

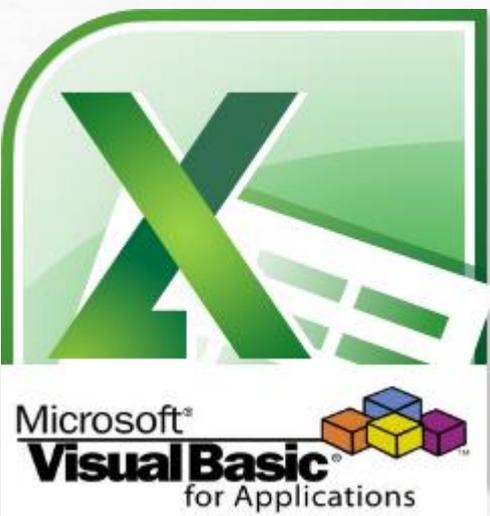


# **MICROSOFT EXCEL**

## **HERRAMIENTAS INFORMÁTICAS II**



**Prof. Ing. Norma Cañizares**



**Microsoft Excel**

**Visual Basic for Applications**



# Visual Basic for Applications

- **Direccionamiento con la Propiedad End.**
- **Búsqueda de un Registro mediante Código.**
- **Ejemplos: Operaciones sobre Registros.**

**Microsoft Excel**

# SELECCIONAR UNA CELDA O UN RANGO

- Para seleccionar una celda o un rango utilice el método SELECT del objeto RANGE.

## Ejemplos:

Range("b2","b10").Select *seleccionar un rango.*

Range("b2:b10").Select *seleccionar un rango.*

Range("b2").Select *seleccionar una celda.*

### Range.Método Select

Selecciona el objeto.

#### Sintaxis

*expresión*.Select|

*Expresión* Variable que representa un objeto **Range**.

#### Valor devuelto

Variant

#### Comentarios

Para seleccionar una celda o un rango de celdas, use el método **Select**. Para activar una sola celda, use el método **Activate**.

- También, puede utilizar el Método Activate para activar una sola celda.

## Ejemplo:

Range("b2").Activate

# COMO DESPLAZARNOS A TRAVÉS DE LA HOJA DE CALCULO MEDIANTE CÓDIGO

- Direccionamiento con xlDown, xlUp, xlToRight, xlToLeft
- Usamos la propiedad END, del objeto Range, el cual tiene unos valores constantes que nos permiten desplazarnos a través de la hoja de calculo. Estas propiedades son :

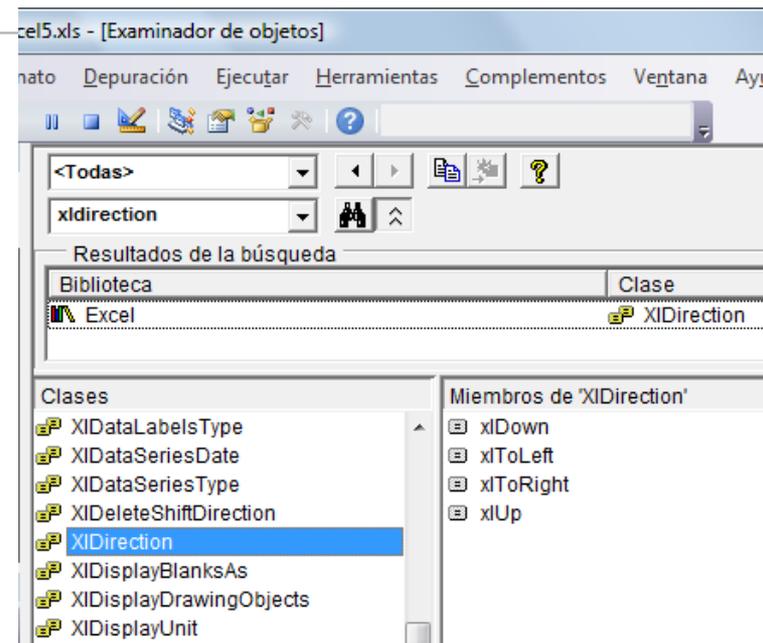
Instrucción	Teclas Combinadas	
End(xlDown)	Fin + Flecha Abajo	Ctrl + Flecha Abajo
End(xlUp)	Fin + Flecha Arriba	Ctrl + Flecha Arriba
End(xlToRight)	Fin + Flecha Derecha	Ctrl + Flecha Derecha
End(xlToLeft)	Fin + Flecha Izquierda	Ctrl + Flecha Izquierda

**Sintaxis:**

***Expresión.End(Dirección)***

*Expresión* : Valor que representa un objeto range

*Dirección* : Direccion de Desplazamiento (xlDirection)



# DESPLAZARSE HASTA LA ULTIMA CELDA CON DATO Y SELECCIONARLA

```
Option Explicit  
  
Sub direccionamiento()  
Range("A2").End(xlDown).Select  
End Sub
```

Toma de referencia la celda A2

Direcciona a la ultima celda con dato

Esta acción, devuelve la ultima celda con datos, que para este caso es el objeto Celda A6.

Una vez, que se obtiene la ultima celda con dato, objeto Celda A6, se lo selecciona.

	A	B	C
	Código de Producto	Nombre de Producto	Precio Unit.
1			
2	1	Lápiz	\$ 22,00
3	2	Goma	\$ 15,00
4	3	Cuaderno	\$ 150,00
5	4	Regla	\$ 35,00
6	5	Plasticola	\$ 55,00
7			
8			
9			
10			
11			

# SELECCIONAR UN RANGO

- **Seleccionar un Rango Constante**

```
Option Explicit

Sub SeleccionarRango()
'seleccionar un rango constante

Range("A2:A6").Select

Range("A2", "A6").Select

End Sub
```

Estas 2 líneas de código, producen el mismo efecto, seleccionan el rango "A2:A6"

	A	B	C
1	Código de Producto	Nombre de Producto	Precio Unit.
2	1	Lápiz	\$ 22,00
3	2	Goma	\$ 15,00
4	3	Cuaderno	\$ 150,00
5	4	Regla	\$ 35,00
6	5	Plasticola	\$ 55,00
7			
8			
9			
10			
11			

- **Seleccionar un Rango Variable (en tiempo de ejecución)**

```
Option Explicit

Sub SeleccionarRango()
'seleccionar un rango variable en tiempo de ejecucion

Range("A2", Range("A2").End(xlDown)).Select

End Sub
```

**Range("A2","A6").Select**

Con esta línea de código, SIEMPRE se selecciona el rango correcto, independientemente si se añade o se eliminan registros de la Tabla de datos.

# OBTENER EL NRO DE FILA DE UNA CELDA

Para obtener el numero de fila de una celda seleccionada utilizamos la **Propiedad Row** del **Objeto Range**.

## **Range.Row**

- Devuelve el numero de la primera fila de la primer área del rango (**Long** de solo lectura).

### Sintaxis

#### *Expresión.Row*

**Expresión:** Variable que representa un objeto Range.

### Ejemplos:

```
Msgbox(Range("A1").Row)
```

```
Nrofila= Range("A1").Row
```

```
Nrofila= Range("A1").End(xldown).Row
```

```
Nrofila = Activecell.Row
```

# DESPLAZARSE HASTA LA ULTIMA CELDA CON DATO Y OBTENER SU NRO DE FILA.

```
Option Explicit

Sub direccionamiento()

Dim UltimaFila As Long

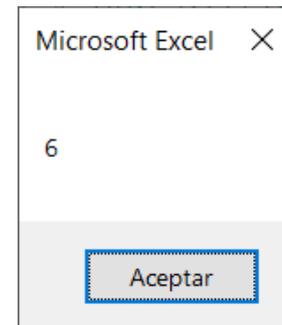
UltimaFila = Range("A2").End(xlDown).Row
MsgBox UltimaFila

End Sub
```

Esta acción, devuelve la ultima celda con datos, que para este caso es el objeto Celda A6.

Una vez, que se obtiene la ultima celda con dato, objeto Celda A6, se devuelve el numero de fila y se lo muestra mediante un mensaje.

	A	B	C
	Código de Producto	Nombre de Producto	Precio Unit.
1			
2	1	Lápiz	\$ 22,00
3	2	Goma	\$ 15,00
4	3	Cuaderno	\$ 150,00
5	4	Regla	\$ 35,00
6	5	Plasticola	\$ 55,00
7			
8			
9			
10			
11			



*Otra forma de obtener el nro de la ultima fila con datos, es utilizar la dirección "xlUp" de la propiedad End del objeto Range, como en el siguiente ejemplo:*

```
UltimaFila = Range("A100000").End(xlUp).Row
```

# AÑADIR UN REGISTRO CON DATOS AL FINAL DE LA TABLA

En este ejemplo, suponemos que la validación de datos, de los parámetros del artículo a añadir, se encuentra en el Evento Change de Worksheets.

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00

Antes de ejecutar el código

La variable **UltimaFila** obtiene el Nro de fila del ultimo registro con datos, que en este caso es la **Fila numero 15.**

```
Private Sub CMB_Alta_Click()  
Dim UltimaFila As Long  
Dim sTitulo As String, sMensaje As String, sBotones As String  
Dim Respuesta As Integer  
  
sTitulo = "Confirmación"  
sMensaje = "Desea agregar el registro actual? "  
sBotones = vbYesNo + vbInformation  
  
If IsEmpty(Range("D4")) Or IsEmpty(Range("D5")) Or IsEmpty(Range("D6")) Then  
MsgBox ("Debe completar los 3 parametros para poder dar de Alta el Artículo")  
Else  
Respuesta = MsgBox(Title:=sTitulo, Prompt:=sMensaje, Buttons:=sBotones)  
If Respuesta = vbYes Then  
UltimaFila = Range("C100000").End(xlUp).Row  
Cells(UltimaFila + 1, 3) = Range("D4")  
Cells(UltimaFila + 1, 4) = Range("D5")  
Cells(UltimaFila + 1, 5) = Range("D6")  
Cells(UltimaFila + 1, 5).NumberFormat = "$ #,##0.00"  
Range("D4") = ""  
Range("D5") = ""  
Range("D6") = ""  
End If  
End If  
End Sub
```

`Cells(UltimaFila + 1, 3) = Range("D4")`

$$15 + 1 = 11$$

`Cells (16, 3) = Range("D4")`

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

Después de ejecutar el código

# EVITAR DUPLICIDAD DE REGISTROS DURANTE EL ALTA DE UN REGISTRO

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

Código de Producto	Nombre de Producto	Precio Unit.
5	Semicirculo	\$ 46,00

**Añadir Registro**

Con este código, ¿Que sucedería, si el usuario añade el siguiente registro a la tabla de productos?

En que casos, se debería permitir el alta de un nuevo producto a la tabla de datos?

```
Private Sub CMB_Alta_Click()  
Dim UltimaFila As Long  
Dim sTitulo As String, sMensaje As String, sBotones As String  
Dim Respuesta As Integer  
  
sTitulo = "Confirmación"  
sMensaje = "Desea agregar el registro actual? "  
sBotones = vbYesNo + vbInformation  
  
If IsEmpty(Range("D4")) Or IsEmpty(Range("D5")) Or IsEmpty(Range("D6")) Then  
    MsgBox ("Debe completar los 3 parametros para poder dar de Alta el Artículo")  
Else  
    Respuesta = MsgBox(Title:=sTitulo, Prompt:=sMensaje, Buttons:=sBotones)  
    If Respuesta = vbYes Then  
        UltimaFila = Range("C100000").End(xlUp).Row  
        Cells(UltimaFila + 1, 3) = Range("D4")  
        Cells(UltimaFila + 1, 4) = Range("D5")  
        Cells(UltimaFila + 1, 5) = Range("D6")  
        Cells(UltimaFila + 1, 5).NumberFormat = "$ #,##0.00"  
        Range("D4") = ""  
        Range("D5") = ""  
        Range("D6") = ""  
    End If  
End If  
End Sub
```



# Visual Basic for Applications

- **Direccionamiento con la Propiedad End.**
- **Búsqueda de un Registro mediante Código.**
- **Ejemplos: Operaciones sobre Registros.**

**Microsoft Excel**

# BUSCAR UN REGISTRO MEDIANTE CÓDIGO CON EL MÉTODO FIND DEL OBJETO RANGE

## Sintaxis

**[Rango de búsqueda].Find(What:=Valor\_a\_buscar, LookAt:=xlWhole).Activate**

- **Valor devuelto:** objeto Range que representa la primera celda en la que se encuentra información.
- **LookAt:** opcional. De tipo variante, indica si el valor encontrado debe ser rigurosamente igual al valor buscado o en parte solamente. Ejemplo, en la matriz: 6210, 4105, 540, 163, 154, 132, 10, buscas el valor 10. Para indicar a VBA que buscas exclusivamente 10, utilizarás **LookAt: = xlWhole**. En cambio, para indicarle que 6210 o 4105 te conviene (ambas contienen 10), utilizarás: **LookAt: = xlPart**.

Ej. de Rango de Búsqueda:

- Cells (toda la hoja)
  - [B:B] (toda la columna B)
  - [3:3] (toda la fila 3)
  - Range("A2:B99") (un rango definido).
- Cells.Find(What:=valor\_a\_buscar).Activate

**Si encuentra el valor devuelve un Objeto Range, de lo contrario devuelve Error.**

# EJEMPLO: CONSULTAR UN REGISTRO (S/RUTINA DE TRATAMIENTO DE ERRORES)

	A	B	C	D	E	F	G	
1								
2			<b>Consulta de Artículos de Librería</b>					
3								
4			<b>Código de Producto</b>	<input type="text" value="2"/>				
5			<b>Nombre de Producto</b>	Goma		<b>Consultar Registro</b>		
6			<b>Precio Unit.</b>	\$ 15,00				
7								
8								
9								
10								
11			<b>Código de Producto</b>	<b>Nombre de Producto</b>	<b>Precio Unit.</b>			
12			1	Lápiz	\$ 22,00			
13			2	Goma	\$ 15,00			
14			3	Cuaderno	\$ 150,00			
15			4	Regla	\$ 35,00			
16			5	Plasticola	\$ 55,00			
17			6	Escuadra	\$ 45,00			
18								
19								
20								
21								
22								

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

Para el ejemplo, se consulta el código de producto 2, y al ejecutar el código, como lo encuentra, se devuelven los datos asociados al producto.

## Option Explicit

```
Private Sub BTN_Consulta_Click()  
Dim ValorBuscar As Integer  
Dim FilaEncontrado As Long  
  
If IsEmpty(Range("D4")) Then  
    MsgBox ("Debe ingresar un Cod. de Producto p/consultar")  
Else  
    ValorBuscar = Range("D4")  
    FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row  
    Range("D5") = Cells(FilaEncontrado, 4)  
    Range("D6") = Cells(FilaEncontrado, 5)  
End If  
  
End Sub
```

# EJEMPLO: CONSULTAR UN REGISTRO INEXISTENTE

## (S/RUTINA DE TRATAMIENTO DE ERRORES)

Consulta de Artículos de Librería		
Código de Producto	10	Consultar Registro
Nombre de Producto		
Precio Unit.		

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

Que pasa con la ejecución del código, si el producto buscado no se encuentra???

Microsoft Visual Basic

Se ha producido el error '91' en tiempo de ejecución:

Variable de objeto o bloque With no establecido

Continuar

Finalizar

Depurar

Ayuda

Option Explicit

```
Private Sub BTN_Consulta_Click()  
Dim ValorBuscar As Integer  
Dim FilaEncontrado As Long  
  
If IsEmpty(Range("D4")) Then  
    MsgBox ("Debe ingresar un Cod. de Producto p/consultar")  
Else  
    ValorBuscar = Range("D4")  
    FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row  
    Range("D5") = Cells(FilaEncontrado, 4)  
    Range("D6") = Cells(FilaEncontrado, 5)  
End If  
  
End Sub
```

# EJEMPLO: CONSULTAR UN REGISTRO C/RUTINA DE TRATAMIENTO DE ERRORES

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

En este ejemplo, se implementa una Rutina de Tratamiento de Errores, para controlar aquellas situaciones en las que el código consultado no existe, y en consecuencia impedir que la ejecución del código termine abruptamente.

Option Explicit

```
Private Sub BTN_Consulta_Click()  
Dim ValorBuscar As Integer  
Dim FilaEncontrado As Long  
  
On Error GoTo TratarErrores  
    If IsEmpty(Range("D4")) Then  
        MsgBox ("Debe ingresar un Cod. de Producto p/consultar")  
    Else  
        ValorBuscar = Range("D4")  
        FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row  
        Range("D5") = Cells(FilaEncontrado, 4)  
        Range("D6") = Cells(FilaEncontrado, 5)  
    End If  
Exit Sub  
  
TratarErrores:  
    MsgBox ("El Código de Producto Consultado NO EXISTE")  
    Range("D4").Select  
  
End Sub
```

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

# EJEMPLO: AÑADIR UN REGISTRO C/RUTINA DE TRATAMIENTO DE ERRORES

Para el registro que se intenta añadir, este código emitiría un mensaje de duplicidad de producto.

Alta de Artículos de Librería		
Código de Producto	5	Añadir Registro
Nombre de Producto	Semicirculo	
Precio Unit.	\$ 46,00	

Option Explicit

```
Private Sub CMB_Alta_Click()
Dim UltimaFila As Long
Dim FilaEncontrado As Long
Dim ValorBuscar As Integer
Dim sTitulo As String, sMensaje As String, sBotones As String
Dim Respuesta As Integer

On Error GoTo TratarErrores
sTitulo = "Confirmación"
sMensaje = "Desea agregar el registro actual? "
sBotones = vbYesNo + vbInformation

If IsEmpty(Range("D4")) Or IsEmpty(Range("D5")) Or IsEmpty(Range("D6")) Then
MsgBox ("Debe completar los 3 parametros para poder dar de Alta el Artículo")
Else
ValorBuscar = Range("D4")
FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row
MsgBox ("El Código de Producto Ingresado YA EXISTE")
Range("D4") = Empty
End If
Exit Sub
TratarErrores:
Respuesta = MsgBox(Title:=sTitulo, Prompt:=sMensaje, Buttons:=sBotones)
If Respuesta = vbYes Then
UltimaFila = Range("C11").End(xlDown).Row
Cells(UltimaFila + 1, 3) = Range("D4")
Cells(UltimaFila + 1, 4) = Range("D5")
Cells(UltimaFila + 1, 5) = Range("D6")
Cells(UltimaFila + 1, 5).NumberFormat = "$ #,##0.00"
Range("D4") = ""
Range("D5") = ""
Range("D6") = ""
End If
End Sub
```

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

En este ejemplo, el alta del registro solo se efectúa, para aquellos casos en donde el código de producto a añadir no existe en la tabla de datos. Por lo tanto, el alta se produce en el conjunto de instrucciones para el Tratamiento de Errores.

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

# BUSCAR UN REGISTRO MEDIANTE CÓDIGO CON LA FUNCIÓN MATCH

**WorksheetFunction.Match**, devuelve la posición relativa de un elemento en una matriz que coincida con un valor especificado en el orden indicado.

## Sintaxis

**WorksheetFunction.Match(Valor\_a\_buscar, Rango de búsqueda, Tipo\_de\_coincidencia)**

- Si el argumento de tipo\_de\_coincidencia es 0, MATCH busca el primer valor que sea exactamente igual al argumento de valor\_buscado. El argumento de matriz\_buscada puede estar en cualquier orden.
- **Valor devuelto:** La función MATCH devuelve un valor numérico.
- **Nota:**

La función MATCH no distingue entre mayúsculas y minúsculas cuando busca una coincidencia.

Si el parámetro **Tipo\_de\_coincidencia** es 0 y el **Valor\_a\_buscar** es un valor de texto, puede usar comodines en este parámetro.

Ej.

**Posicion=WorksheetFunction.Match(55, WorkSheets("Tabla Productos").Range("A:A"), 0)**

**Si encuentra el valor devuelve la posición relativa del elemento, de lo contrario devuelve Error.**

# Eliminar un Registro mediante Código

## Worksheet.Rows (propiedad)

- Devuelve un objeto Range que representa todas las filas de la hoja de cálculo especificada. Objeto Range de sólo lectura.

**Sintaxis:**                    **Expresión.Rows**

*Expresión* : Variable que representa un objeto Worksheet.

**Comentarios:** El uso de esta propiedad sin un calificador de objeto equivale a usar ActiveSheet.Rows.

## Método Delete

- Elimina el objeto especificado.

**Sintaxis:**                    **Expresión.Delete**

*Expresión:* Variable que representa un objeto Range.

## **Ejemplo**

```
Worksheets("Sheet1").Rows(3).Delete
```

*En este ejemplo se elimina la fila tres de Sheet1.*

# Ejemplo: Eliminar un Registro

	A	B	C	D
1		Ejemplo: Eliminar una Fila		
2				
3		Codigo a Eliminar:	5	
4				
5		Codigo	Nombre	
6		1	Nombre 1	
7		2	Nombre 2	
8		3	Nombre 3	
9		4	Nombre 4	
10		6	Nombre 6	
11		7	Nombre 7	
12		8	Nombre 8	
13		9	Nombre 9	
14		10	Nombre 10	
15				

(General)

Eliminar

Option Explicit

```
Sub Eliminar()
```

```
Dim Codigo As Integer
```

```
Dim FilaEncontrado As Integer
```

```
Codigo = Range("C3").Value
```

```
FilaEncontrado = Range("B6", Range("B6").End(xlDown)).Find(WHAT:=Codigo, LookAt:=xlWhole).Row
```

```
ActiveSheet.Rows(FilaEncontrado).Delete
```

```
End Sub
```



# Visual Basic for Applications

- **Ejemplos: Operaciones sobre Registros**
  - **Agregar un Registro.**
  - **Consultar un Registro.**
  - **Eliminar un Registro.**
  - **Modificar un Registro.**
  - **Listar Registros.**

**Microsoft Excel**

# EJEMPLO: AGREGAR UN REGISTRO

Option Explicit

```
Private Sub CMB_Alta_Click()
Dim UltimaFila As Long
Dim FilaEncontrado As Long
Dim ValorBuscar As Integer
Dim sTitulo As String, sMensaje As String, sBotones As String
Dim Respuesta As Integer
```

```
On Error GoTo TratarErrores
sTitulo = "Confirmación"
sMensaje = "Desea agregar el registro actual? "
sBotones = vbYesNo + vbInformation
```

```
If IsEmpty(Range("D4")) Or IsEmpty(Range("D5")) Or IsEmpty(Range("D6")) Then
MsgBox ("Debe completar los 3 parametros para poder dar de Alta el Artículo")
```

```
Else
ValorBuscar = Range("D4")
FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row
MsgBox ("El Código de Producto Ingresado YA EXISTE")
Range("D4") = Empty
End If
```

```
Exit Sub
```

```
TratarErrores:
```

```
Respuesta = MsgBox(Title:=sTitulo, Prompt:=sMensaje, Buttons:=sBotones)
```

```
If Respuesta = vbYes Then
```

```
UltimaFila = Range("C11").End(xlDown).Row
Cells(UltimaFila + 1, 3) = Range("D4")
Cells(UltimaFila + 1, 4) = Range("D5")
Cells(UltimaFila + 1, 5) = Range("D6")
Cells(UltimaFila + 1, 5).NumberFormat = "$ #,##0.00"
Range("D4") = ""
Range("D5") = ""
Range("D6") = ""
```

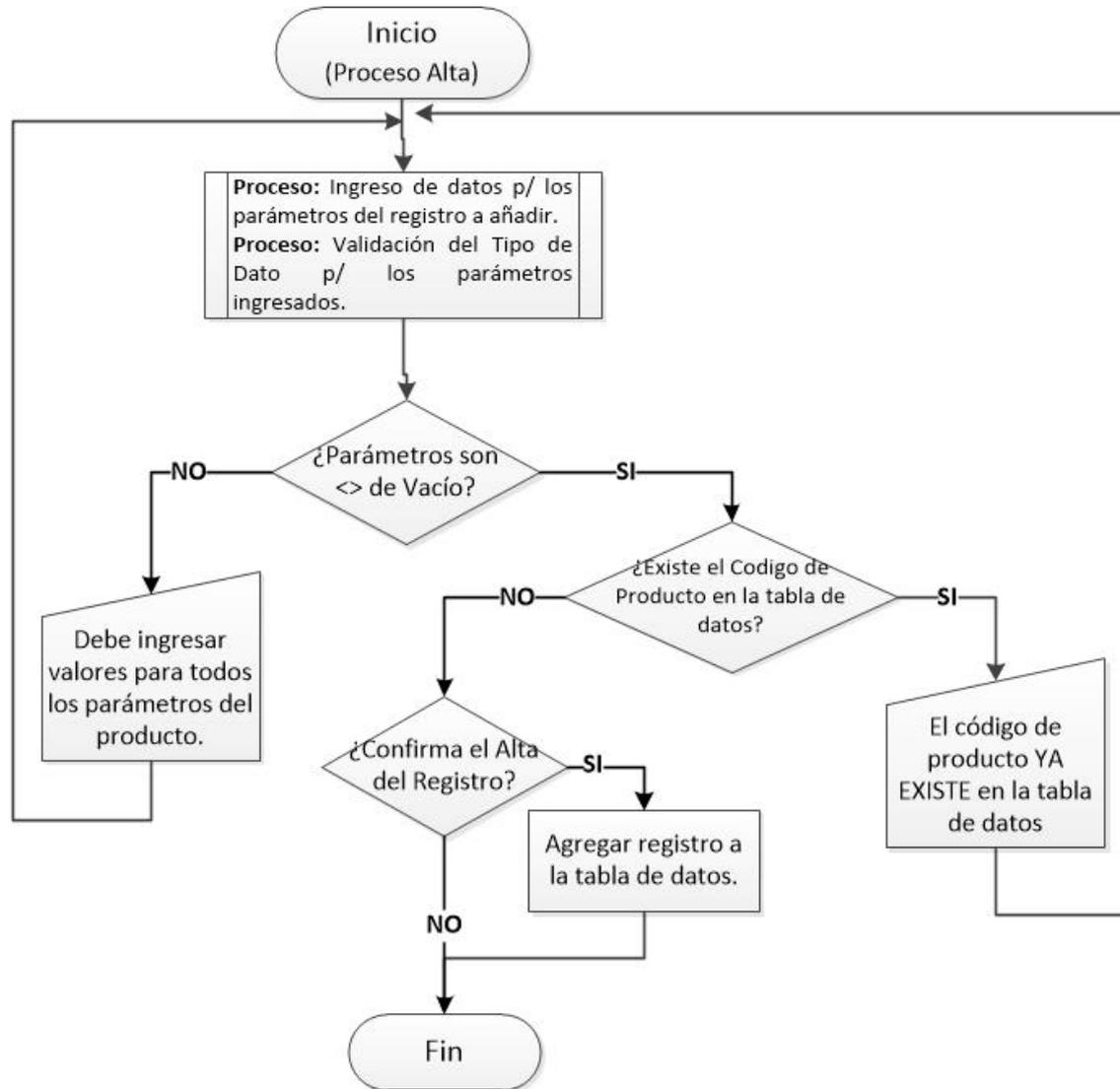
```
End If
```

```
End Sub
```

	A	B	C	D	E	F	G
1							
2			<b>Alta de Artículos de Librería</b>				
3							
4			<b>Código de Producto</b>	5		<b>Añadir Registro</b>	
5			<b>Nombre de Producto</b>	Semicirculo			
6			<b>Precio Unit.</b>	\$ 46,00			
7							
8							
9							
10			<b>Código de Producto</b>	<b>Nombre de Producto</b>	<b>Precio Unit.</b>		
11			1	Lápiz	\$ 22,00		
12			2	Goma	\$ 15,00		
13			3	Cuaderno	\$ 150,00		
14			4	Regla	\$ 35,00		
15			5	Plasticola	\$ 55,00		
16			6	Escuadra	\$ 45,00		
17							
18							
19							

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

# DIAGRAMA DE FLUJO: AGREGAR UN REGISTRO



# Ejemplo: Consultar un Registro

	A	B	C	D	E	F	G	
1								
2			<b>Consulta de Artículos de Librería</b>					
3								
4			<b>Código de Producto</b>	<input type="text" value="2"/>		<b>Consultar Registro</b>		
5			<b>Nombre de Producto</b>	Goma				
6			<b>Precio Unit.</b>	\$ 15,00				
7								
8								
9								
10								
11			<b>Código de Producto</b>	<b>Nombre de Producto</b>	<b>Precio Unit.</b>			
12			1	Lápiz	\$ 22,00			
13			2	Goma	\$ 15,00			
14			3	Cuaderno	\$ 150,00			
15			4	Regla	\$ 35,00			
16			5	Plasticola	\$ 55,00			
17			6	Escuadra	\$ 45,00			
18								
19								
20								
21								
22								

Option Explicit

```
Private Sub BTN_Consulta_Click()
Dim ValorBuscar As Integer
Dim FilaEncontrado As Long

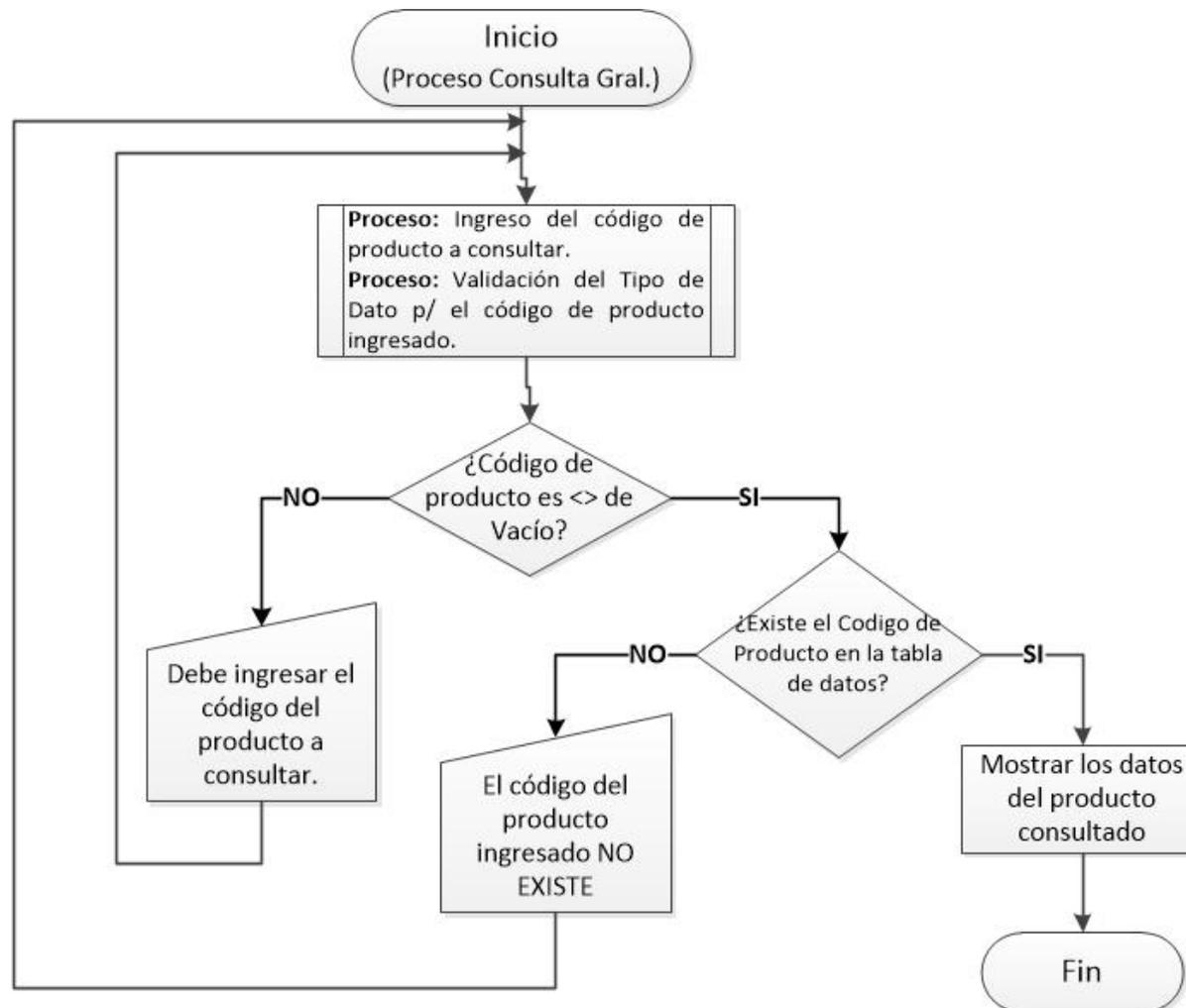
On Error GoTo TratarErrores
If IsEmpty(Range("D4")) Then
MsgBox ("Debe ingresar un Cod. de Producto p/consultar")
Else
ValorBuscar = Range("D4")
FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row
Range("D5") = Cells(FilaEncontrado, 4)
Range("D6") = Cells(FilaEncontrado, 5)
End If
Exit Sub

TratarErrores:
MsgBox ("El Código de Producto Consultado NO EXISTE")
Range("D4").Select

End Sub
```

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

# DIAGRAMA DE FLUJO: Consultar un Registro (GRAL.)



```
Option Explicit
Public AUXcodConsultadoBaja As Long
```

```
Private Sub BTN_Consulta_Click()
Dim ValorBuscar As Long
Dim FilaEncontrado As Long
On Error GoTo TratarErrores
If IsEmpty(Range("D4")) Then
MsgBox ("Debe ingresar un Cod. de Producto p/consultar")
Else
ValorBuscar = Range("D4")
FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row
Range("D5") = Cells(FilaEncontrado, 4)
Range("D6") = Cells(FilaEncontrado, 5)
Range("E4") = "Art. Consultado"
AUXcodConsultadoBaja = Cells(FilaEncontrado, 3)
End If
Exit Sub
TratarErrores:
MsgBox ("El Código de Producto ingresado NO EXISTE")
Range("D4").Select
End Sub
```

```
Private Sub BTN_Eliminar_Click()
Dim FilaEncontrado As Long
Dim sTitulo As String, sMensaje As String, sBotones As String
Dim Respuesta As Integer
```

```
sTitulo = "Confirmación"
sMensaje = "¿Confirma la Eliminación del registro actual? "
sBotones = vbYesNo + vbInformation
```

```
On Error GoTo TratarErrores
If IsEmpty(Range("E4")) Then
MsgBox ("La eliminación de un registro, se realiza sobre un producto consultado. Consulte un producto.")
Else
FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=AUXcodConsultadoBaja, Lookat:=xlWhole).Row
If AUXcodConsultadoBaja = Range("D4") Then
Respuesta = MsgBox(Title:=sTitulo, Prompt:=sMensaje, Buttons:=sBotones)
If Respuesta = vbYes Then
Rows(FilaEncontrado).Delete
MsgBox ("El Producto ha sido Eliminado")
Range("D4") = Empty
Range("D5") = Empty
Range("D6") = Empty
Range("E4") = Empty
AUXcodConsultadoBaja = 0
End If
Else
MsgBox ("Si desea eliminar el registro, No debe modificar el código de producto consultado.")
Range("D4") = AUXcodConsultadoBaja
End If
Exit Sub
TratarErrores:
MsgBox ("El Código de Producto Consultado NO EXISTE ")
End Sub
```

# Ejemplo: Eliminar un Registro (Fila)

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

Baja de Artículos de Librería	
Código de Producto	4 Art. Consultado
Nombre de Producto	Regla
Precio Unit.	\$ 35,00

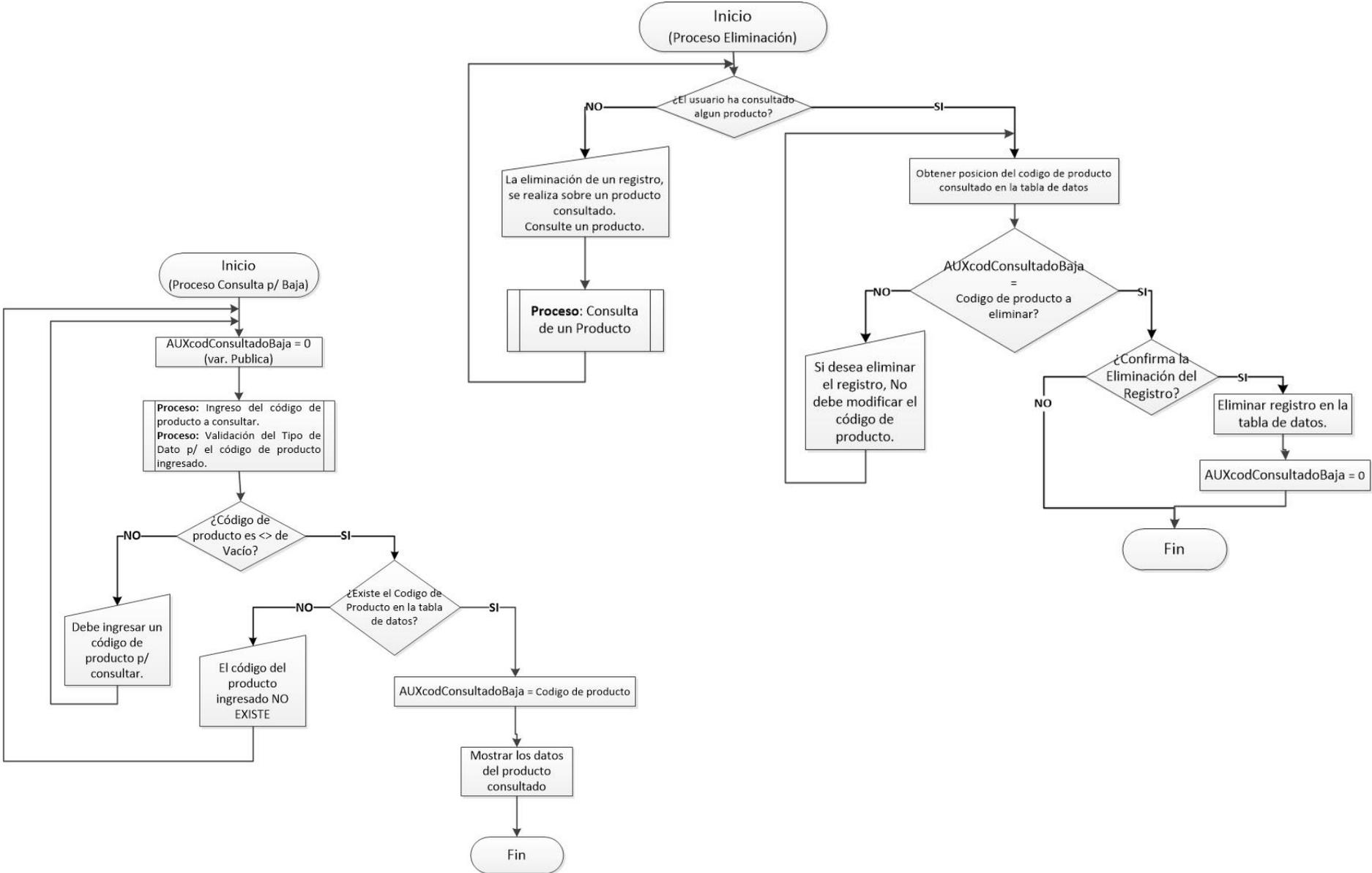
  

Menú	
<input type="button" value="Consultar Registro"/>	<input type="button" value="Eliminar Registro"/>

En este ejemplo, antes de permitir la eliminación de un registro, se verifica que el usuario haya consultado algún registro.

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

# DIAGRAMA DE FLUJO: Eliminar un Registro



```
Option Explicit
Public AUXcodConsultadoMod As Long
```

```
Private Sub BTN_Consulta_Click()
Dim ValorBuscar As Long
Dim FilaEncontrado As Long
On Error GoTo TratarErrores
If IsEmpty(Range("D4")) Then
MsgBox ("Debe ingresar un Cod. de Producto p/consultar")
Else
ValorBuscar = Range("D4")
FilaEncontrado = Range("C12", Range("C12").End(xlDown)).Find(what:=ValorBuscar, Lookat:=xlWhole).Row
Range("D5") = Cells(FilaEncontrado, 4)
Range("D6") = Cells(FilaEncontrado, 5)
Range("E4") = "Art. Consultado"
AUXcodConsultadoMod = Cells(FilaEncontrado, 3)
End If
Exit Sub
TratarErrores:
MsgBox ("El Código de Producto Consultado NO EXISTE")
Range("D4").Select
End Sub
```

```
Private Sub BTN_Modificar_Click()
Dim FilaEncontrado As Long
Dim FilaEncontradoAux As Long
Dim sTitulo As String, sMensaje As String, sBotones As String
Dim Respuesta As Integer
```

```
sTitulo = "Confirmación"
sMensaje = "¿Confirma la Actualizacion del registro actual? "
sBotones = vbYesNo + vbInformation
```

```
On Error GoTo TratarErrores
If IsEmpty(Range("E4")) Then
MsgBox ("La modificación de un registro, se realiza sobre un producto consultado. Consulte un producto.")
Else
If IsEmpty(Range("D4")) Or IsEmpty(Range("D5")) Or IsEmpty(Range("D6")) Then
MsgBox ("Si desea actualizar el registro consultado, todos los parametros deben tener datos")
Else
FilaEncontrado = Range("C11", Range("C11").End(xlDown)).Find(what:=AUXcodConsultadoMod, Lookat:=xlWhole).Row
If AUXcodConsultadoMod = Range("D4") Then
Error (65535)
Else
FilaEncontradoAux = Range("C11", Range("C11").End(xlDown)).Find(what:=Range("D4"), Lookat:=xlWhole).Row
MsgBox ("El Código de Producto con el que se quiere actualizar el registro YA SE ENCUENTRA ASIGNADO a otro producto")
Range("D4") = AUXcodConsultadoMod
End If
End If
End If
Exit Sub
TratarErrores:
Respuesta = MsgBox(Title:=sTitulo, Prompt:=sMensaje, Buttons:=sBotones)
If Respuesta = vbYes Then
Cells(FilaEncontrado, 3) = Range("D4")
Cells(FilaEncontrado, 4) = Range("D5")
Cells(FilaEncontrado, 5) = Range("D6")
MsgBox ("El Producto ha sido Actualizado")
Range("D4") = Empty
Range("D5") = Empty
Range("D6") = Empty
Range("E4") = Empty
AUXcodConsultadoMod = 0
End If
End Sub
```

# Ejemplo: Modificar un Registro

Código de Producto	Nombre de Producto	Precio Unit.
1	Lápiz	\$ 22,00
2	Goma	\$ 15,00
3	Cuaderno	\$ 150,00
4	Regla	\$ 35,00
5	Plasticola	\$ 55,00
6	Escuadra	\$ 45,00

Actualización de Artículos de Librería	
Código de Producto	1 Art. Consultado
Nombre de Producto	Lápiz
Precio Unit.	\$ 22,00

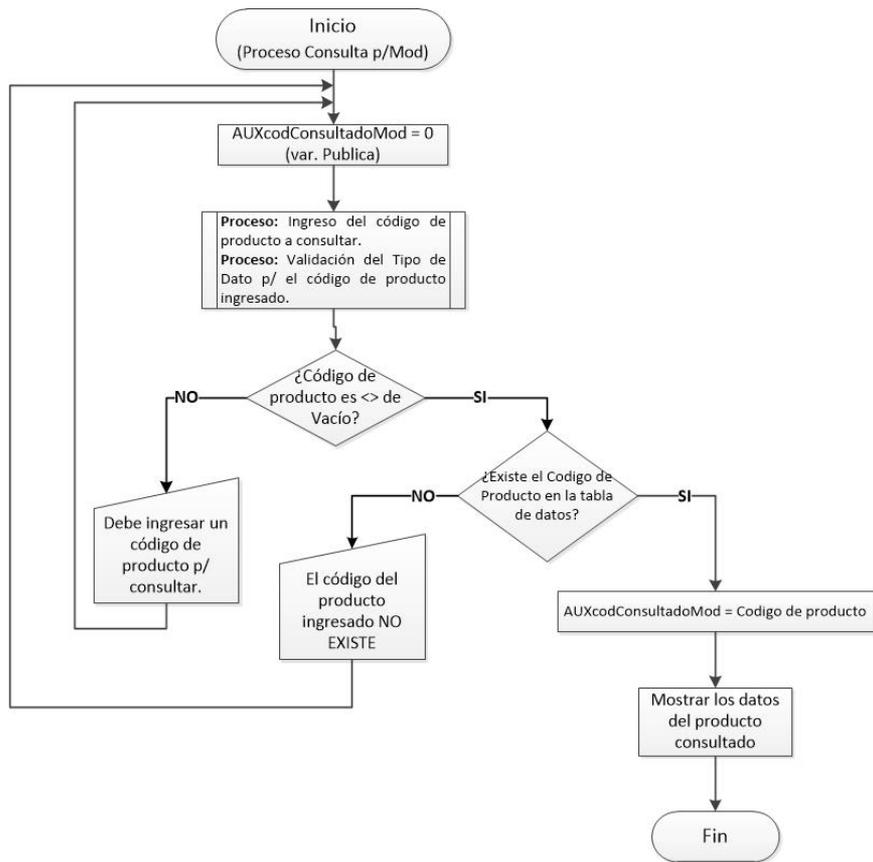
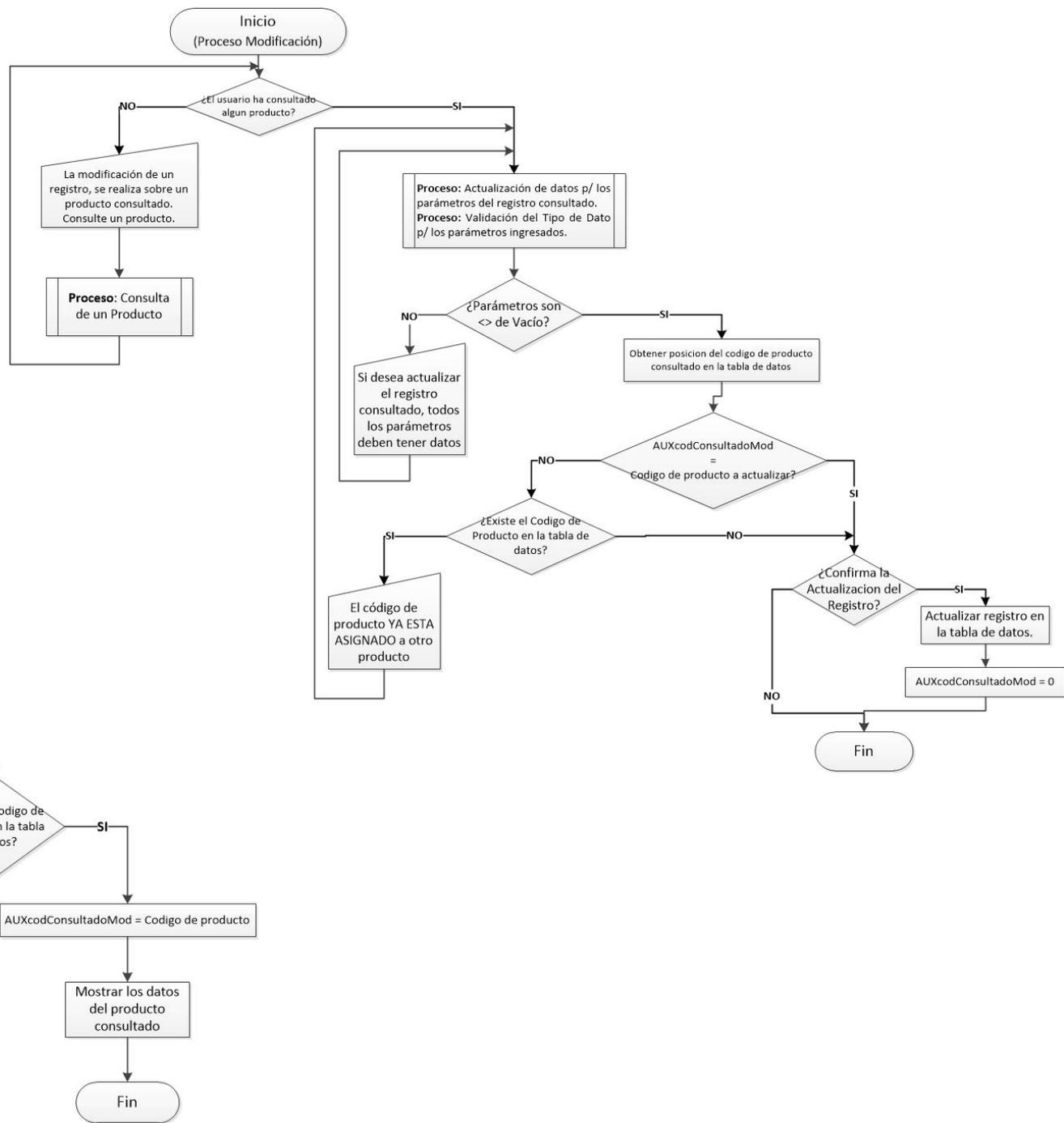
  

Menú	
<input type="button" value="Consultar Registro"/>	
<input type="button" value="Modificar Registro"/>	

En este ejemplo, antes de permitir la modificación de un registro, se verifica que el usuario haya consultado algún registro.

En este ejemplo, suponemos que la validación de datos, del código de artículo a consultar, se encuentra en el Evento Change de Worksheets.

# DIAGRAMA DE FLUJO: Modificar un Registro



# Ejemplo: Listar Registros

## Listar Empleados de un Área

### Option Explicit

```
Private Sub BTN_ListarReg_Click()  
Dim i As Long  
Dim sArea As String  
Dim UltimaFila As Long  
Dim NroFila As Long  
  
sArea = Range("C5").Value  
NroFila = 9  
UltimaFila = Worksheets("Tabla Empleados").Range("A1").End(xlDown).Row  
  
For i = 2 To UltimaFila  
    If Worksheets("Tabla Empleados").Cells(i, 3).Value = sArea Then  
        Cells(NroFila, 2).Value = Worksheets("Tabla Empleados").Cells(i, 1).Value  
        Cells(NroFila, 3).Value = Worksheets("Tabla Empleados").Cells(i, 2).Value  
        Cells(NroFila, 4).Value = Worksheets("Tabla Empleados").Cells(i, 3).Value  
        Cells(NroFila, 5).Value = Worksheets("Tabla Empleados").Cells(i, 4).Value  
        NroFila = NroFila + 1  
    End If  
Next i  
  
End Sub  
  
Private Sub Worksheet_Change(ByVal Target As Range)  
Dim UltimaFila As Long  
  
Select Case Target.Address  
    Case "$C$5"  
        UltimaFila = Range("B1000000").End(xlUp).Row  
        If UltimaFila > 8 Then  
            Range(Cells(9, 2), Cells(UltimaFila, 5)) = Empty  
        End If  
    End Select  
  
End Sub
```

	A	B	C	D	E	F
1						
2						
3						
4						
5			Area: Producción		Listar Registros	
6						
7						
8		NOMBRE Y APELLIDO	TELEFONO	ÁREA	SALARIO	
9		Laura Morales	290999	Producción	\$ 21.419,00	
10		Luisa Rodríguez	916236	Producción	\$ 24.232,00	
11		Diana Parra	680999	Producción	\$ 26.968,00	
12		Sandra Muñoz	335912	Producción	\$ 37.349,00	
13		Lina Gonzales	242005	Producción	\$ 35.342,00	
14		Jeimmy Neuta	732173	Producción	\$ 38.534,00	
15		Nataly Cuervo	425163	Producción	\$ 31.855,00	
16						
17						
18						
19						
20						
21						
22						
23						