

## EJEMPLOS INTEGRADORES

### Ejemplo 1 – Macro para generar alta y modificación de datos en una planilla de pacientes de una clínica.

Esta macro permite, a partir de un formulario, realizar el alta o modificación individual de datos en una planilla de pacientes de una clínica.

La planilla está ubicada en la primera hoja (cuyo nombre es "Pacientes") con el siguiente encabezado en el rango A1:E1

ID	DNI	Nombre	Obra Social	Estado
----	-----	--------	-------------	--------

Por otro lado, existen 2 botones asociados a 2 macros: uno para el ALTA y otro para la MODIFICACIÓN DE DATOS.

En la columna "Estado" se indica si el paciente está ACTIVO o INACTIVO (la tabla no permite eliminarlo, solo ponerlo como inactivo modificando los datos).

## FORMULARIO EN VISTA OBJETO

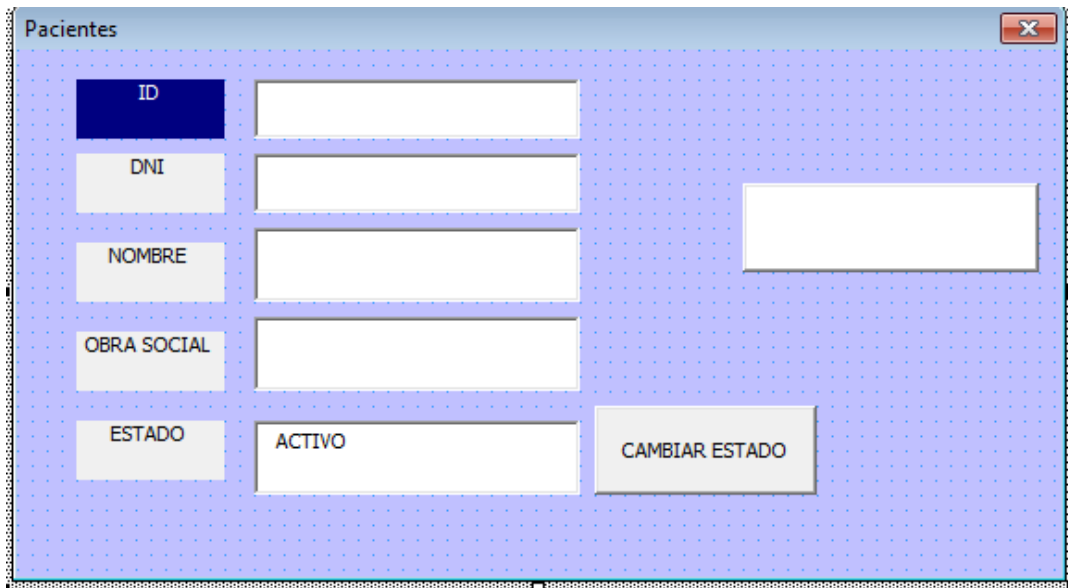
El formulario cuenta con los siguientes controles:

5 **etiquetas** (en inglés, *labels*) → 1 por cada encabezado de la tabla

5 **cuadros de texto** (en inglés, *textbox*) → 1 por cada dato de la tabla

2 **botones** (en inglés, *command button*) → 1 para cambiar el estado ACTIVO/INACTIVO y otro para generar el ALTA y la MODIFICACIÓN de datos.

**Nota:** El botón de ALTA y de MODIFICACIÓN no lleva ningún texto de título (en inglés, la propiedad se llama *caption*) ya que dependiendo la macro que utilicemos le cargará la palabra "ALTA" o "MODIFICACIÓN".



### Propiedades de los controles

Se puede establecer cualquier propiedad en los controles (colores, tamaño, etcétera). Lo relevante es ponerles un NOMBRE (propiedad NAME en inglés) para luego utilizarlos en la programación.

Los **etiquetas** se llaman: LBLID, LBLDNI, LBLNOMBRE, LBLOBRASOCIAL y LBLESTADO

Los **cuadros** de texto se llaman: TXTID, TXTDNI, TXTNOMBRE, TXTOBRASOCIAL y TXTESTADO

Los **botones** se llaman: BT\_ABM (el botón que permite dar el alta o modificación) y BTCAMBIARESTADO (el que permite cambiar de ACTIVO a INACTIVO el cuadro correspondiente).

## FORMULARIO EN VISTA CÓDIGO

En el formulario en vista código, se agregará el siguiente código de programación:

1. Para el botón **BT\_ABM** (Cuando ocurra el evento HACER CLICK EN EL BOTÓN)

```
Private Sub BT_ABM_Click()
```

```
Dim UF As Integer, SF As Integer, FB As Integer
```

```
'Creamos las variables UF = última fila, SF = siguiente fila y FB = fila buscada
```

```
If BT_ABM.Caption = "ALTA" Then
```

```
'CONJUNTO DE COMANDOS QUE SE HARÁN SI EL BOTÓN DICE "ALTA"
```

```
UF = WorksheetFunction.CountA(Range("A:A"))
```

```
'Indica la ÚLTIMA FILA (contando la cantidad de celdas escritas)
```

```
SF = UF + 1
```

```
'Indica la SIGUIENTE FILA
```

```
With ActiveSheet
```

```
'Carga en cada celda de la FILA SIGUIENTE un dato del formulario
```

```
.Cells(SF, 1).Value = TXTID.Value
```

```
.Cells(SF, 2).Value = TXTDNI.Value
```

```
.Cells(SF, 3).Value = UCase(TXTNombre.Value)
```

```
.Cells(SF, 4).Value = UCase(TXTObraSocial.Value)
```

```
.Cells(SF, 5).Value = TXTEstado.Value
```

```
End With
```

```
MsgBox "Paciente N° " & TXTID.Value & " agregado correctamente"
```

```
TXTID.Value = TXTID.Value + 1                                'Nuevo paciente: Prepara el N° siguiente de ID
TXTDNI.Value = ""                                          'Limpia el valor del cuadro de DNI
TXTNombre.Value = ""                                       'Limpia el valor del cuadro de NOMBRE
TXTObraSocial.Value = ""                                    'Limpia el valor del cuadro de OBRA SOCIAL
TXTEstado.Value = "ACTIVO"                                  'Restablece el valor del cuadro de ESTADO a ACTIVO

Elseif BT_ABM.Caption = "MODIFICAR" Then                    'CONJUNTO DE COMANDOS QUE SE HARÁN SI EL BOTÓN DICE "MODIFICACIÓN"

FB = WorksheetFunction.Match(Val(TXTID.Value), Range("A1:A100"), 0)    'Busca el N° de fila en donde se encuentra el ID

Cells(FB, 2).Value = TXTDNI.Value                            'Asigna el valor del cuadro de DNI a la celda de la tabla
Cells(FB, 3).Value = UCase(TXTNombre.Value)                'Asigna el valor del cuadro de NOMBRE a la celda de la tabla
Cells(FB, 4).Value = UCase(TXTObraSocial.Value)            'Asigna el valor del cuadro de OBRA SOCIAL a la celda de la tabla
Cells(FB, 5).Value = TXTEstado.Value                        'Asigna el valor del cuadro de ESTADO a la celda de la tabla

MsgBox "El paciente " & TXTID.Value & " ha sido MODIFICADO exitosamente."

Unload FormPacientes                                       'Cierra el formulario

End If

End Sub
```

2. Para el botón **BTCambiarEstado** (Cuando ocurra el evento HACER CLICK EN EL BOTÓN)

```
Private Sub BTCambiarEstado_Click()  
  
If TXTEstado.Value = "ACTIVO" Then  
  
    TXTEstado.Value = "INACTIVO"  
  
Elseif TXTEstado.Value = "INACTIVO" Then  
  
    TXTEstado.Value = "ACTIVO"  
  
End If  
  
End Sub
```

**Nota:** Son dos macros distintas que funcionan a partir de un evento "HACER CLICK EN UN BOTÓN".

## En el MÓDULO

### MACROS ASOCIADAS A LOS BOTONES

Este es el código asociado a la macro del botón de **ALTA**:

La macro muestra el formulario con las siguientes características:

1. El botón de BT\_ABM pasa a tener el texto **ALTA** para que cumpla esa función cuando se lo presione.
2. Se carga el ID con el siguiente ID que continúe (según el último ID en la tabla).

#### Private IDPaciente As Integer

**Sub PacienteA()**

'Muestra formulario para realizar **ALTA** de paciente

**Dim UF As Integer**

**UF = WorksheetFunction.CountA(Range("A:A"))**

'Calcula la cantidad de filas de la base de datos (última fila usada)

#### With FormPacientes

**.BT\_ABM.Caption = "ALTA"**

'Introduce el texto **ALTA** en el botón

**.BT\_ABM.BackColor = RGB(51, 153, 255)**

**.BT\_ABM.ForeColor = RGB(0, 0, 160)**

**.TXTID.Value = UF**

'Establece el nuevo ID para el nuevo paciente

**.TXTDNI.Enabled = True**

**.TXTNombre.Enabled = True**

**.TXTObraSocial.Enabled = True**

**.TXTEstado.Enabled = True**

**.BTCambiarEstado.Enabled = True**

```
.Show 'Muestra el formulario  
End With  
End Sub
```

Este es el código asociado a la macro del botón de **MODIFICACIÓN**:

La macro primero consulta cual es el número de ID del paciente cuyos datos se quieren modificar.

Luego se valida dicha información introducida:

- Si se introduce un número existente en la tabla, se muestra el formulario.
- Si se introduce un número NO existente en la tabla, se muestra un mensaje de error.
- Si se introduce un valor NO numérico, se muestra un mensaje de error.

En el caso en que se muestra el formulario, se muestra con las siguientes características:

- El botón de BT\_ABM pasa a tener el texto **MODIFICACIÓN** para que cumpla esa función cuando se lo presione.
- Se carga el ID con el siguiente ID escrito en el INPUTBOX.
- Se rellenan todos los cuadros de texto del formulario con el dato correspondiente a ese paciente (con un BUSCARV se busca la información en la tabla)

```
Sub PacienteM() 'Muestra formulario para realizar MODIFICACIÓN de paciente  
ChequeoNumero = InputBox("Escribir el ID del paciente:", "Modificación Paciente", "") 'Permite al usuario escribir Número de ID  
  
'Maneja error: Controla que SE ESCRIBA UN NÚMERO en el INPUTBOX y que EXISTA en la tabla (con la función CONTAR.SI)  
If IsNumeric(ChequeoNumero) And WorksheetFunction.CountIf(Range("A:A"), ChequeoNumero) = 1 Then  
IDPaciente = ChequeoNumero 'Asigna a la variable IDPaciente el valor numérico escrito por el usuario  
With FormPacientes
```

```
.BT_ABM.Caption = "MODIFICAR"           'Introduce el texto MODIFICACIÓN en el botón
.BT_ABM.BackColor = RGB(155, 205, 100)
.BT_ABM.ForeColor = RGB(0, 70, 50)
.TXTID.Value = IDPaciente
.TXTDNI.Value = WorksheetFunction.VLookup(IDPaciente, Range("A:G"), 2, 0)   'Carga la información en el formulario (función BUSCARV)
.TXTNombre.Value = WorksheetFunction.VLookup(IDPaciente, Range("A:G"), 3, 0)
.TXTObraSocial.Value = WorksheetFunction.VLookup(IDPaciente, Range("A:G"), 4, 0)
.TXTEstado.Value = WorksheetFunction.VLookup(IDPaciente, Range("A:G"), 5, 0)
.TXTDNI.Enabled = True
.TXTNombre.Enabled = True
.TXTObraSocial.Enabled = True
.TXTEstado.Enabled = True
.BTCambiarEstado.Enabled = True
.Show           'Muestra el formulario
End With
'Maneja error: Controla que SE ESCRIBA UN NÚMERO en el INPUTBOX aunque no exista en la tabla (con la función CONTAR.SI)
Elseif IsNumeric(ChequeoNumero) And WorksheetFunction.CountIf(Range("A:A"), ChequeoNumero) = 0 Then
  MsgBox "El paciente seleccionado no figura en la base de datos.", vbOKOnly, "Paciente inexistente"

'Maneja error: Controla que NO SE ESCRIBA UN NÚMERO en el INPUTBOX
Elseif Not IsNumeric(ChequeoNumero) Then
  MsgBox "Debe escribir un ID válido (sólo números).", vbOKOnly, "Error"
End If

End Sub
```