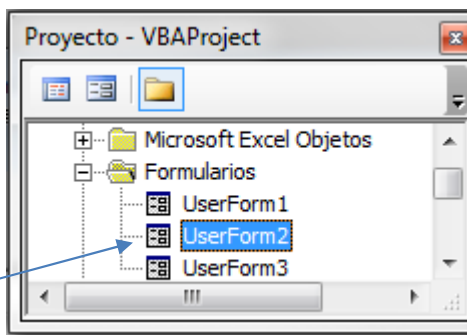
Procederemos a realizar un **Módulo** que permita ejecutar los formularios que nos permitirá conocer más de los objetos de los formularios. Para este caso los resultados se mostrarán en una hoja de Excel denominada **"ResultadoF2"** que se encontrará en nuestro libro de trabajo **"MacrosC.xlsm"** como **"Hoja4"**. Recuerde que este último punto lo aprendió en el laboratorio anterior. Al igual que el anterior formulario usted deberá analizar qué es lo que hace cada Estructura del código al ejecutar el formulario y complementar la documentación del código para aquellas estructuras principales. <<Pregunte a su instructor si lo considera necesario>>

Resultado Esperado:

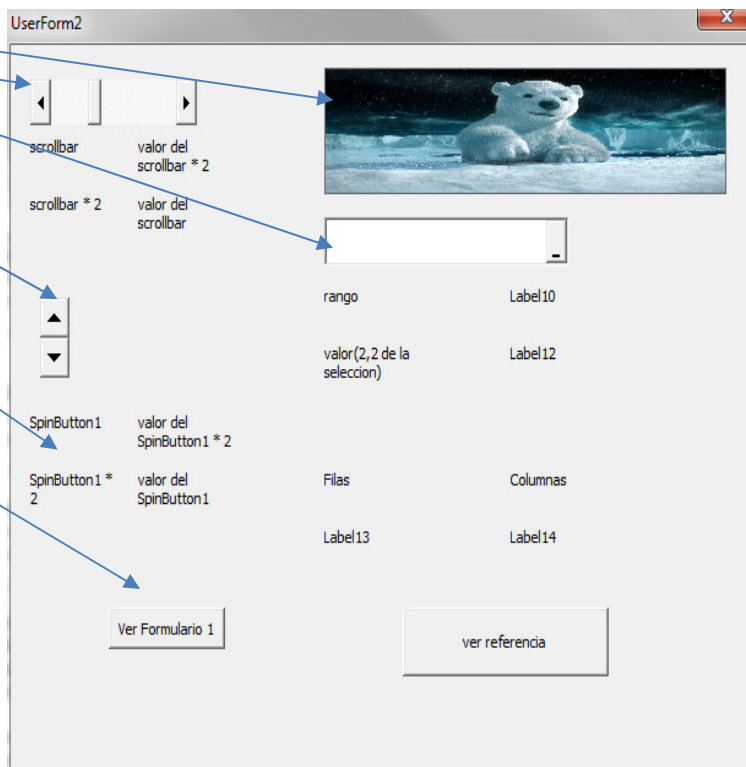
	A	B
1		3 Clic
2	Valor de SpinButton	6
3	Valor de ScrollBar	60

PROCEDIMIENTO

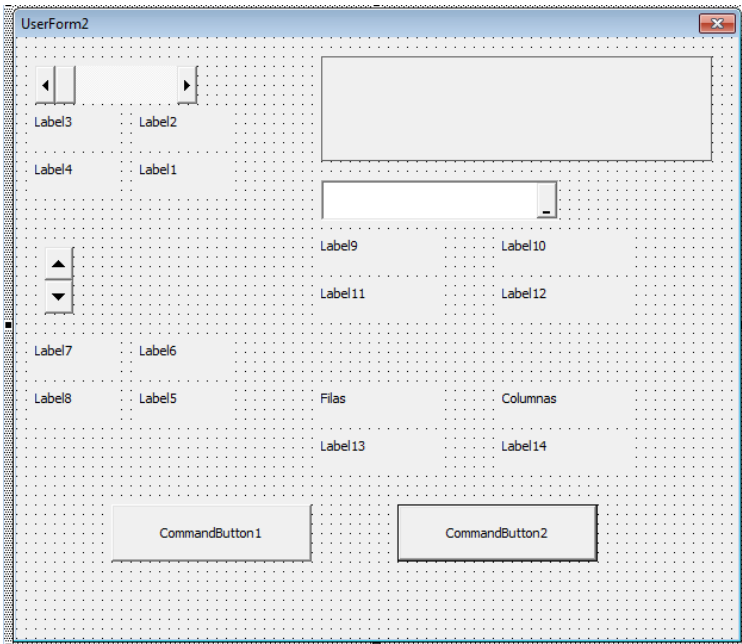
1. Presione La Teclas **Alt + F11**, para entrar al editor de **Visual Basic**.
2. Activa las siguientes opciones:
 - De clic en el **Menú Ver** y elija la opción **Explorador de Proyectos**
 - De clic en el **Menú ver** y elija la opción **Ventana Propiedades**
3. Del **Menú Insertar** elija la Opción **UserForm**. Esto inserta el Formulario que programaremos con controles.
4. En el **Explorador de Proyecto** se observará que se insertó el **UserForm**.
5. Ahora crearas un formulario con el siguiente aspecto:



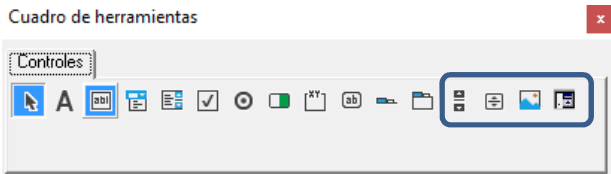
- Una Ventana de Imagen
 - Una Barra de desplazamiento
 - Un Botón de Referencia de celdas
 - Un Botón de Números
- 16 Etiquetas:
- Etiquetas de la 1 a la 4: Para mostrar los valores de la Barra de desplazamiento
 - Etiqueta de la 5 a la 8 : Para mostrar los valores de el botón de Números
 - Etiquetas restantes: Para mostrar los valores del botón de referencia.
- 2 Botones de Comandos:
- Botón de comando 1: Para ver Formulario 1
 - Botón de comando 2: Para ver los valores seleccionados con el botón de referencia.



La muestra que se presenta a continuación es una Vista de Diseño de Formulario. Usted deberá diseñar el formulario Ingresando cada objeto en el orden y ubicación de la muestra. Esto le ayudará a suministrarle funcionamiento a cada objeto del formulario y que se encuentra relacionado con el código que se ingresará en el paso 7.



6. Elija del **Cuadro de Herramientas** cada uno de los Controles y Arrastre hasta dibujar en el Formulario **UserForm2**. Realice los mismos pasos para el resto de objetos que se mencionan en el paso 5.



7. Las propiedades que cambiaras será únicamente para los siguientes objetos

No	Control	Caption	Name	Propiedad
1	Image1		Image1	Imagen de fondo y tamaño
2	ScrollBar1			
3	SpinButton1			
4	CommandButton1	Ver Formulario 1	CommandButton1	
5	CommandButton2	Ver Referencia	CommandButton2	

8. Digita el código pulsando doble clic al **Botón de Comando "CommandButton1"** y aparecerá la pantalla del Editor de código. Haz lo mismo con el resto de los objetos que acá se mencionan.

9. Ahora procedemos a escribir el siguiente código para el formulario y cada uno de los botones como se muestra a continuación:

LINEAS DE CODIGO:

```
Private Sub CommandButton1_Click()
    UserForm2.Hide
    UserForm1.Show
End Sub


Private Sub CommandButton2_Click()
    Dim rango As Range
    Set rango = Range(RefEdit1.Text)
    Label10.Caption = rango.Address
    rango.Select
    Label13.Caption = rango.Rows.Count
    Label14.Caption = rango.Columns.Count
    Label12.Caption = CStr(rango(2, 2))
End Sub

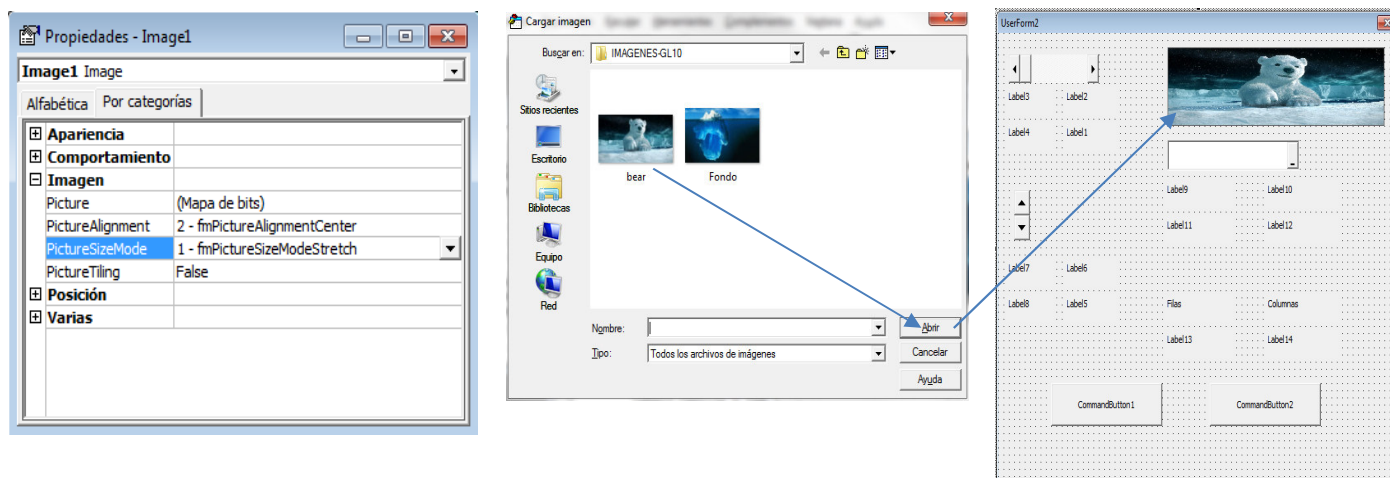
Private Sub ScrollBar1_Change()
    Label1.Caption = ScrollBar1.Value * 2
    Label2.Caption = ScrollBar1.Value
End Sub


Private Sub SpinButton1_Change()
    Label5.Caption = SpinButton1.Value * 2
    Label6.Caption = SpinButton1.Value
End Sub
```

```
Private Sub UserForm_Activate()
    CommandButton1.Caption = "Ver Formulario 1"
    CommandButton1.AutoSize = True
    Label1.Caption = "valor del scrollbar"
    Label2.Caption = "valor del scrollbar * 2"
    Label3.Caption = "scrollbar"
    Label4.Caption = "scrollbar * 2"
    ScrollBar1.Min = -100
    ScrollBar1.Max = 200
    ScrollBar1.SmallChange = 2
    ScrollBar1.LargeChange = 10
    SpinButton1.Min = -500
    SpinButton1.Max = 300
    SpinButton1.SmallChange = 20
    Label5.Caption = "valor del SpinButton1"
    Label6.Caption = "valor del SpinButton1 * 2"
    Label7.Caption = "SpinButton1"
    Label8.Caption = "SpinButton1 * 2"
    Image1.Picture = LoadPicture("c:\IMAGENES-GLXX\Fondo.jpg")
    Image1.PictureSizeMode = fmPictureSizeModeStretch
    Label9.Caption = "rango"
    Label11.Caption = "valor(2,2 de la seleccion)"
    CommandButton2.Caption = "ver referencia"
End Sub
```

Otro método para ubicar la imagen de fondo al objeto **Image1**

Para cambiar la Propiedad de la Imagen basta con activar la ventana de propiedades (de la vista de Diseño de formularios) y seleccionar el objeto "**Image1**". Luego elegir la pestaña **Por Categorías**, elegir la categoría "**Imagen**" y de ella seleccionar el botón  dentro de la propiedad **Picture**. A partir de ahí deberá buscar la carpeta "**IMAGENES-GLXX**" que usted descargó y ubicó en la unidad "**C:**", y seleccionar la imagen con nombre "**bear.jpg**". En cuanto a la propiedad **PictureSizeMode** seleccione en base a lo que se encuentra en la muestra.



10. Probar funcionamiento. Para ello procedemos a presionar el botón Ejecutar **Sub/UserForm** en la barra de herramienta  o presionamos **F5**. Aparecerá el formulario que se muestra en la página 12.
11. Dentro del botón de comando 1 invocará el formulario1 y ocultar el formulario actual. En medio del código del botón de comando deberá de digitar las siguientes 2 líneas:
 UserForm2.Hide
 UserForm1.Show
12. Recuerde estar guardando los cambios realizados al Libro "**MacrosC.xlsm**". Al terminar los pasos, en el libro se debe encontrar la Hoja resultante de la muestra de la página 12, que se encuentra lista para ser utilizado. Antes de guardar asegúrese de haber nombrado la hoja resultante como "**ResultadoF2**".

No olvide subir el libro desarrollado de forma gradual y antes de su laboratorio, esto, previamente a la entrega para calificar de: *MacrosB.xlsm* y *MacrosC.xlsm* al enlace *Entrega de archivos Guía 9* en el curso virtual.

AGREGAR USUARIO Y CONTRASEÑA A NUESTROS DOCUMENTOS CON VBA

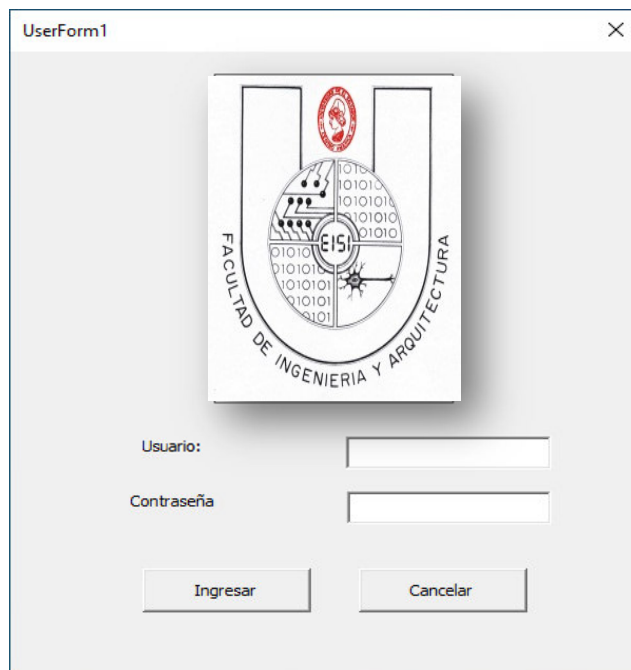
A VBA se le pueda dar una cantidad inimaginable de usos, pero uno muy importantes es el de poder validar la entrada a documentos y hojas.

En este caso realizará el procedimiento para proteger un documento con usuario y contraseña por medio de un formulario básico en VBA para Excel.

Resultado esperado:

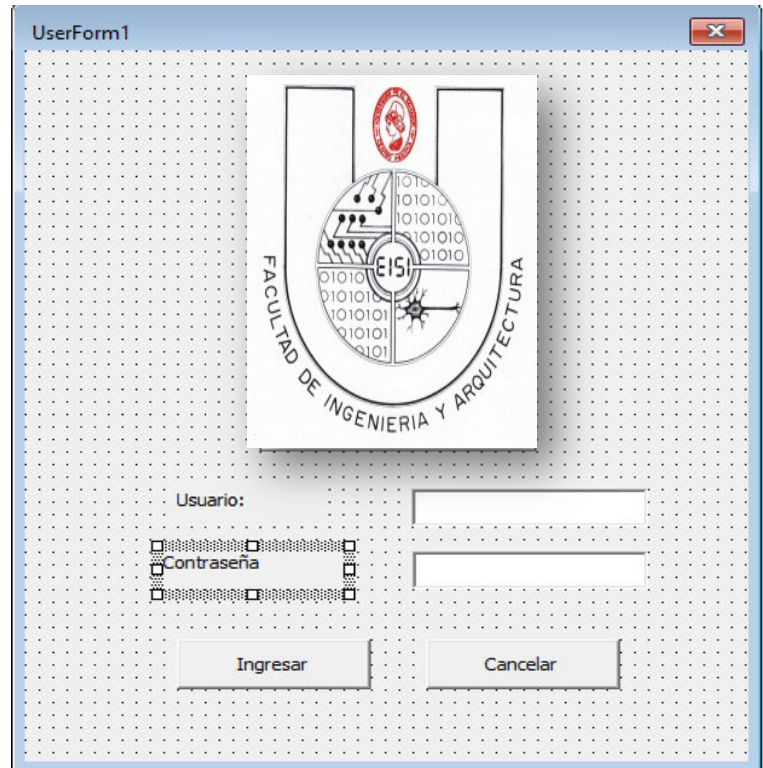
PROCEDIMIENTO

1. Cree una nueva hoja nombrada como "**contra**" en el libro "**MacrosC.xlsm**"



2. Diseñe el siguiente formulario que contiene los siguientes elementos:

- Dos etiquetas, una para **Usuario** y otra para **Contraseña**
- Un cuadro de texto **txtUsuario**
- Un cuadro de texto **txtContra**
- Un botón **btnIngresar**
- Un botón **btnCancelar**



3. Ahora haga doble clic en el botón de comando **Ingresar**, esto le abrirá el editor de texto y por defecto se ubicará en el método para hacer clic del botón de **Ingresar**, a continuación, digite el siguiente código:

```
If Trim(txtUsuario) = "" Then
    MsgBox "Ingrese Usuario, intente de nuevo", vbExclamation, "Alerta"
    txtUsuario.SetFocus
    Exit Sub
End If

If Trim(txtContra) = "" Then
    MsgBox "Ingrese Contraseña", vbExclamation, "Alerta"
    txtContra.SetFocus
    Exit Sub
End If

If Trim(txtUsuario.Text) = "MSM115" And Trim(txtContra.Text) = "Cicloli-2020" Then
    MsgBox ("Acceso Correcto"), vbInformation, "Ha ingresado"
    Application.Visible = True
    End
Else
    MsgBox ("Las credenciales ingresadas son incorrectas, intente de nuevo"), vbExclamation, "Alerta"
End If
```

La primera parte es para definir si el texto de usuario o texto está vacío, si es así, que se deje el curso para escribir en ese cuadro de texto y que lo intente de nuevo el usuario.

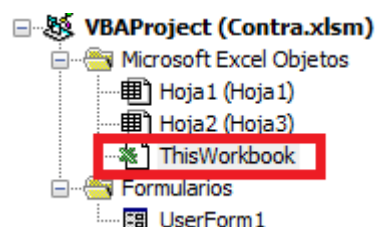
La segunda parte es para validar el usuario y la contraseña, deben de ser **MSM115** y **Cicloli-2020**, respectivamente, si se cumplen estas condiciones se muestra el mensaje de **Acceso Correcto** y se hace visible el documento para el usuario.

4. Ahora regrese al diseño del formulario y haga doble clic en el botón **Cancelar** esto le abrirá el editor de texto y por defecto se ubicará en el método para hacer clic del botón de **Cancelar**, a continuación, digite el siguiente código:

```
Unload Me
```

Esto solo es para cerrar el formulario y el documento cuando se presione este botón.

5. Para ejecutar este código cuando se inicia el documento debe ejecutarlo desde el libro, para eso haga doble clic en **ThisWorkbook** en la parte derecha



6. Digitamos el siguiente código:

```
Private Sub Workbook_Open()  
    Application.Visible = False  
    UserForm1.Show  
End Sub
```

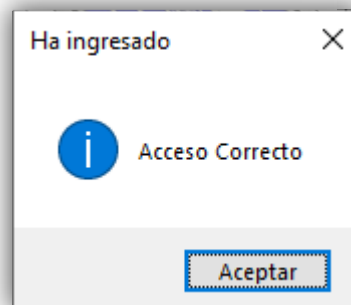
Esto hará que no se muestre el documento hasta que se ejecute el formulario y se ingrese las credenciales correctas

7. Guarde y cierre el documento.

8. Entre de nuevo al documento y pruebe su formulario en vista de ejecución.

The image shows a UserForm1 dialog box. At the top center is the logo of the Facultad de Ingeniería y Arquitectura (EISI). Below the logo, there are two text input fields. The first is labeled 'Usuario:' and the second is labeled 'Contraseña'. At the bottom of the form, there are two buttons: 'Ingresar' (Login) and 'Cancelar' (Cancel).

Al ingresar las credenciales correctas aparecerá el cuadro de mensaje siguiente:



Al presionar **Aceptar** se mostrará el documento por completo

9. Recuerde estar guardando los cambios realizados al Libro "**MacrosC.xlsm**". Al terminar los pasos, en una nueva hoja denominada **Contra** se debe encontrarse la Hoja resultante. Antes de guardar asegúrese de haber nombrado la hoja resultante como "**Contra**".

No olvide subir y entregar para calificar los libros desarrollado máximo 24 horas antes de su revisión: MacrosB.xlsm y MacrosC.xlsm al enlace [Entrega de archivos Guía 9](#) en el curso virtual. De ser necesario suba gradual y conforme a su avance para su entrega oficial.

PROBLEMAS PROPUESTOS

Estos ejercicios consúltelos con su material de clases o con su Instructor, si no los puede Desarrollar. Puede consultarlo dentro de la hora de Reunión de Consulta o enviar consultas antes por mensajería interna o correo electrónico según se pacte con su instructor, prácticas libres y en casa. Desarrolle cada ejercicio en un libro diferente denominada **Propuestos.xlsm** y cuando los finalice, súbalos al enlace: **“Entrega de archivos Guía 9. Ejercicio Propuesto”**, nombre cada archivo según el número de ejercicio.

Ejercicio 1: Haga el ejercicio de conversión de moneda utilizando las sentencias IF THEN ELSE, y presentando en una hoja de Excel los siguientes datos:

1. El monto a convertir y
2. La conversión, especificando el tipo de moneda a la que se convirtió.

Ejercicio 2: Digite los siguientes datos en la hoja de su libro de trabajo.

Madrid	Inserte un módulo en vba y digite el siguiente código
Paris	
Roma	
Londres	
Ciudad de México	Option Explicit
Lima	Function UNA(LISTA As Range)
Bogotá	Dim n As Long
Santiago	Dim ale As Long
Buenos Aires	Randomize
Caracas	n = LISTA.Count
Quito	ale = Int(Rnd * n) + 1
La Paz	UNA = LISTA.Value2(ale, 1)
San José	End Function
San Salvador	Realice los pasos necesarios para ejecutar la función, observe y analice su resultado
Santo Domingo	
Panamá	
Ciudad de Guatemala	

Ejercicio 3: Inserte un nuevo módulo en vba y copie el siguiente código. La función devuelve la suma de dos valores que se le pasan como parámetros desde un procedimiento.

```
Sub profun1()
    ActiveCell.Value = Suma(ActiveSheet.Range("A1").Value,
        ActiveSheet.Range("A2").Value)
End Sub
```

```
Function Suma(V1 As Integer, V2 As Integer) As Integer
    Dim Total As Integer
    Total = V1 + V2
    Suma = Total
End Function
```

Realice los pasos necesarios para ejecutar la función, observe y analice su resultado

Ejercicio 4: Cree la función que se presentan a continuación. La función calcula el factorial de un número entero

```
Function facto(n)
    Dim i As Integer
    facto = 1
    i = 0
    Do
        i = i + 1
        facto = facto * i
    Loop While i <> n
End Function
```

Ejercicio 5: Copie los datos de la Imagen de muestra y calcule la edad insertando un módulo en vba, donde va a digitar el código que se muestra.

Sub nacimiento()

Dim dias As Integer, Dsemana As Integer, Factual As Date, d As String, cumple As Date

Rem Dsemana es una variable que da un número que indica el día de la semana

Rem dado por la función WEEKDAY, que en Excel es =DIASEM(fecha)

Static Fncimient As Date

Factual = Date 'Date es la función de VBA equivalente a =HOY()

Fncimient = Factual

Fncimient = InputBox(Prompt:="Introduzca su fecha de nacimiento", _

Title:="Formato DD-MM-AAAA", Default:=Fncimient)

dias = Factual - Fncimient

Dsemana = Application.WorksheetFunction.Weekday(Fncimient)

Select Case Dsemana

Case 1: d = "Domingo"

Case 2: d = "Lunes"

Case 3: d = "Martes"

Case 4: d = "Miercoles"

Case 5: d = "Jueves"

Case 6: d = "Viernes"

Case 7: d = "Sabado"

End Select

MsgBox Prompt:="Usted nació un " & d & " hace " & dias & " días" & Chr(10) _

& "Tiene " & CalEdad(Fncimient) & " Años", _

Title:="Esta información es correcta siempre que hoy sea " & Factual

End Sub

'Función que calcula la edad en años

Function CalEdad(Fncimient As Date)

Dim Fecha As Date

CalEdad = Abs(DateDiff("yyyy", Fncimient, Date))

Fecha = DateAdd("YYYY", CalEdad, Fncimient)

If Fecha > Date Then CalEdad = CalEdad - 1

End Function

	A	B	C	D	E	F
1						
2	Dia de Nacimiento y Edad					
3						
4						
5		Fecha Actual	07/10/2008			
6						
7	Tratamiento	Nombre	Apellido	Fecha nacimiento	Edad	
8	Sr.	Jose	García	12/06/1969	39 Años	
9	Sra.	Ana	Bronchalo	15/04/1953	55 Años	
10	Sra.	Isabel	Borrillo	29/02/1952	56 Años	
11	Sr.	Juan	Montoya	31/12/1947	60 Años	
12	Sr.	Andrés	Pérez	01/01/1984	24 Años	
13	Sra.	Pino	Sanroman	29/06/1962	46 Años	

Ejercicio 6: Diseñar una Función que permite convertir Kilómetros en Millas y viceversa. Sabiendo que un millón de Millas equivale a 1.609.344 Kilómetros

Ejercicio 7: Diseñar una Función que permita calcular el descuento aplicado a clientes. Un 3% si la venta es >= a un millón y un 1% más es menor.

Ejercicio 8: Cree una hoja en la que existen diferentes botones de opción y cada uno de ellos tiene diferentes valores de descuentos así: a. el domingo es del 5 b. el lunes del 10% c. el resto de semana no hay descuento. Se aplica a una lista de productos, y que de acuerdo a la selección del botón de opción se muestra el descuento en la hoja de cálculo y las respectivas operaciones en ella, como muestra el ejemplo:

Producto	Valor	Cantidad	Total	Descuento	Valor a pagar
Mouse	\$ 25.000,00	10	\$ 250.000,00	\$ 12.500,00	\$ 237.500,00
CPU	\$ 1.200.000,00	25	\$ 30.000.000,00	\$ 1.500.000,00	\$ 28.500.000,00
Monitor	\$ 380.000,00	32	\$ 12.160.000,00	\$ 608.000,00	\$ 11.552.000,00
Teclado	\$ 80.000,00	45	\$ 3.600.000,00	\$ 180.000,00	\$ 3.420.000,00

Descuento actual 5,00%

☒ Descuento Domingo

☐ Descuento Lunes

☐ Descuento resto de semana

Ejercicio 9: Procederemos a realizar un **Módulo** que permita ejecutar los formularios que nos permitirá conocer más de los objetos de los formularios. Usted deberá analizar qué es lo que hace cada Estructura del código al ejecutar el formulario y complementar la documentación del código para aquellas estructuras principales. <<Pregunte a su instructor si lo considera necesario>>

Resultado Esperado:

	A	B	C	D
1	Segundo Valor			Valor de Objeto
2	Tercer Valor		ComboBox1	Segundo Valor
3	Cuarto Valor		ListBox1	El valor 3
4	Quinto Valor		CheckBox1	Desactivado
5	Sexto valor		Opción del Fr	Opción 1!!!!
6	Séptimo Valor		Opción del Fr	Opción 2!!!!

- Ahora crearas un formulario con el siguiente aspecto:

Un ComboBox

Un CheckBox1

Un ListBox1

2 Frame:

- Frame1: Posea 3 Botones de Opciones
- Frame2: Posea 2 Botones de Opciones

4 Etiquetas:

- Etiquetas de la 1 a la 2: Para mostrar los valores del ComboBox.
- Etiqueta de la 3 a la 4 : Para mostrar los valores de los CheckBox
- Etiquetas restantes: Para mostrar los valores del botón de referencia.

2 Botones de Comandos:

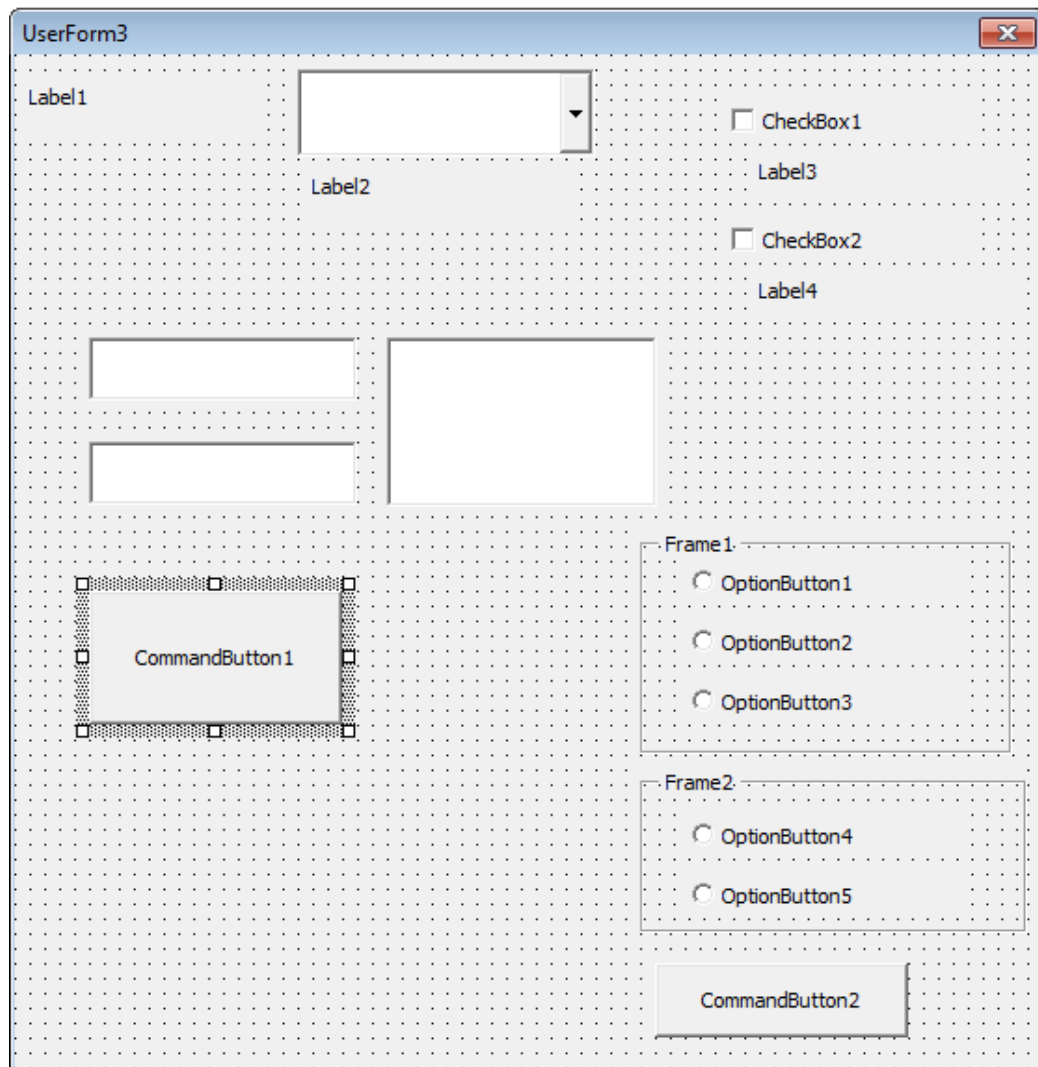
- Botón de comando 1:
 - Para Ver contenido de TextBox (clic) y también de los ListBox.... (clic)
 - Botón de comando 2: Para ver los valores seleccionados con el botón de opciones o Ver OptionButton.....

2 Cuadros de texto: Introducir el valor de los cuadros de texto

The screenshot shows a VBA UserForm titled 'UserForm3'. It contains the following elements:

- A **ComboBox** at the top left with a label 'Texto que presenta, puede ser bastante grande' pointing to it.
- A **CheckBox** on the top right with labels 'Activado' (checked) and 'Desactivado' (unchecked).
- A **ListBox** in the middle left with a label 'Label2' pointing to it.
- Two **Frame** containers:
 - Frame1** contains three **OptionButton** controls labeled 'OptionButton1', 'OptionButton2', and 'OptionButton3'.
 - Frame2** contains two **OptionButton** controls labeled 'OptionButton4' and 'OptionButton5'.
- Two **CommandButton** controls:
 - 'Ver contenido de TextBox y tambien de los ListBox....' with a label pointing to it.
 - 'Ver seleccion de OptionButton....' at the bottom right.
- Various **Label** controls for text display, including 'Label4' near the checkbox.

- Digite en la Hoja los valores de A1 a A6. Según Se Muestra: Segundo Valor, Tercer Valor, Cuarto Valor, Quinto Valor, Sexto valor, Séptimo Valor. Para ello debe de adaptar el código que se le muestra en las páginas siguientes, tomando en cuenta la experiencia que ha adquirido con la programación VBA. Puede además apoyarse en la ayuda de su instructor.
- La muestra que se presenta a continuación es una Vista de Diseño de Formulario. Usted deberá diseñar el formulario Ingresando cada objeto en el orden y ubicación de la muestra. Esto le ayudará a suministrarle funcionamiento a cada objeto del formulario y que se encuentra relacionado con el código que se ingresará en el paso siguiente.



4. Elija del **Cuadro de Herramientas** cada uno de los Controles y Arrastre hasta dibujar en el Formulario **USERFORM3**. Realice los mismos pasos para el resto de objetos que se mencionaron anteriormente.



5. Las propiedades que se cambiaras se hará únicamente dentro del código para cada objeto.
6. Ahora procedemos a escribir el siguiente código para el formulario y cada uno de los botones como se muestra a continuación:

LINEAS DE CODIGO:

- A este código habrá que hacerle algunas mejoras para que los datos aparezcan en la hoja " ResultadoF3 ".
- Se sugiere además que después de haber completado cada objeto, modifique objeto CheckBox para que se pueda elija varias opciones

Private Sub UserForm_Activate()

```

'-----etiqueta
With Label1
    .Caption = "Texto que presenta, puede ser bastante grande....."
    .BackColor = RGB(255, 0, 0)
    .Font.Bold = True
    .Font.Underline = True
    .Font.Name = "Poor Richard"
    .Font.Size = 9
    .AutoSize = True
End With

```

```

'----- Textbox
    TextBox1.SetFocus

'----- Boton de comando
    CommandButton1.Caption = "Ver contenido de TextBox y tambien de los ListBox...."
    CommandButton1.AutoSize = True
    CommandButton2.Caption = "Ver seleccion de OptionButton...."
    CommandButton2.AutoSize = True

'-----ComboBox
    With ComboBox1
        .AddItem "primer valor"
        ThisWorkbook.Sheets("Hoja5").Select
        .AddItem Range("A1").Value
        .AddItem Range("A2").Value
        .AddItem Range("A3").Value
        .AddItem Range("A4").Value
        .AddItem Range("A5").Value
        .AddItem Range("A6").Value
    End With

'-----Listbox
    With ListBox1
        .AddItem "el primer valor"
        For i = 1 To 9
            .AddItem ("El " & CStr(i) & " valor")
        Next
    End With

'-----CheckBox
    CheckBox1.Caption = "Activado"
    CheckBox1.Value = True

    CheckBox2.Caption = "Desactivado"
    CheckBox2.Value = False

'-----Botones de Opcion
    OptionButton1.GroupName = "grupo1"
    OptionButton2.GroupName = "grupo1"
    OptionButton3.GroupName = "grupo1"

    OptionButton4.GroupName = "grupo2"
    OptionButton5.GroupName = "grupo2"
End Sub

Private Sub UserForm_Click()
    Static colorformR As Integer
    Static colorformG As Integer
    Static colorformB As Integer
    colorformR = colorformR + 15
    colorformG = colorformG + 30
    colorformB = colorformB + 45

    If colorformR >= 255 Then
        colorformR = 15
        colorformG = 30
        colorformB = 45
    End If

    UserForm1.BackColor = RGB(colorformR, colorformG, colorformB)

End Sub

```



```
Private Sub UserForm_DblClick(ByVal Cancel As MSForms.ReturnBoolean)
    UserForm1.Hide
End Sub
```

```
Private Sub UserForm_Terminate()
    MsgBox "Cerro la ventana!!!!"
End Sub
```

```
Private Sub CommandButton1_Click() 'para manejo de los TextBox
    Dim vTB1 As String
    Dim vTB2 As String
    vTB1 = TextBox1.Value
    MsgBox "el valor introducido fue: " & vTB1
    vTB2 = TextBox2.Value
    MsgBox "el valor introducido fue: " & vTB2
    TextBox1.Text = ""
    TextBox2.Text = ""
End Sub
```

```
Private Sub ComboBox1_Click() ' Para el manejo del ComboBox
    MsgBox "el valor seleccionado fue: " & ComboBox1.Value
    Label2.Caption = ComboBox1.Value
    Label2.AutoSize = True
    ComboBox1.Value = ""
End Sub
```

```
Private Sub ListBox1_Exit(ByVal Cancel As MSForms.ReturnBoolean)
    ' Para el manejo del ListBox
    MsgBox ListBox1.Value
End Sub
```

```
Private Sub CheckBox1_Change() ' para el manejo de los CheckBox
Select Case CheckBox1.Value
    Case True
        Label3.Caption = "Activado"
        CheckBox1.Caption = "Activado"
        CheckBox2.Value = False
    Case False
        Label3.Caption = "Desactivado"
        CheckBox1.Caption = "Desactivado"
        CheckBox2.Value = True
End Select
```


```
End Sub
Private Sub CheckBox2_Change() ' para el manejo de los CheckBox
Select Case CheckBox2.Value
    Case True
        Label4.Caption = "Activado"
        CheckBox2.Caption = "Activado"
        CheckBox1.Value = False
    Case False
        Label4.Caption = "Desactivado"
        CheckBox2.Caption = "Desactivado"
        CheckBox1.Value = True
End Select
End Sub
```

Private Sub CommandButton2_Click() ' Para el manejo de los Botones de Opcion

```
If OptionButton1.Value = True Then
    MsgBox "eligio la primera opcion del grupo 1"
Else
    If OptionButton2.Value = True Then
        MsgBox "eligio la segunda opcion del grupo 1"
    Else
        MsgBox "eligio la Tercera opcion del grupo 1"
    End If
End If

If OptionButton4.Value = True Then
    MsgBox "eligio la Primera opcion del grupo 2"
Else
    MsgBox "eligio la Segunda opcion del grupo 2"
End If
```

End Sub

-
7. Probar funcionamiento. Para ello procedemos a presionar el botón Ejecutar **Sub/UserForm** en la barra de herramienta  o presionamos **F5**. Aparecerá el formulario que se muestra en la página 18.
 8. Crea un Módulo para invocar el formulario. Nombra al módulo como **"FORM3"** que servirá para invocar el formulario actual. En medio de este código deberá de digitar las siguientes 2 líneas:
Load UserForm3
UserForm3.Show
 9. Ahora debemos crear un botón desde la hoja para llamar al formulario. Dibujamos el botón según como se explicó en la guía de laboratorio 8 y asigna el modulo creado en el paso 8. Recuerde estar guardando con frecuencia los cambios realizados.

Ejercicio 10: Adapte el código del formulario 1 de la guía de laboratorio 8 para que los datos del ingresados al formulario **"Dias Vividos"** se inserten después de haber ingresado el último registro (es decir que se inserte una fila por debajo del registro anterior o último ingresado) y no en la primera ubicación especificada como se hizo en el formulario **"Dias Vividos"**

Ejercicio 11: Adapte el código del formulario 3 de la guía de laboratorio 8 para que el formulario **"Alumnos-msm115"** se le agregue un botón que tenga la funcionalidad de eliminar un registro especificado después de haberlo buscado y encontrado.

Recuerde que es responsabilidad del Estudiante asegurarse que es lo que hace cada línea de comando que se encuentra en cualquiera de las guías de laboratorio de VBA. Todos estos elementos pueden evaluarse a la hora de cada evaluación. Además investigue otras funciones VBA implementadas para la búsqueda, financieras, estadísticas como lo vimos en el ejemplo para buscar carnet en ejercicio **"Alumnos-msm115"**.

Ejercicio 12: Debe diseñar el código necesario para que se presente en la celda A1 un saludo personalizado de forma que si la hora es anterior a las 12 del día se refleje el mensaje "Buenos días Sr X hoy es día de mes del 20xx". Debe solicitar el nombre del usuario por teclado, el día, el mes (con nombre) y el año deberá sacarlo del sistema. Para ello utilice la función de ayuda con de VBA y el examinador de objetos.

Ejercicio 13: crear un procedimiento para sacar todos los factores primos de un número introducido por teclado.

Ejercicio 14: Después de realizar todos los ejercicios propuestos, cree una copia del documento que incluye estos, y en vez de ponerle un formulario de contraseña y usuario al libro, hágalo por hoja, con las siguientes credenciales:

Usuario: admin

Contraseña: 1234

No olvide subir y entregar para calificar los libros desarrollado máximo 24 horas antes de su revisión: *Propuestos.xlsm* al enlace *Entrega de archivos Guía 9. Ejercicio Propuesto* en el curso virtual. De ser necesario suba gradual y conforme a su avance para su entrega oficial.