



POR QUÉ PROGRAMAR ?

Cansado de las tareas rutinarias ?

Víctor Julio Alvarez Q.

Ingeniero Geógrafo

Especialista en Planeación, Experto en SIG

PERSONALIZACIÓN... POR QUÉ ?

La personalización de una determinada actividad se manifiesta en:

- Existencia de **tareas repetitivas** frecuentes que generan **rutina, tedio, sueño, cansancio** y por consiguiente son fuente **de errores, imprecisiones y repeticiones innecesarias**.
- Trabajos **rutinarios pero esporádicos** que implican **olvidos de procesos** y por tanto pérdidas de tiempo.
- Expertos Vs Necesidades diarias... quién lo hace ?



Cómo se hace ?

Identificado **un proceso** se procede a automatizarlo y luego **se amplía su aplicación al conjunto** de eventos que se puedan presentar.

Esto se hace con un conjunto de instrucciones que permiten la repetición de procesos con diferentes datos o variables.

PERSONALIZACIÓN ?

- Búsqueda en la WEB ?
- Puedo hacerlo ?
- Un amigo ?
- Contratar a alguien ?
- Dejar así... no actuar !!

PROGRAMACIÓN...

- a. Ambiente mundial, futuro*
- b. Virtualidad*
- c. Complejidad : lenguajes*
- d. Rutinas : macrolenguajes*

Sistemas operativos



IDE
API



IDE

Integrated Development Environment

Normalmente incluye : editor de código fuente, compilador y/o un intérprete, constructor herramientas de automatización, un depurador.

Eclipse... Android Studio...

NeatBeans... Visual Studio... VBA Excel..

API

Application Programming Interface

Diseñadas específicamente para un uso expreso, para permitir la comunicación entre aplicaciones.

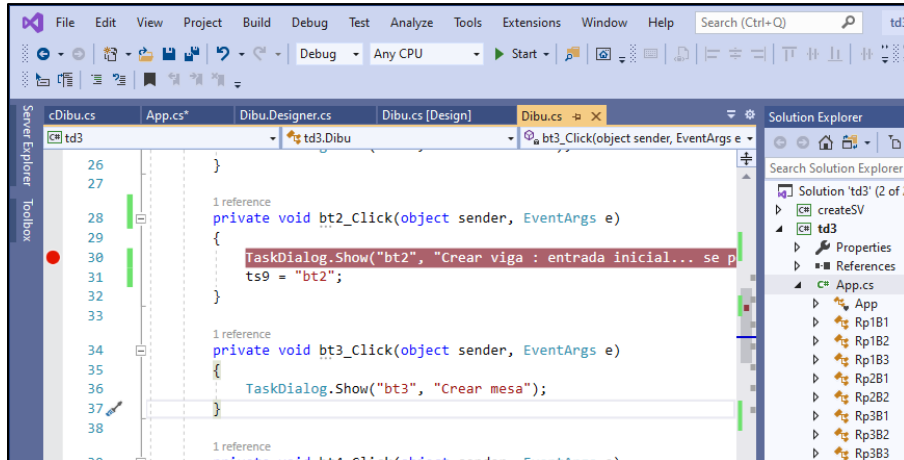
Facebook... Excel... Word....

Autocad... Revit...

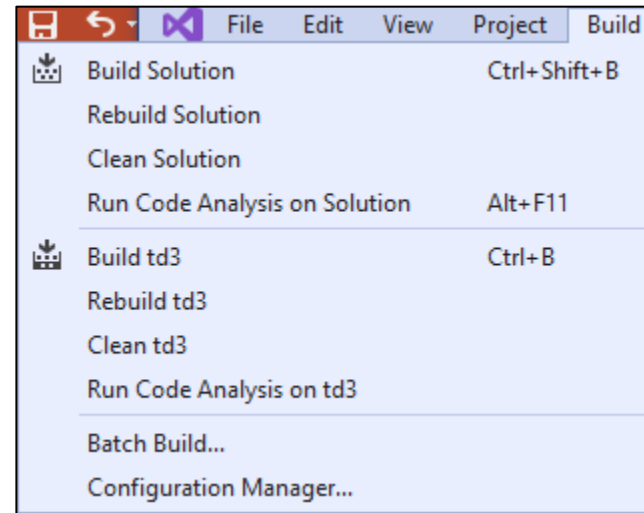
IDE *Integrated Development Environment* :

Visual Studio

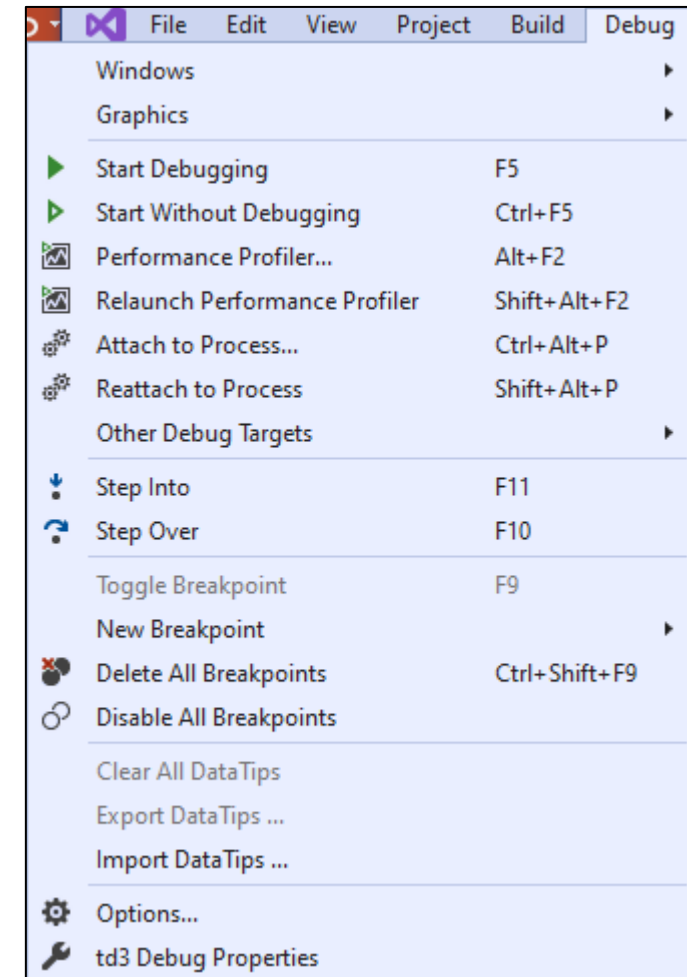
editor de código fuente



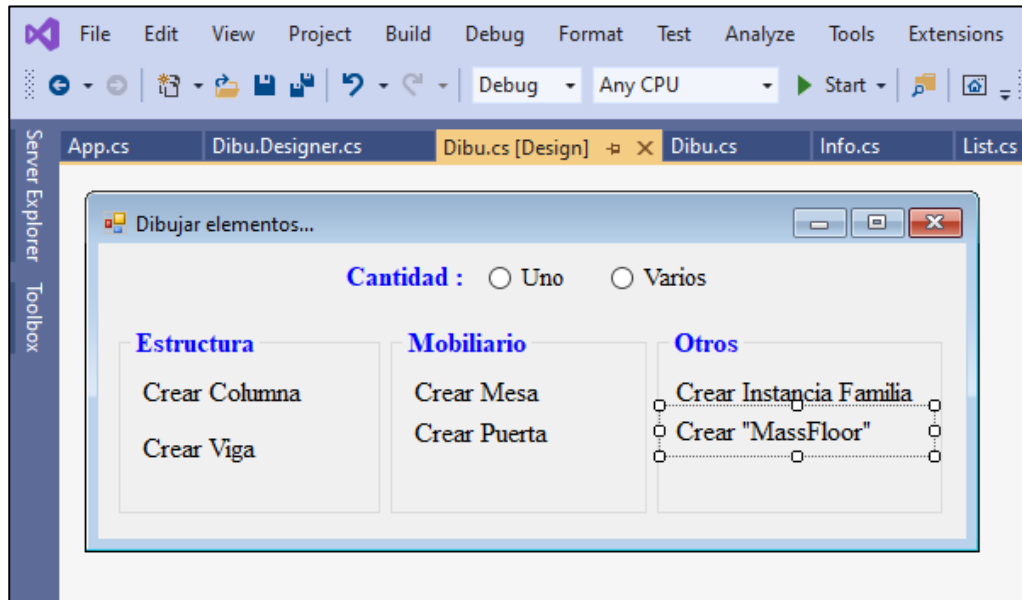
compilador y/o un intérprete



un depurador.



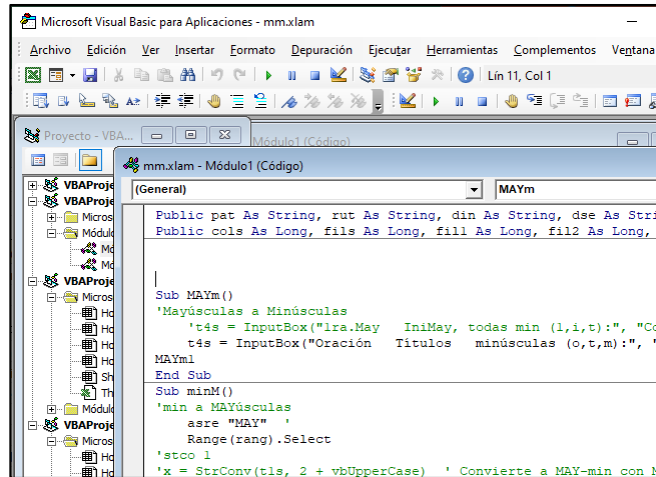
constructor herramientas de automatización



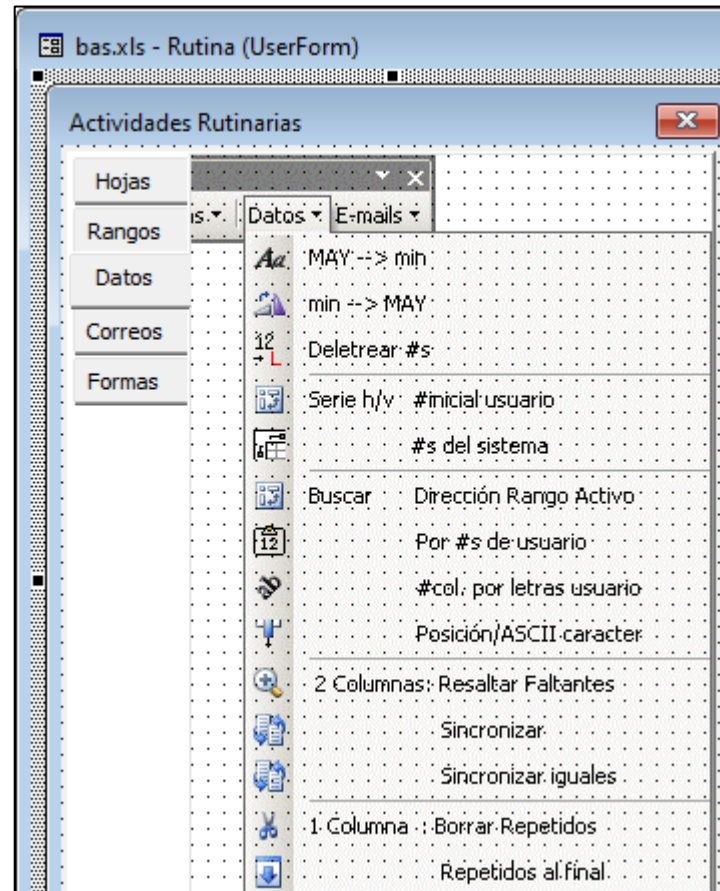
IDE *Integrated Development Environment* :

VBA Excel

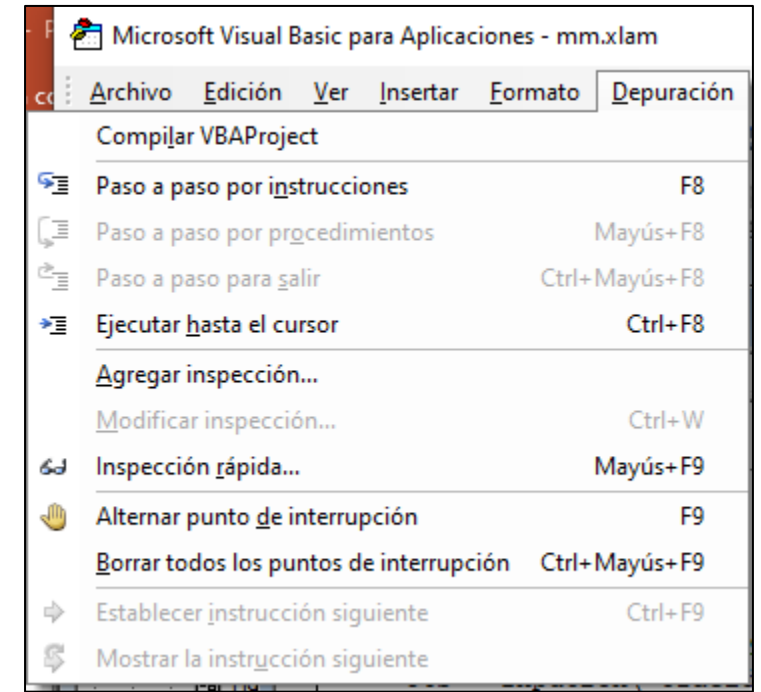
editor de código fuente



constructor herramientas
de automatización



compilador y/o un intérprete
un depurador

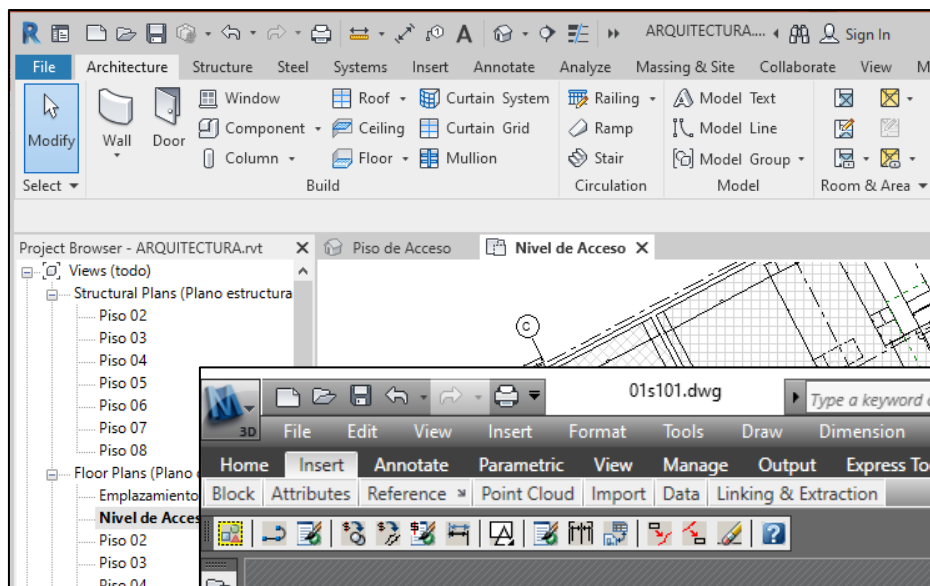


API

Application Programming Interface

Diseñadas para un uso expreso, para permitir la comunicación entre aplicaciones.

Revit



Excel

Pres.xls [Modo de compatibilidad] - Excel

1	Iniciar					
2	Piso	Tema/ COD	DESCRIPCIÓN	UN	CantP	VrUnitP
378		'200110	Ing. Redes	Mes	0	0
379		'200111	Secretaria	Mes	0	0
380		'200112	Auxiliares de Ingeniería (inspección obra)	Mes	0	0
381		'200113	Auxiliar Contaduría	Mes	0	0
382		'200114	Ingeniero HSEQ		0	0
383		'200115	Almacenista		0	0
384		'200116	Ing. Presupuestos progr		0	0
385		'200117	Residente administrativo		0	0
386		'200118	Ejero		0	0
387		'200119	Ayudantes ambientales		0	0
388		'200120	IVA Contrato		0	0
389						
390						
391						
392						
393						
394	21		Espacios			
395						
396		2101	Comerciales			
397	06	210101	Oficinas	MZ	645.18	

Control Presupuestal Arquitectónico

Pruebas Fundamentos

ItemsPres

Registros

Varios

ESTADÍSTICAS

Contratistas

Despiece hierros

Cantidades/Modelo

Cantidades de Hierros

Varillas: 301

Kgs 1229.87

3414 Cable

5102 Canal Doble

5103 Canal sencillo, Acequia, Vallado

2321 Cantera

2304 Central de energía

2311 Cerca

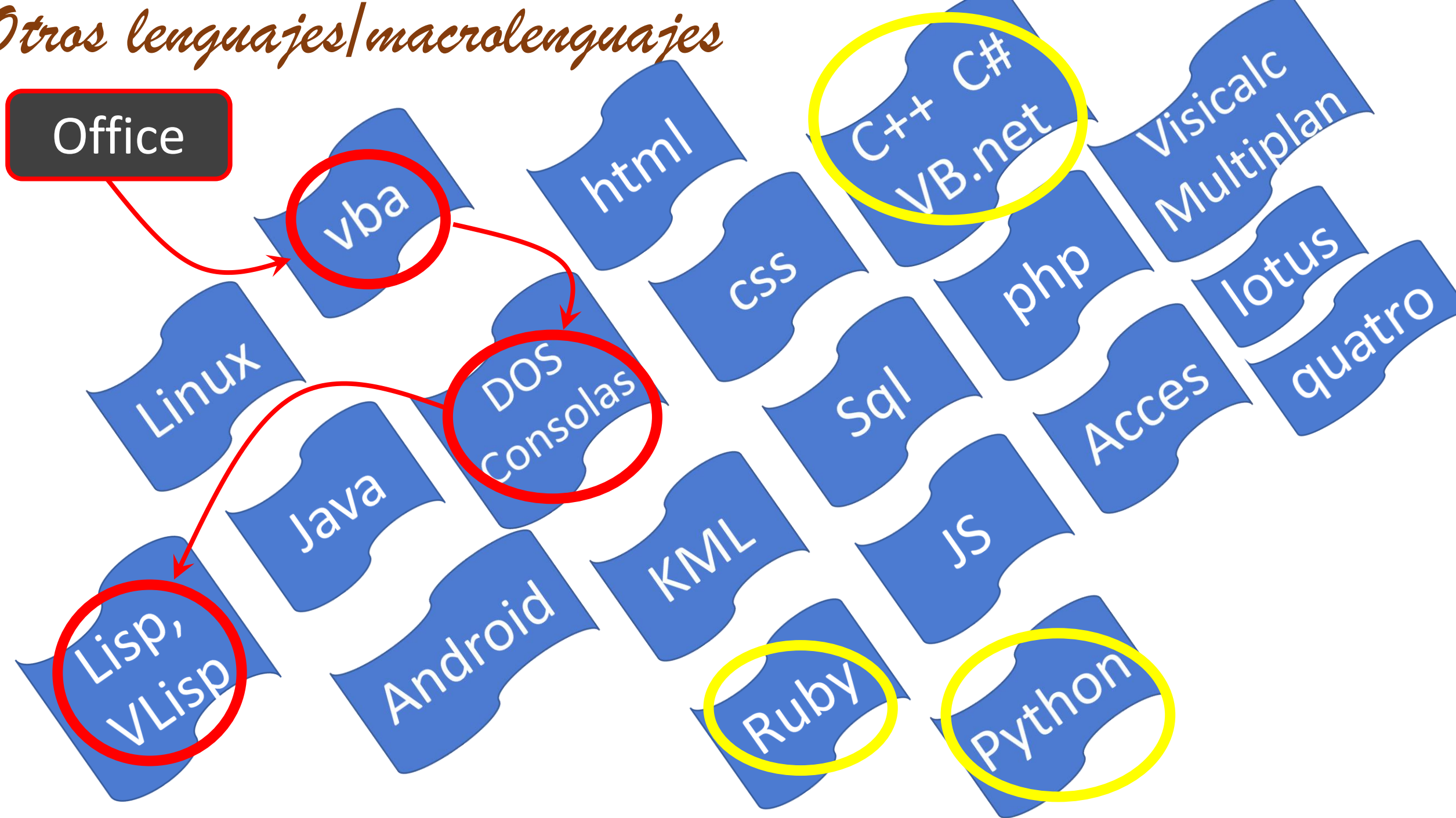
Autocad

Software Gráfico... (sig)



Otros lenguajes/macrolenguajes

Office



QUIÉNES ?

- a. Ingenieros de sistemas ?***
- b. Todos, no expertos ?***
- c. Rutinas diarias menospreciadas ?***
- d. Eventos espaciales ?***

APLICACIONES ESPECÍFICAS

APLICACIONES

BdD

→ Bancos : [TransTerceros.xls](#)

Consultantes : [ContNG.xlsm](#)

Pausas mentales : [pma.xls](#)

Evaluación peajes : [evs.xls](#)

Riesgos Psicosociales : [RiPsi10.xlsm](#)

Registro Correos : [LCorreos.xls](#)

Diseño WEB : [SubDir.xls](#)

Negocios : [Ayudas.xls](#)

Conexiones

Excel → Autocad : [Pres.xls](#)

Control presupuesto : [ConPrAr.xlsm](#)

Excel → Google Earth : [SitiosGE.xls](#)

Excel → Word : [Edno.xls](#)

Excel → SITP : [rutas.php](#)

Excel → Consola → [NotePad](#)

Controles

Control Clientes : [Client.xls](#)

Caja menor : [Camen.xls](#)

Flujo de caja : [FdC.xlsm](#)

Diseño curricular : [Plantilla.xls](#)

Licencias : [Lic-varias.xls](#)

Control producción : [Cont.xls](#)

Manuales

Autocad : [pam.xls](#)

Small World : [misw.xls](#)

Páginas WEB : [web.xls](#)

C# - Revit : [Revit.xlsm](#)

Capacitación : [demos.xlsm](#)

APLICACIONES

uba : Excel - Autocad => .lsp : Torres Telecom

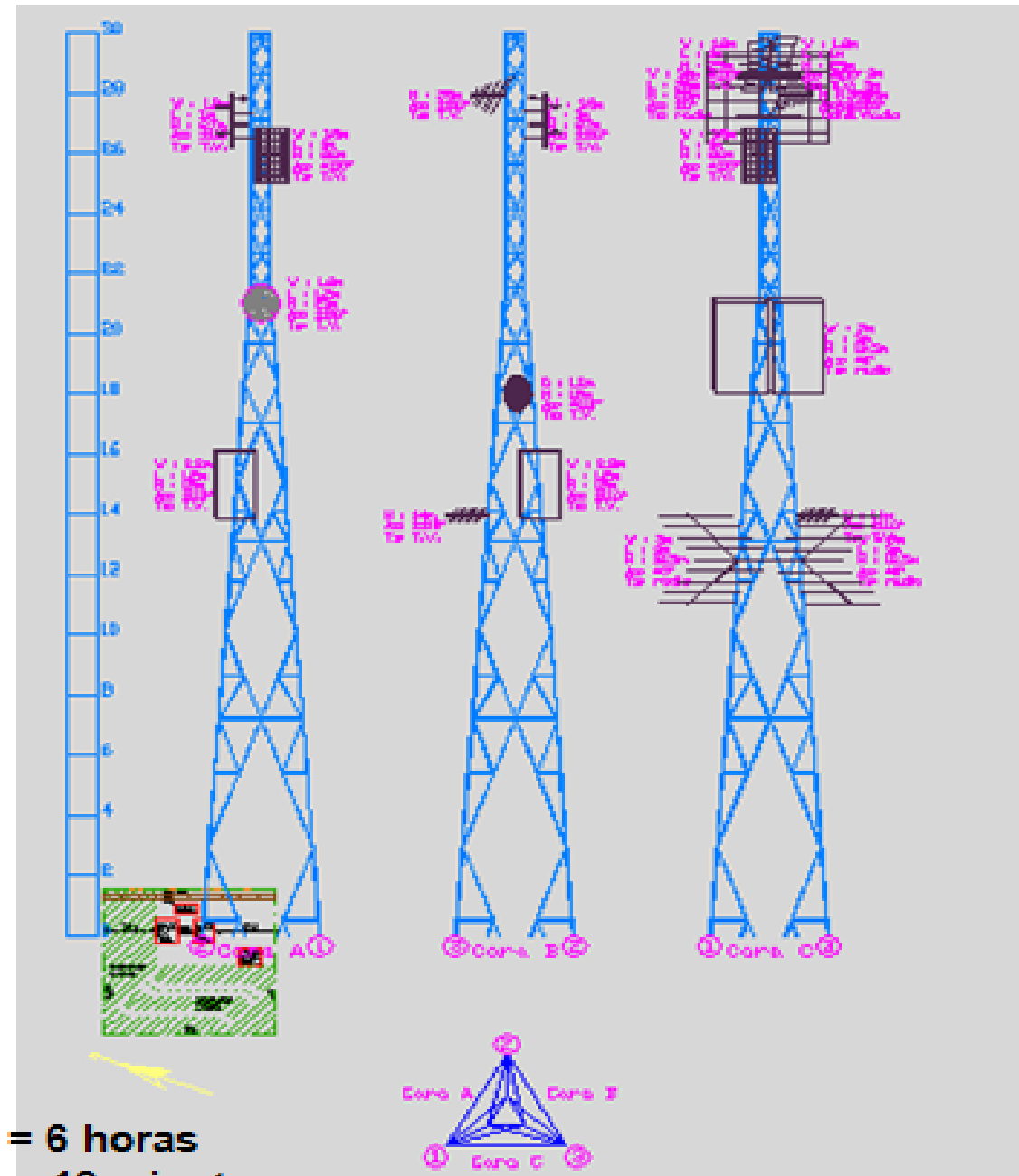
b.- Elaboración de tabla con base en los datos de campo, la cual será leída automáticamente para elaborar el dibujo:

Linea	Antena	Distancia	Frecuencia	Tipo de Antena	Tipo de Servicio	Ancho de Banda	Servicio
30	2	3.5	23574	01	a	290	
V	AZ	HAnt	TA	Nom	Alto	Ancho	Servicio
3	110	14	11	Log Per	2.2	1.2	T.V.
B	110	18	1	Panel	1.2	1.2	T.V.
A	110	21	2	Panel	1.2	1.2	T.V.
2	110	15	3	Log Per	2.2	1.2	T.V.
4	0	28	4	Yagi 2	3	3	radio
C	0	19.5	5	Yagi 2	3	3	radio
4	0	12.5	6	Yagi 2	3	3	radio
1	110	26	7	Log Per	1.8	1	T.V.
2	110	27	10	Log Per	1.8	1	T.V.
3	110	28	9	Log Per	1.8	1	T.V.
4	110	29	8	Log Per	1.8	1	T.V.

c.- Dibujo automáticos de torres y ubicación de antenas:

El éxito de esta aplicación consiste en la ubicación exacta en altura de las antenas, su orientación y su visualización en cada uno de los lados, especialmente de las ubicadas en las aristas de las torres, que deben verse en dos lados según el caso.

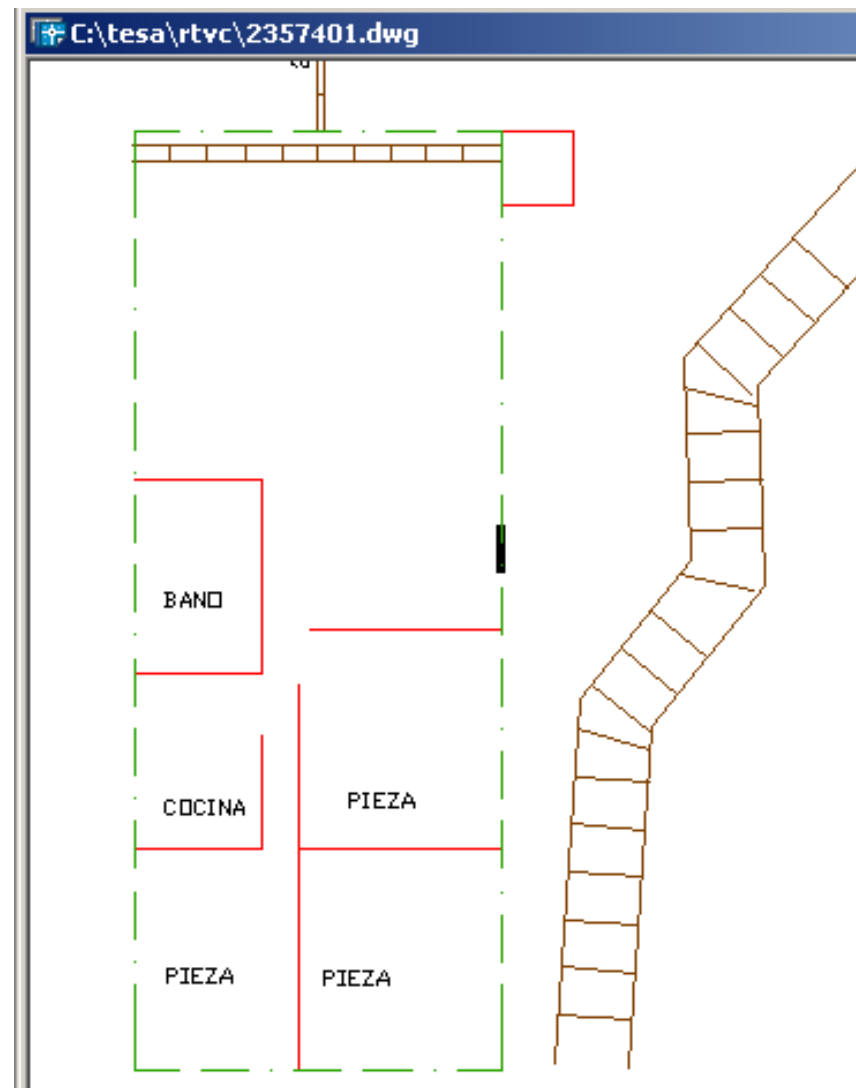
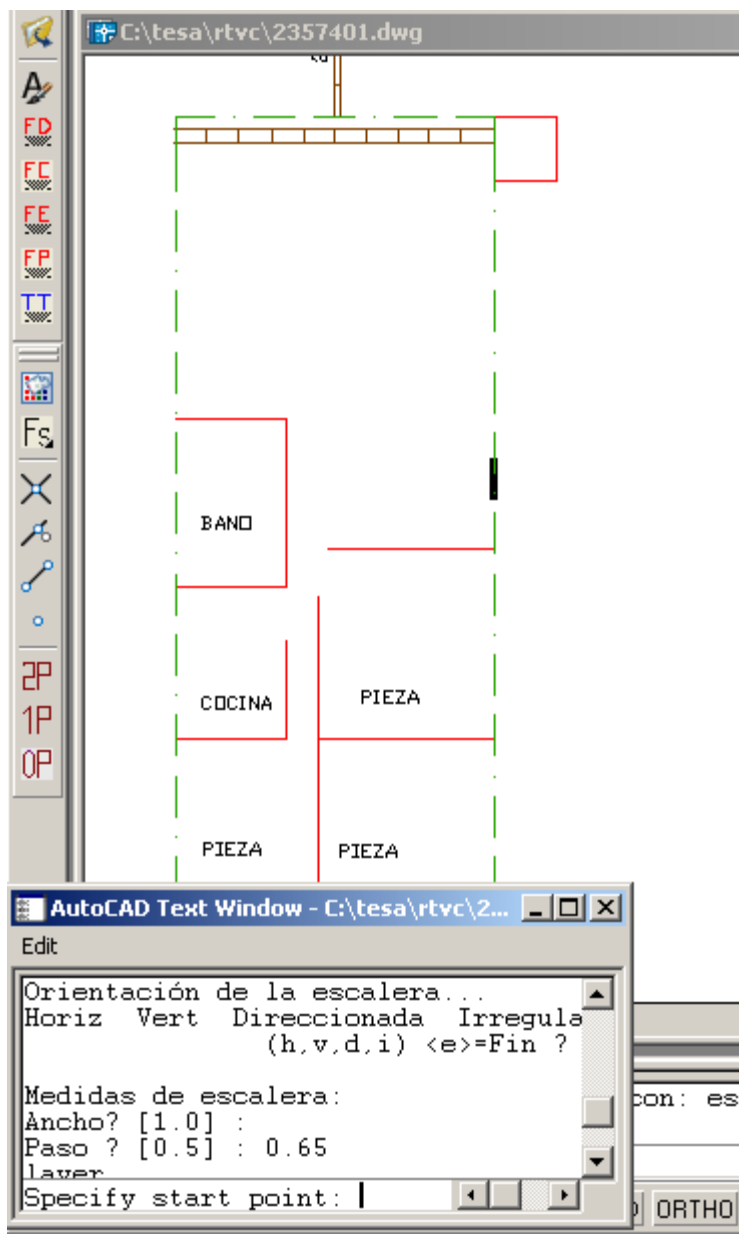
Se debe tener en cuenta que las torres tienen una base rectangular que disminuye en altura lo cual afecta la altura real de posición de las antenas.



Tiempo en forma manual = 6 horas
Tiempo proceso automatizado = 10 minutos

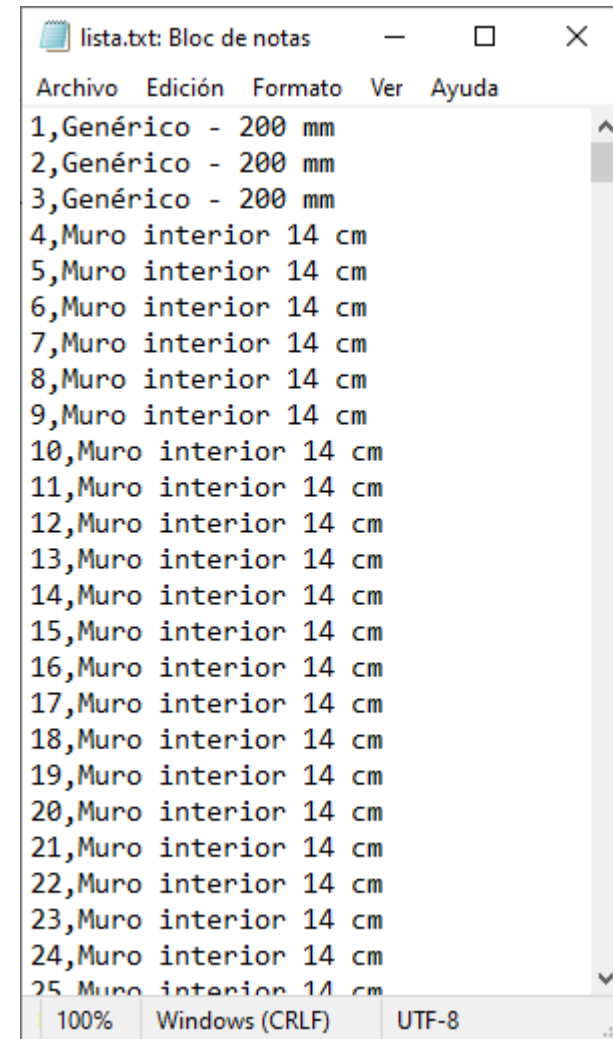
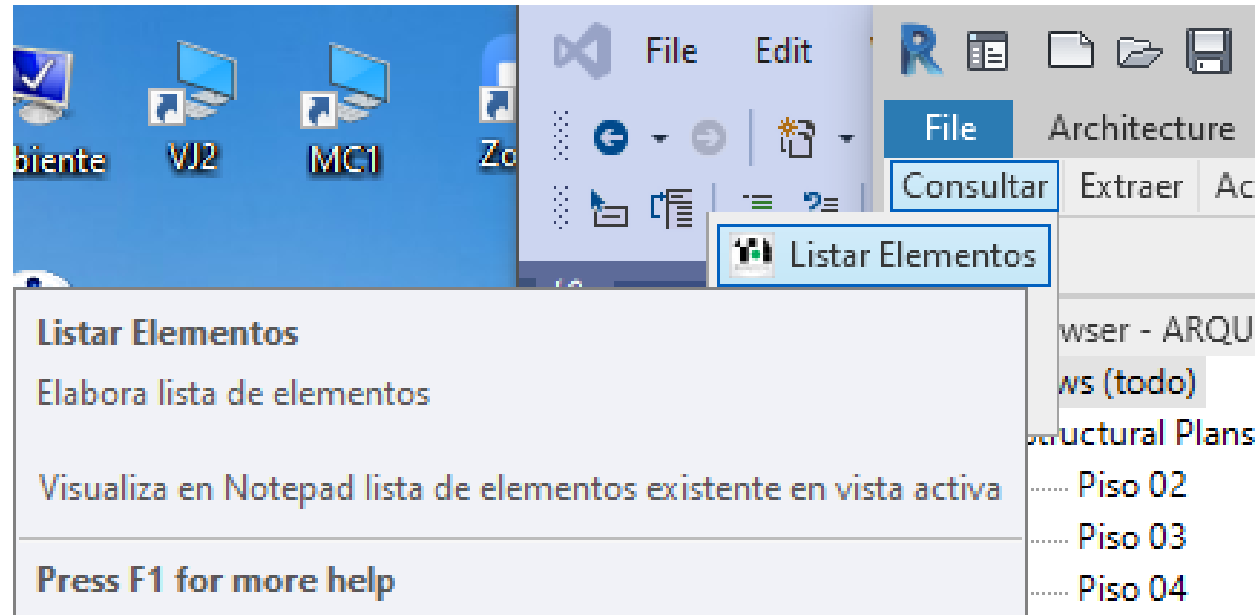
APLICACIONES

uba : Excel – Autocad => Escaleras



APLICACIONES

C# → Revit → Notepad → Excel



EXCEL

Potencialidades

Página principal y capacidades...



CAPACIDAD LIBRO

Hojas

Cant hojas	1024: depende RAM y Disco
Cant Cols :	16.384
Cant Filas :	1'048.576
Cant Colores	16 millones

Celdas:

Cantidad	17.179.869.184
Cars/Celda	$32.767 = 2^{15} - 1 =$ en cálculos.
	1.024 = visibles e imprimibles
1 punto	$1/72" = 0,035$ cm.
Altura máx fila	409 ptos. = $5.68" = 14.4$ cm
Ancho Col	255

Otros

Undo máx	100
Precisión	15 dígitos

Activación VBA... centro de confianza

Excel → Archivo → Opciones

General
Fórmulas
Revisión
Guardar
Idioma
Avanzadas
Personalizar cinta de opciones
Barra de herramientas de acceso rápido
Complementos
Centro de confianza

Protección de la privacidad

Microsoft concede gran importancia a la privacidad. Para obtener ayuda a proteger la privacidad, consulte las declaraciones de privacidad de Microsoft.

[Mostrar la declaración de privacidad de Microsoft Excel](#)
[Declaración de privacidad de Office.com](#)
[Programa de mejora de la experiencia del cliente](#)
[Declaración de privacidad de la Herramienta para enviar comentarios](#)

Seguridad y más información

Para más información sobre la protección de privacidad y seguridad, consulte el Centro de confianza de Microsoft Excel.

[Informática de confianza de Microsoft](#)

Centro de confianza de Microsoft Excel

El Centro de confianza contiene la configuración de seguridad y privacidad. Esta configuración garantiza la seguridad del equipo. Le recomendamos que no la cambie.

[Configuración del Centro de confianza...](#)

Editores de confianza
Ubicaciones de confianza
Documentos de confianza
Catálogos de aplicaciones de confianza
Complementos
Configuración de ActiveX
Configuración de macros
Vista protegida
Barra de mensajes
Contenido externo
Configuración de bloqueo de archivos
Opciones de privacidad

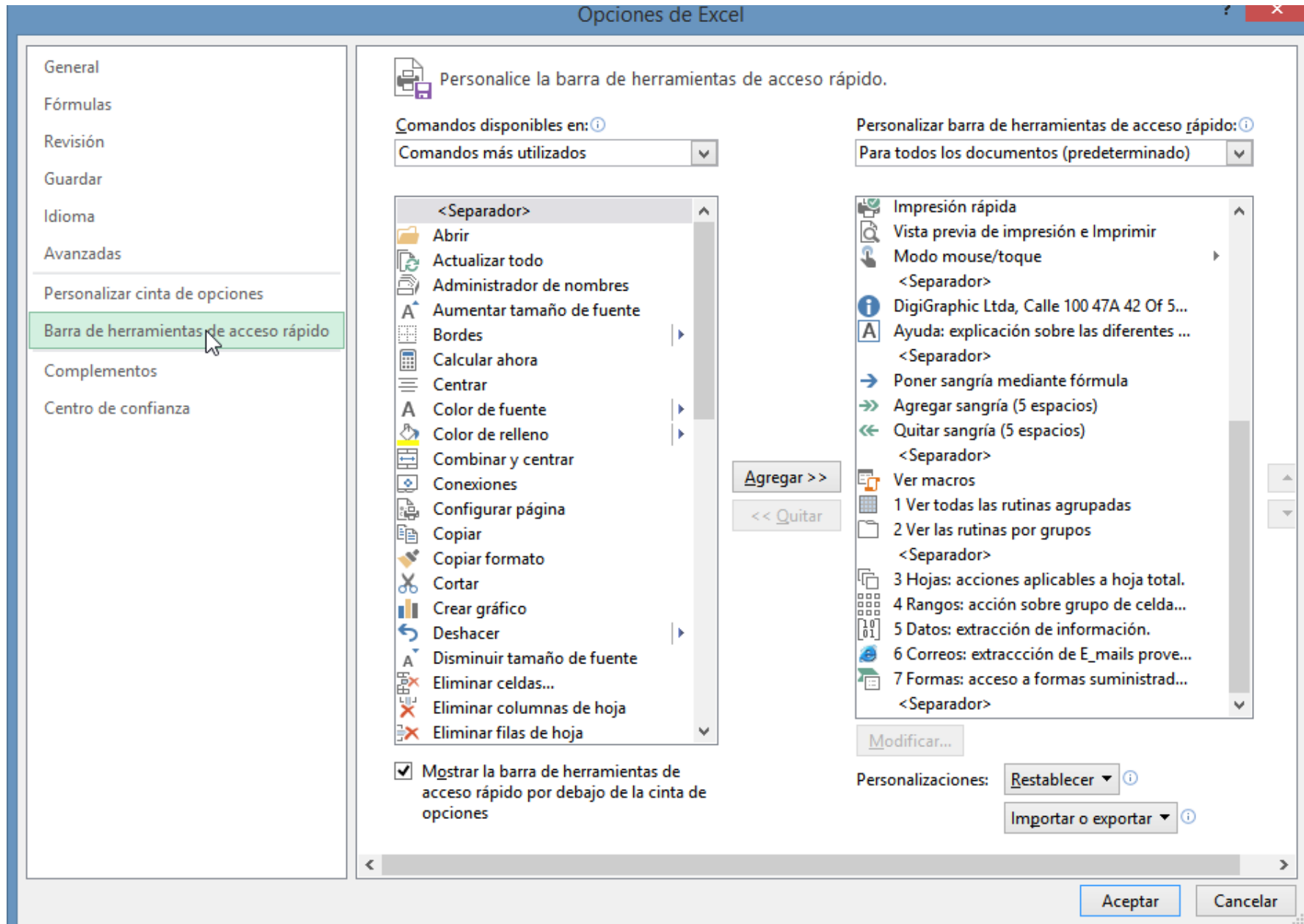
Configuración de macros

☐ Deshabilitar todas las macros sin notificación
☐ Deshabilitar todas las macros con notificación
☐ Deshabilitar todas las macros excepto las firmadas digitalmente
☒ **Habilitar todas las macros (no recomendado; puede ejecutarse código no firmado)**

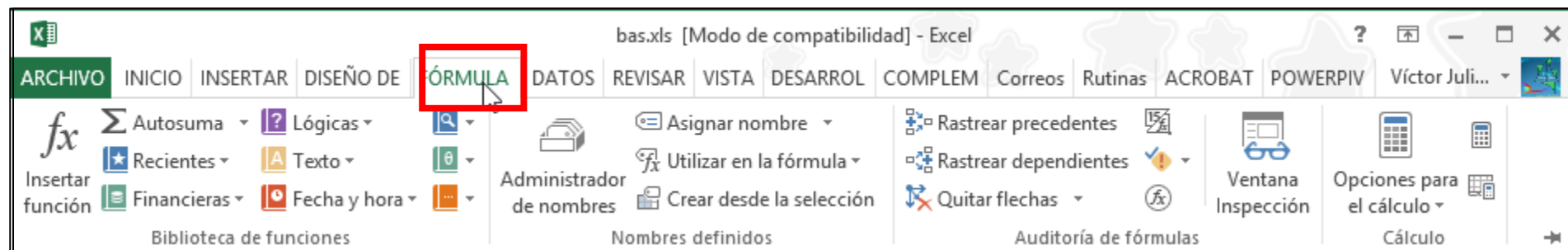
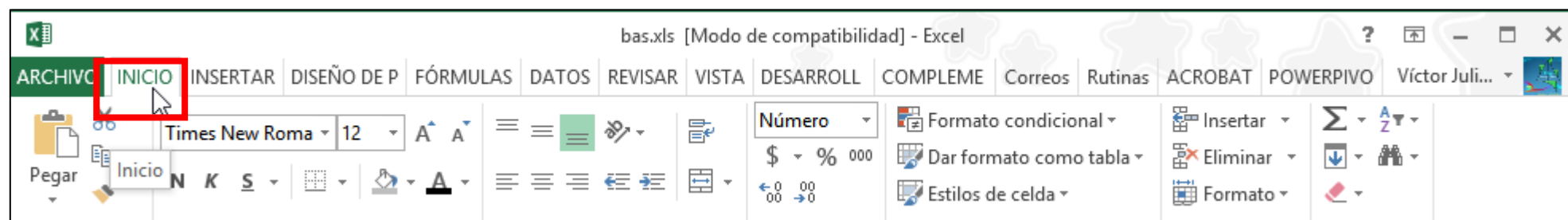
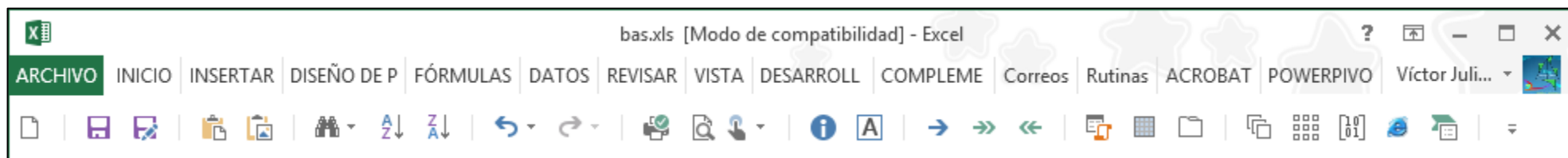
Configuración de la macro del programador

☒ **Confiar en el acceso al modelo de objetos de proyectos de VBA**

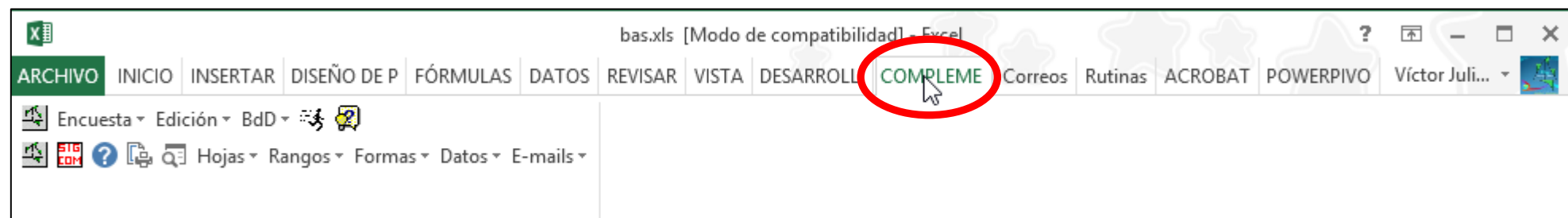
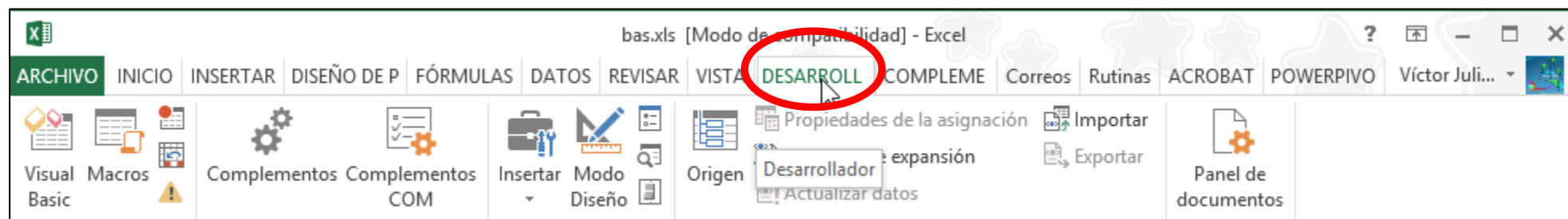
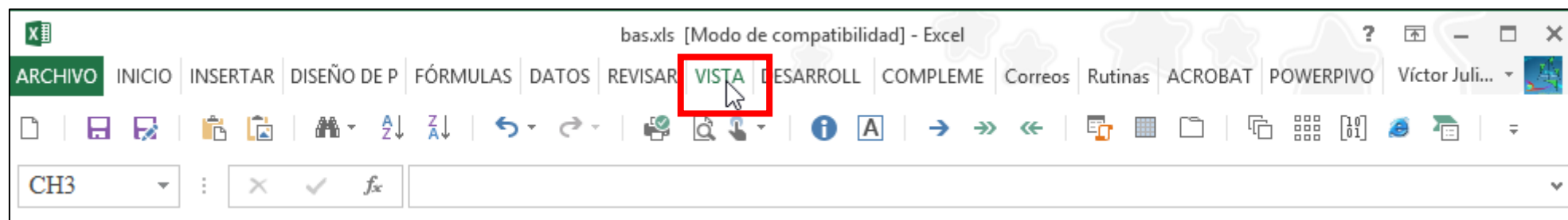
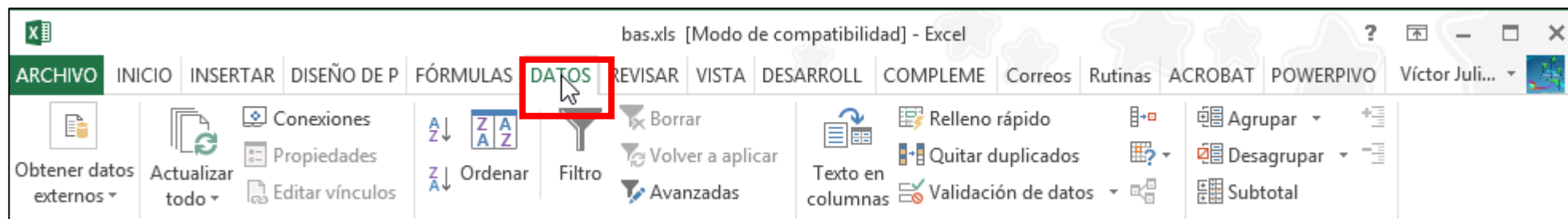
Personalización... barra de acceso rápido



Menús y submenús...



Menús y submenús...



INTERACTIVIDAD

- En toda aplicación es fundamental la **comunicación** entre el sistema y el usuario.
- El sistema necesitará para su procesamiento los **datos de usuario: Entrada**
- El **resultado** de los procesos, **debe ser informado** al usuario: **Salida**
- La **efectividad** de la interacción depende de la **ubicación** en el código.
- Toda interacción es **programada** por el hombre... la máquina es gobernada por este y no lo contrario.

Herramientas: Teclado, Mensajes, Archivos



La actividad de digitalización tiene su propia barra de herramientas con las opciones y teclados más comunes.

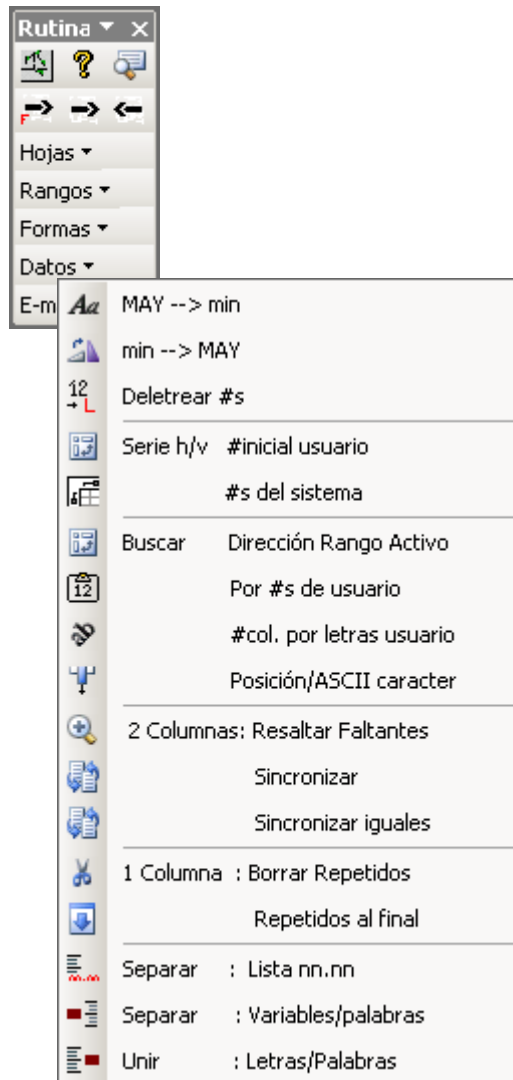
Los teclados virtuales independientes se activan con botones: **FD, FR, FE, FP**; son temporales y desaparecen con la 1ra. función usada, pero quedan activas en teclado físico.



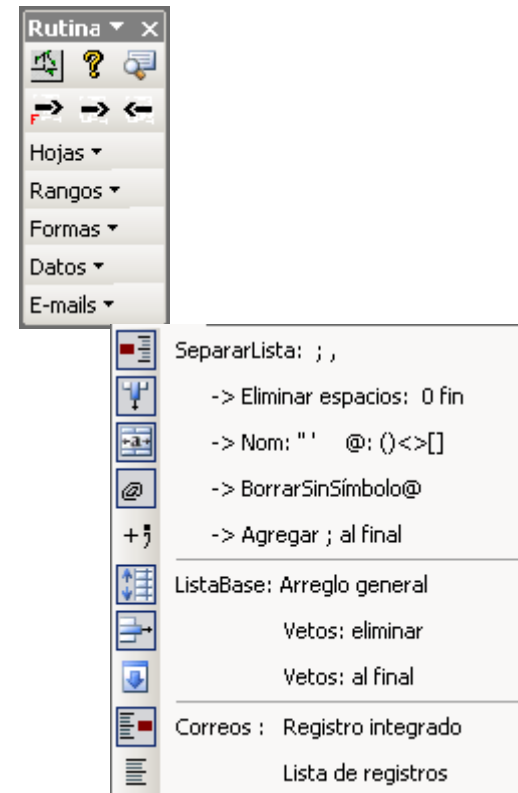
El conjunto de 8 teclados virtuales permanece abierto para ser usados en cualquier momento; muy útil en actividades de edición, estructuración y producción.

Barras de herramientas

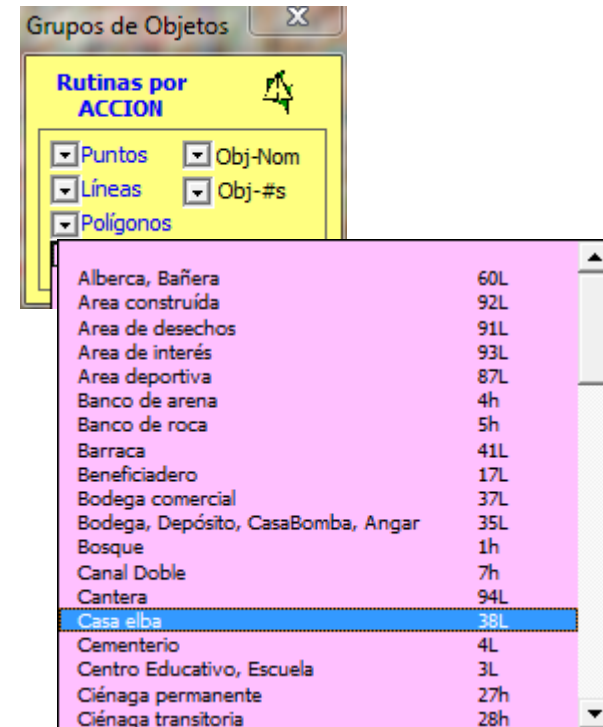
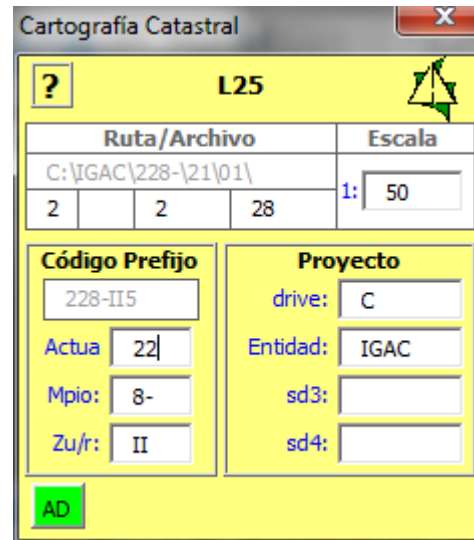
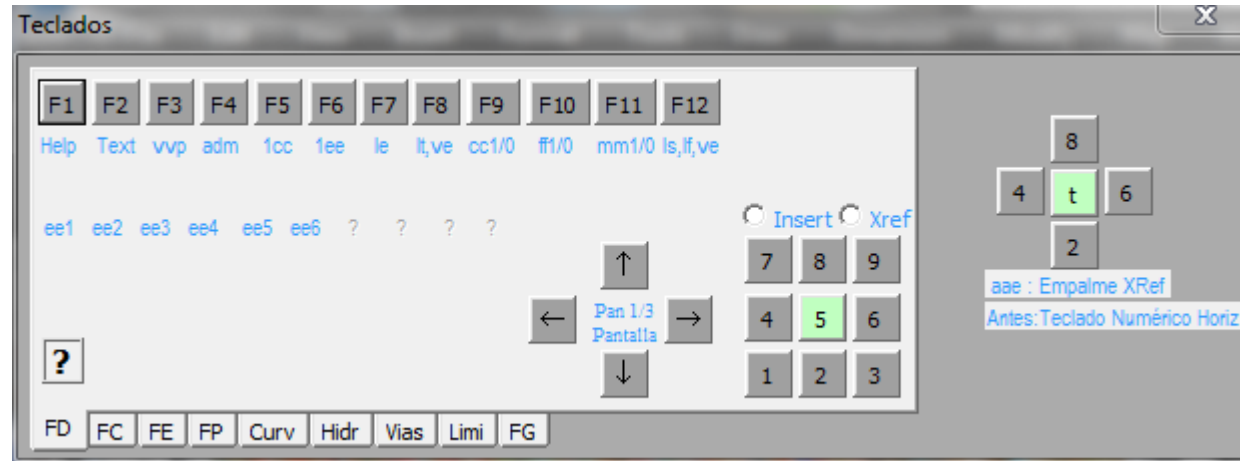
Textos y listados



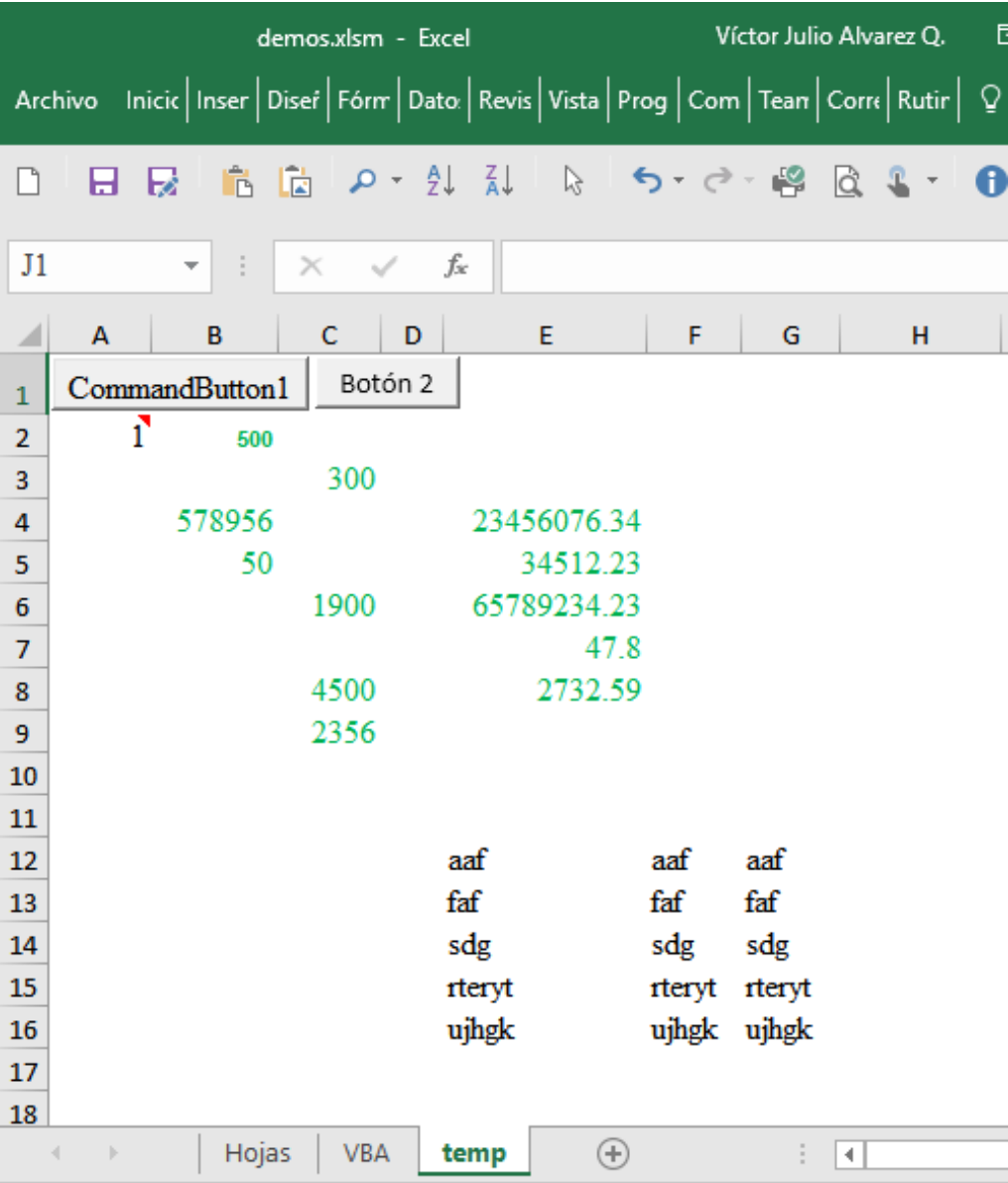
Manejo E_mails



VBA Acad – Cuadros de diálogo

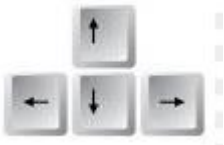


Navegación en hoja



Movimientos rápidos

Uso de teclas : End y



	B	C	D	E	F	G	H	I	J
21	DEMUNDE	MU	CIUDAD	ID	Nombre	Sigla	km2	pobl	
22	05002	05	002	ABEJORRAL	1	Abejorral	Abej	497	18779
23	05004	05	004	ABRIAQUÍ	2	Abriaquí	Abri	293	1870
24	05021	05							3278
25	05030	05							30561
26	05031	05	031	AMALFI	5	Amalfi	Amal	1224	22860
27	05034		034	ANDES	6	Andes	Ande	449	47747
28	05036		036	ANGELÓPOLIS	7	Angelópolis	Ange	87	9631
29	05038		038	ANGOSTURA	8			392	10832
30	05040		040	ANORÍ	9			1447	18166
31	05042	05	042		10	Antioquia	Anti	499	25395
32	05044	05	044	ANZÁ	11	Anzá	Anzá	256	7619
33	05045	05	045	APARTADÓ	12	Apartadó	Apar	607	206885
34	05051	05	051	ARBOLETES	13	Arboletes	Arbo	718	45710

Inicial	Intermedias	Final
Ocupada	vacías	Ocupada
Ocupada	ocupadas	Ocupada
Vacía	vacías	Ocupada

Sub Macro1()

'

' Macro1

'

Range("E4").Select

Range(Selection, Selection.End(xlDown)).Select

Range("E12").Select

Range(Selection, Selection.End(xlToRight)).Select

Range("E4").Select

Range(Selection, Selection.End(xlToLeft)).Select

Range("C16").Select

Range(Selection, Selection.End(xlUp)).Select

Range("E12").Select

Range(Selection, Selection.End(xlDown)).Select

Range(Selection, Selection.End(xlToRight)).Select

Application.Goto Reference:="R1C3763"

Range("E12:G16").Select

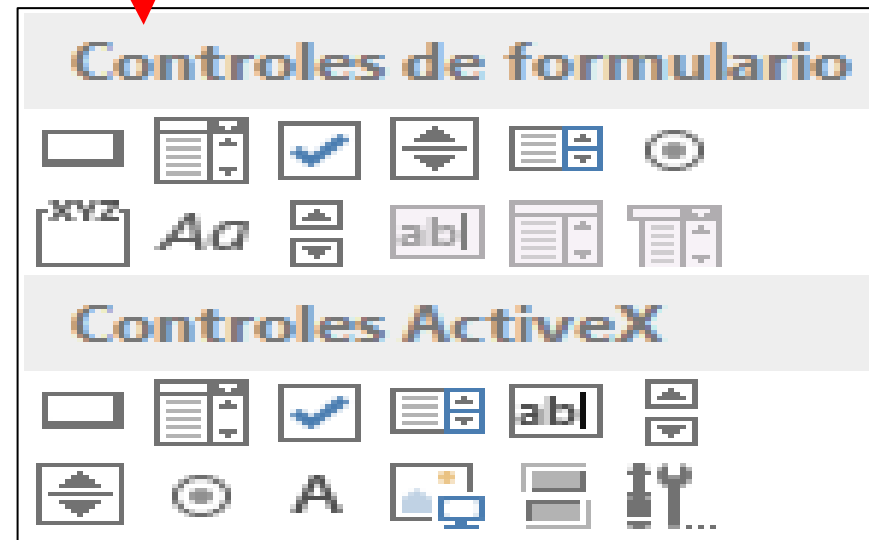
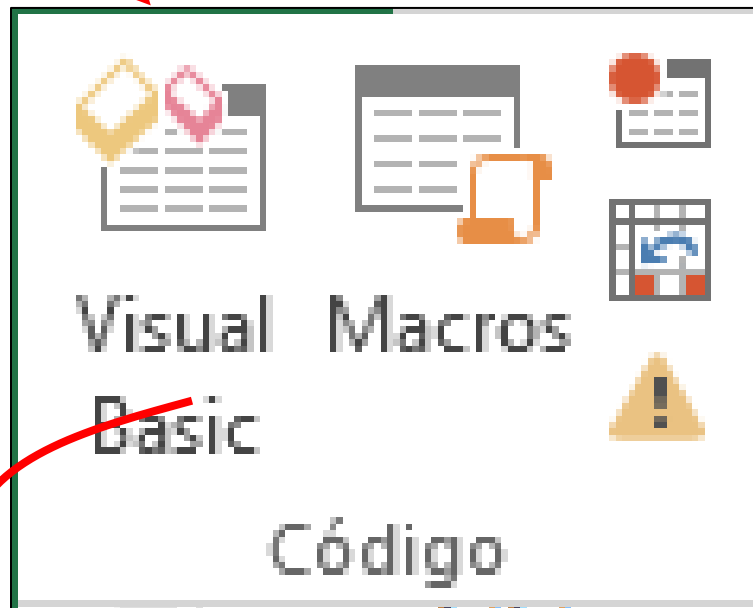
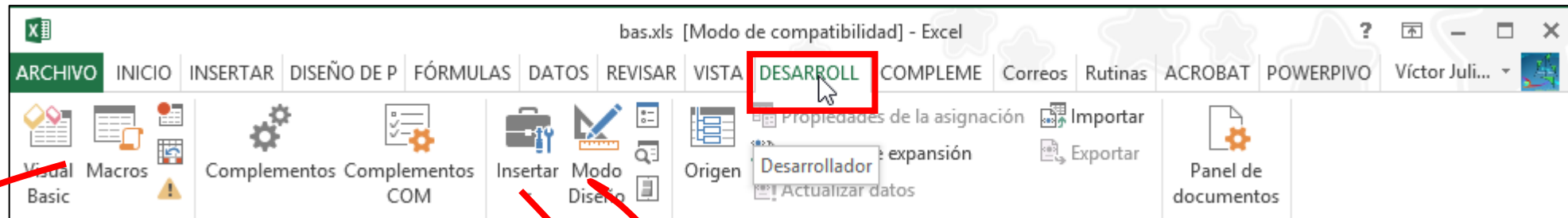
ActiveWorkbook.Names.Add Name:="ensa",

RefersToR1C1:="=temp!R12C5:R16C7"

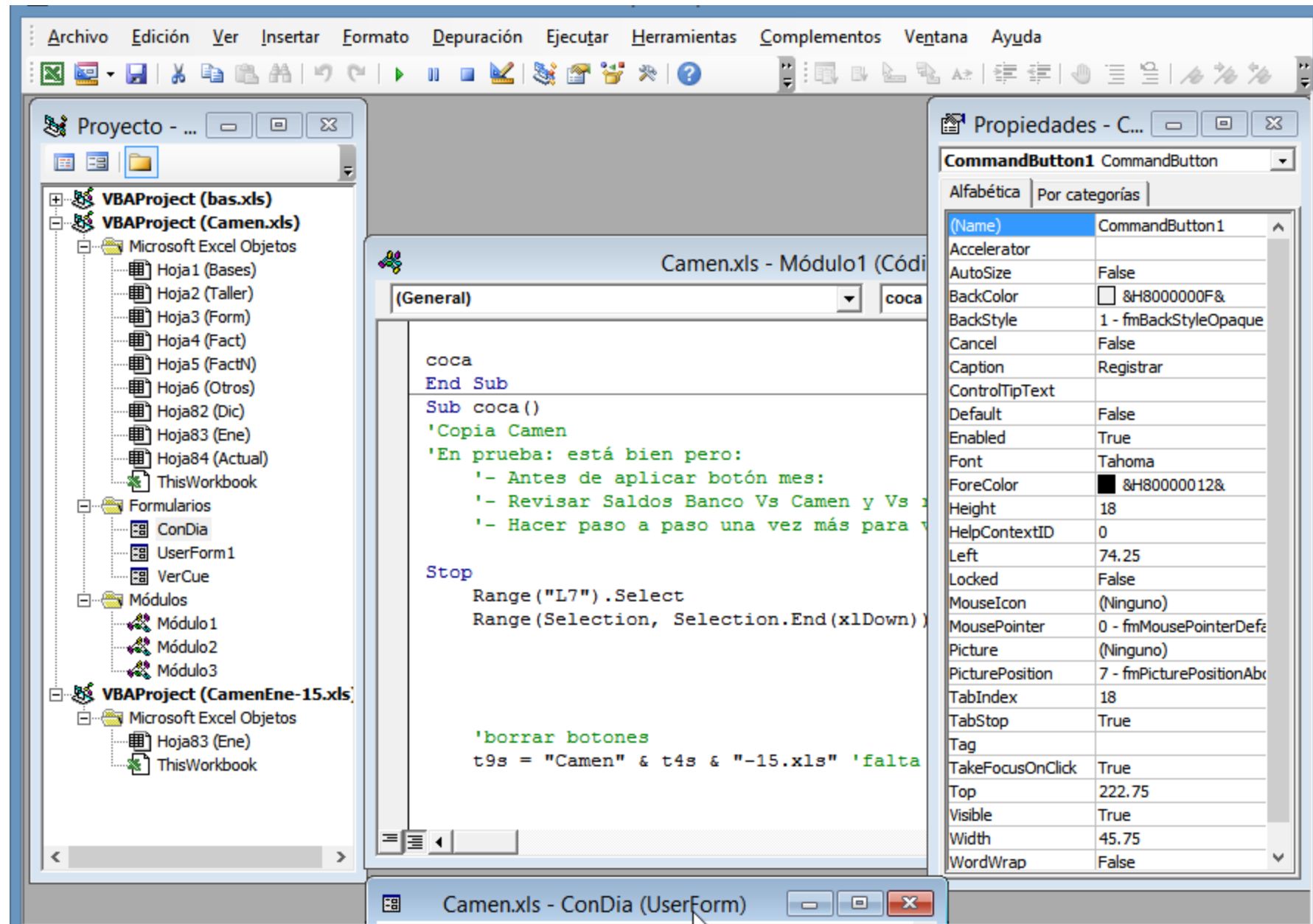
Range("H12").Select

End Sub

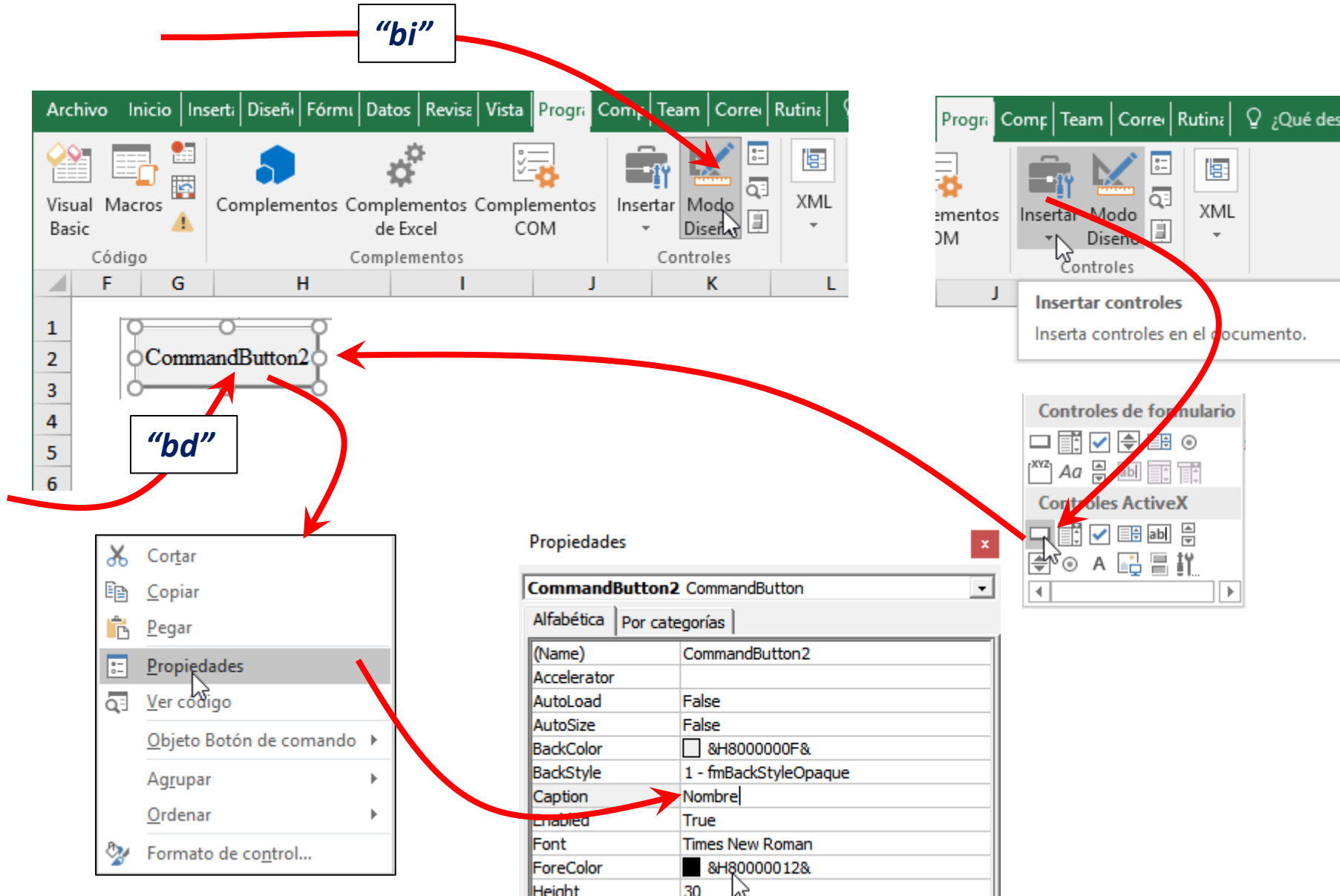
Herramientas... hoja de cálculo/vba



Editor visual basic



Ejemplo de Interacción: *crear botón*



VBA

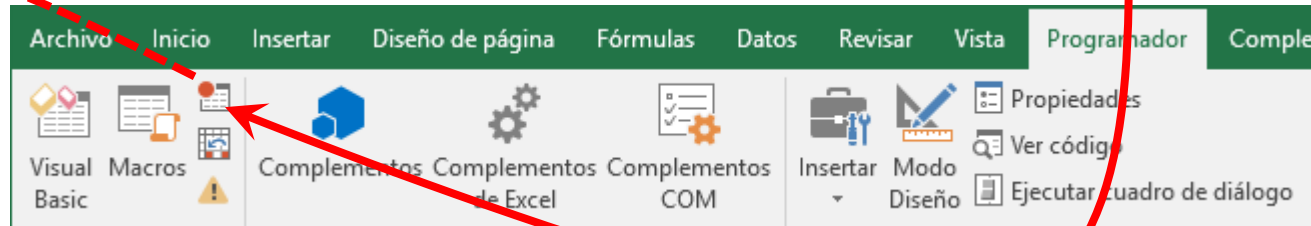
Visual Basic for Applications



DigiGraphic Ltda.
Ingeniería Geográfica

Primera rutina

→ Programador/Desarrollador



Grabar macro

Nombre de la macro:
Macro1

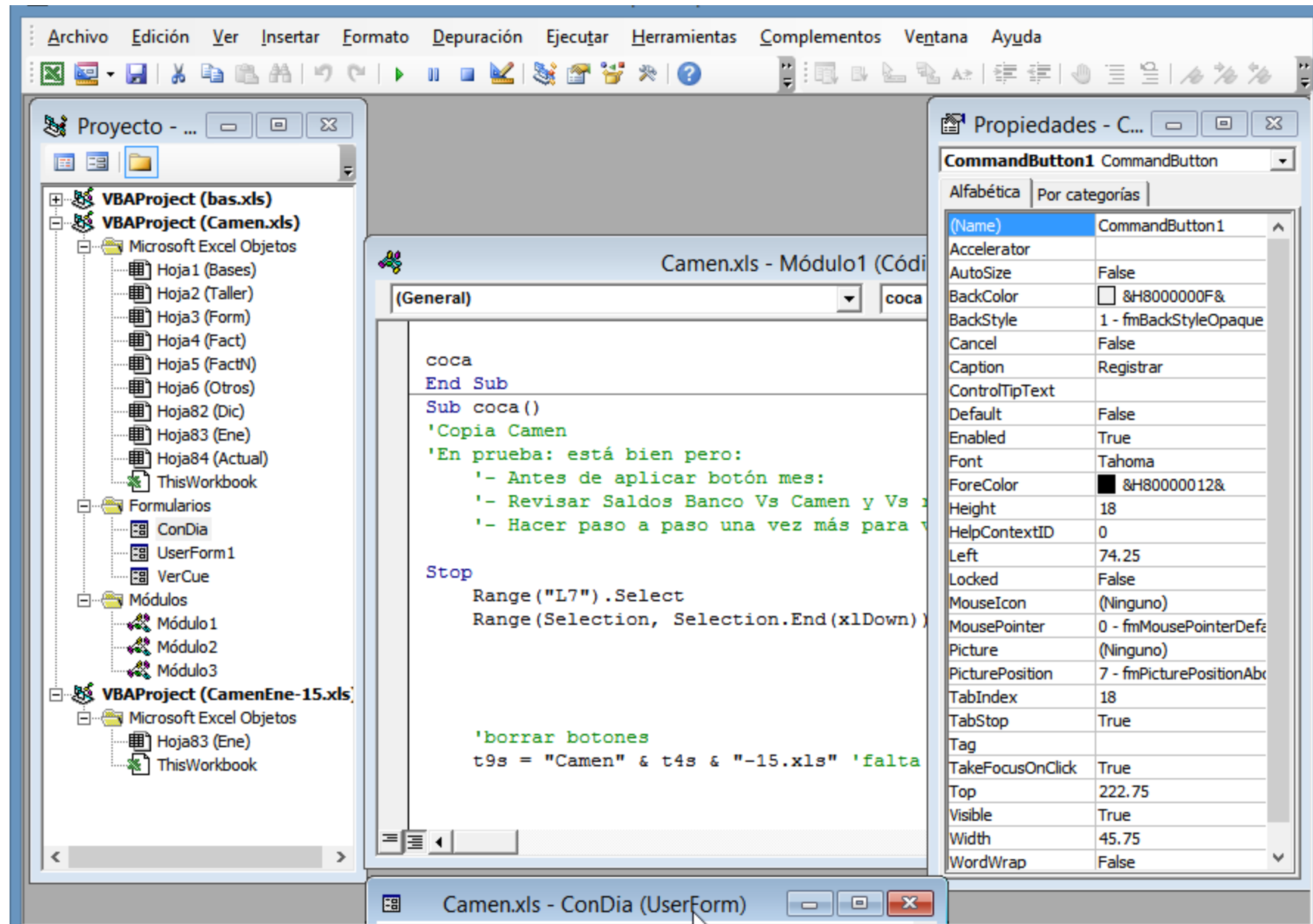
Tecla de método abreviado:
CTRL+

Guardar macro en:
Este libro

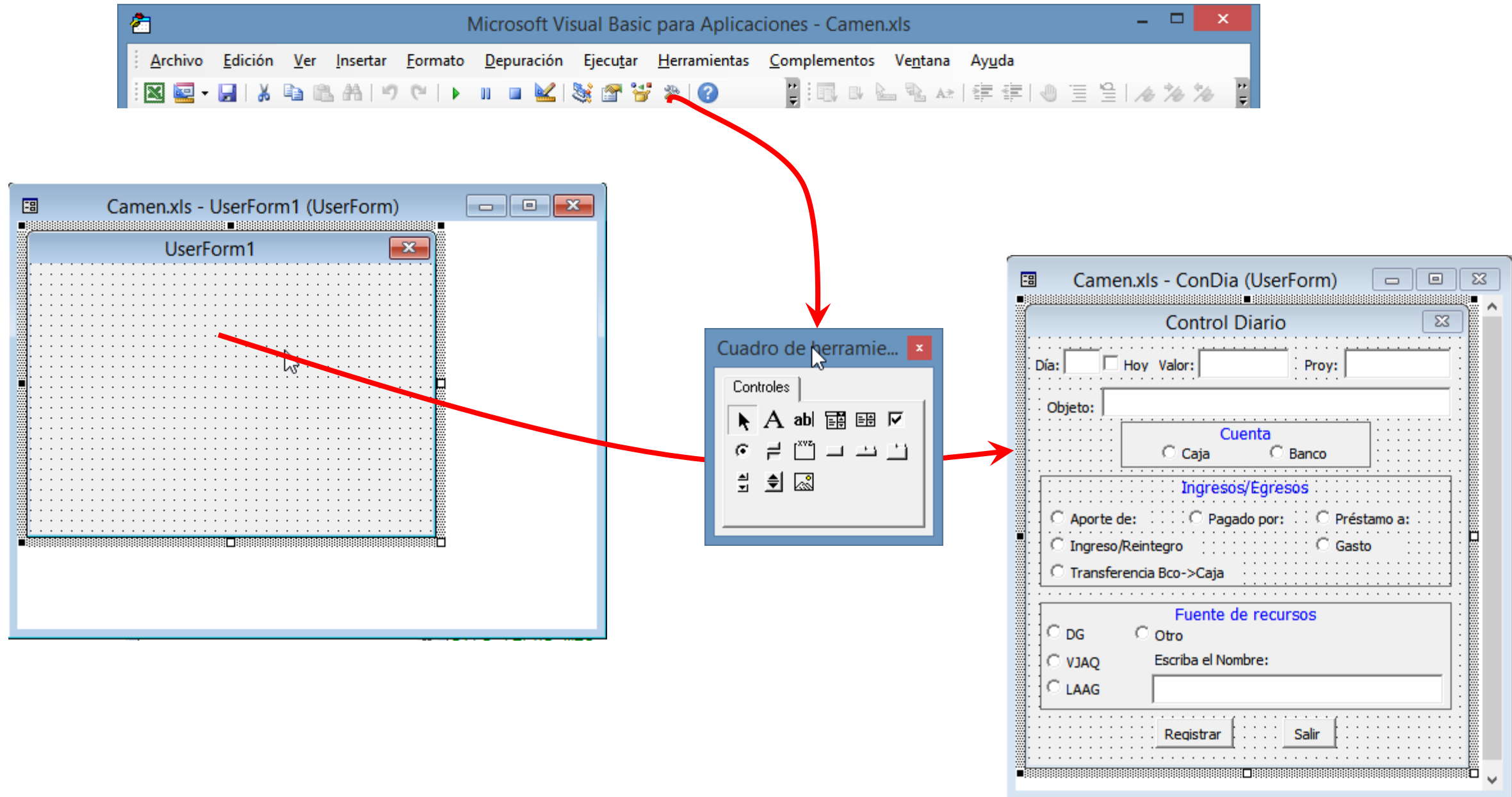
Descripción:

Aceptar Cancelar

Editor visual basic



Visual basic...UserForm



Ejemplo de Interacción: *ejecución*

Estructura *If... else ... End If*

Opciones:

1 : desde código

2 : desde botón

3 : opciones en botón

4 : celdas sensibles

Opción 1 :

[**F5, F8**] en código

Sub demo1()

Sub demo1()

'Interacción

```
temp1 = InputBox(Chr(10) & Chr(10) & Chr(10) & Chr(10) & _  
"Nombre de subdirectorío ? : ", "UDistrital", "Depto1", 0, 0)
```

```
If temp1 = "" Then
```

```
    MsgBox "No insertó nombre"
```

```
Else
```

```
    MsgBox "El subdirectorío será: " & temp1
```

```
End If
```

Stop

End Sub

Opción 2 : Asignar rutina a **botón**

The image illustrates the steps to assign a macro to a button in Microsoft Excel. It is divided into three main sections:

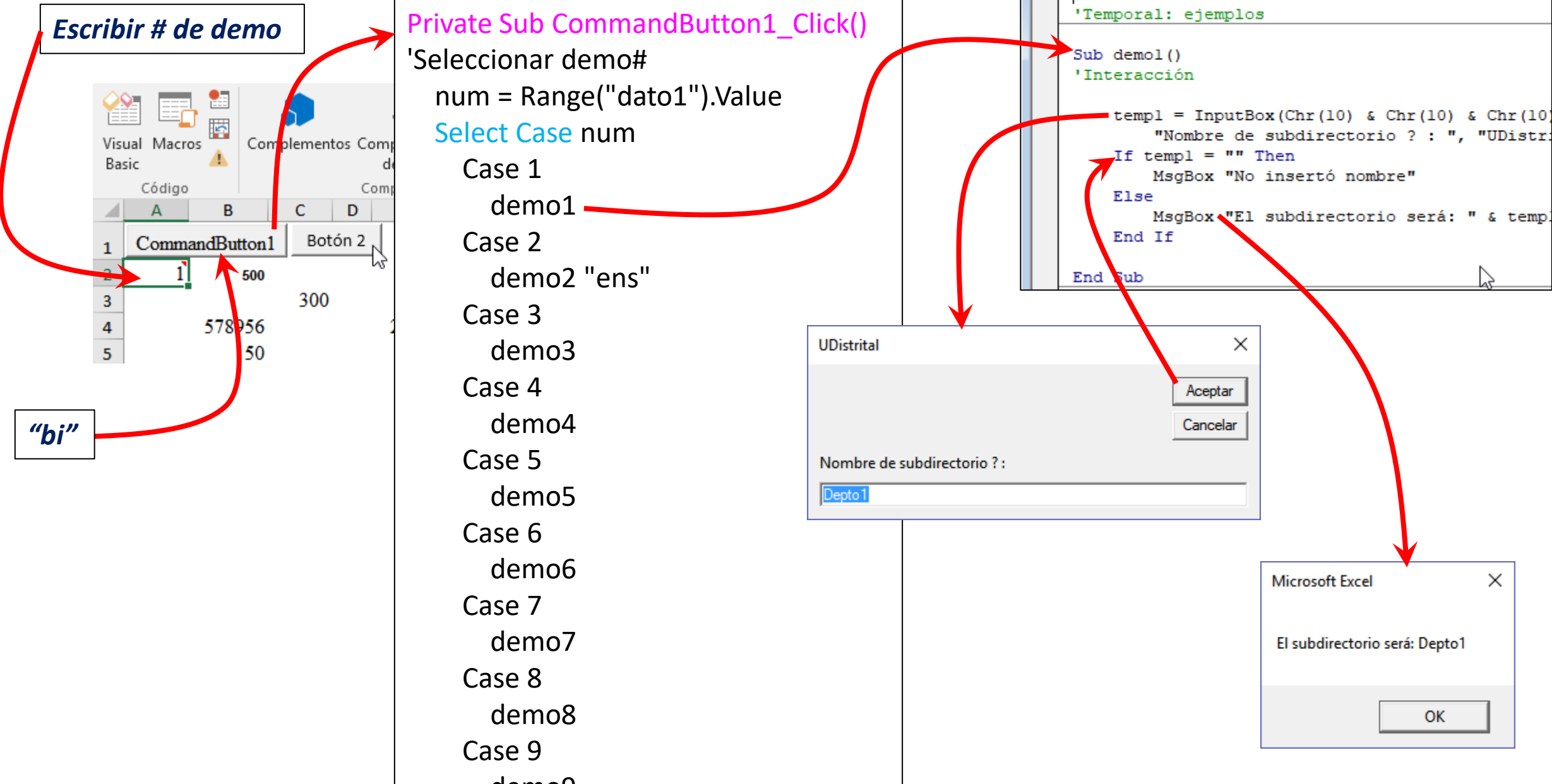
- Top Section:** Shows the Excel ribbon with the 'Complementos' (Add-ins) group selected. A callout box labeled "bd" points to the 'Complementos de Excel' (Excel Add-ins) button.
- Middle Section:** Displays a worksheet with the following data:

	A	B	C	D	E	F	G	H
1	CommandButton1			Botón 2				
2	1	500						
3			300					
4		578956			23456076.34			
5		50			34512.23			
- Bottom Section:** Shows a context menu for 'Botón 2' with the 'Asignar macro...' (Assign Macro...) option highlighted. A red arrow points from this option to the 'Asignar macro' dialog box.

Asignar macro dialog box details:

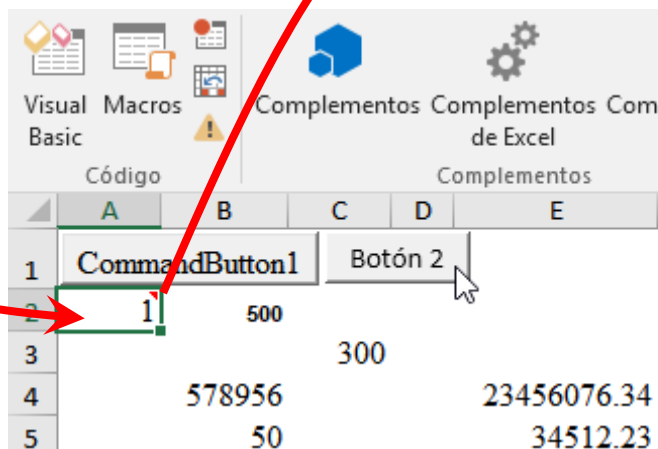
- Nombre de la macro:** A list of macros including demo1, demo10, demo11, demo12, demo13, demo14, demo15, demo16, demo17, demo3, demo4, and demo5. 'demo1' is selected.
- Macros en:** A dropdown menu showing 'Este libro' (This book), which is circled in red.
- Buttons:** 'Modificar' (Modify), 'Grabar...' (Save...), 'Aceptar' (Accept), and 'Cancelar' (Cancel).

Opción 3 : Automatizar asignación de rutinas a **botón**



Opción 4 : Automatizar asignación de rutinas a **celda**

Doble click



```
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)  
'Funcionalidad de celdas
```

```
Cancel = True 'desactiva celda picada
```

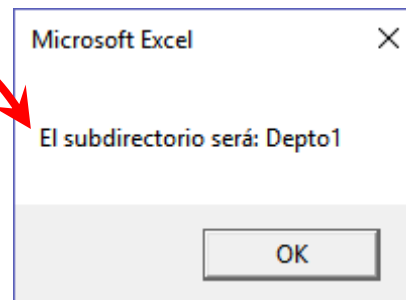
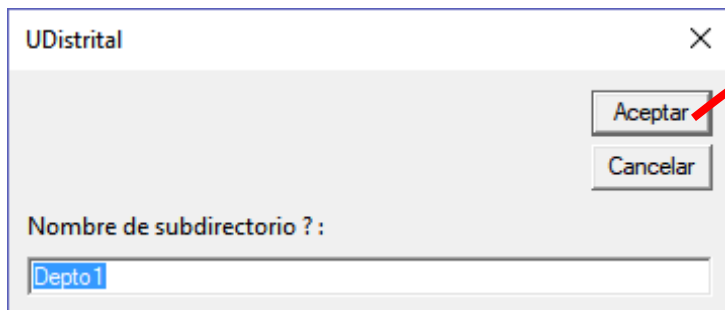
```
temp1 = ActiveCell.Address
```

```
If temp1 = "$A$2" Then
```

```
    CommandButton1_Click
```

```
End If
```

```
End Sub
```



Estructura : If ... End If

Sub demo2()

'Borrar nombre de rango

'tmp = "ens" 'crearlo previamente

If tmp = "" Then

Goto fin

End If

Stop

Application.Goto Reference:=tmp 'iluminar rango tmp

ActiveWorkbook.Names(tmp).Delete

'adicionado para restablecer el rango

ActiveWorkbook.Names.Add Name:="ens", RefersToR1C1:="=pru!R12C5:R16C7"

fin:

End Sub

Estructura : If ... Then... GoSub

Sub demo3()

'Pedir un número para división

Dim Número

Número = InputBox("Escriba un número positivo para dividir entre 2.")

' Sólo se utiliza la subrutina si el usuario escribe un número positivo.

If Número > 0 Then GoSub división 'único caso que no admite Else... End If

If Número < 0 Then

MsgBox "Escribió un número negativo"

End If

MsgBox Número

Exit Sub 'cancela subrutina principal cuando debe terminar.

división:

Número = Número / 2 ' Realiza la división.

Return ' Devuelve control a la línea que
' sigue a la instrucción GoSub.

End Sub

Interacción : Call... y Goto etiqueta

Sub demo4()

'Invocar otra subrutina

Stop

Call demo2("ens") '1ra forma

'demo2 "ens" '2da forma

End Sub

Sub demo2(ByVal tmp As String)

'Borrar nombre de rango

'tmp = "ens" 'crearlo previamente

If tmp = "" Then

Goto fin

End If

Stop

Application.Goto Reference:=tmp 'iluminar rango tmp

ActiveWorkbook.Names(tmp).Delete

'adicionado para restablecer el rango

ActiveWorkbook.Names.Add Name:="ens",

RefersToR1C1:="=pru!R12C5:R16C7"

fin:

End Sub

Estructura : With ... End With

Sub demo5()

'Insertar una fórmula en la **celda E11** y fijarle el tipo de letra.

```
'With Worksheets("Libro1").Worksheets("Hoja1").Cells(11, 5)
```

```
With ActiveSheet.Cells(11, 5) 'fila, columna
```

```
.Formula = "=SQRT(50)"
```

```
With .Font
```

‘with anidado

```
.Name = "Arial"
```

```
.Bold = True
```

```
.Size = 8
```

```
End With
```

```
End With
```

End Sub

Estructuras:

For... Next (anidado)
If..Elseif..else..End If

Sub demo6()

'Genérico para recorrido de celdas en un rango de valores

'Este ejemplo evalúa y colorea valores < 3000 , uno verde y los demás negros

Dim i As Long, j As Long

Range("ensa1").Select

Selection.Font.ColorIndex = xlAutomatic

Selection.ClearContents

Range("C2").Value = 234: Range("B3").Value = 45: Range("C4").Value = 3780

Range("B4").Value = 4598: Range("B6").Value = 25678: Range("C7").Value = 9764

Range("C8").Value = 25686: Range("B5").Value = 367: Range("B7").Value = 4587

Range("B9").Value = 369: Range("C9").Value = 21

Set rango = Range("ensa1")

Stop

For i = 1 To rango.Rows.Count

For j = 1 To rango.Columns.Count

If rango.Cells(i, j).Value < 3000 Then

rango.Cells(i, j).Font.ColorIndex = 3 'rojo

Elseif rango.Cells(i, j).Value = 9764 Then

rango.Cells(i, j).Font.ColorIndex = 4 'verde

Else

rango.Cells(i, j).Font.ColorIndex = 1 'negro

End If

Next j

Next i

End Sub

Estructura: *For [step]... Next*

For CONTADOR ...Step ..

- Step incrementa/disminuye en 2 la variable j cada vez que se ejecuta el bucle
- Cuando el bucle deje de ejecutarse, total representará la suma de 2, 4, 6, 8 y 10.
- No es necesario incluir la variable después de **Next**, pero por comprensión se ha dejado en los 2 ejemplos

Sub demo7()

'Decremento de valores

For j = 10 To 2 Step -2

Total = Total + j

MsgBox "El total es: " & Total

Next j

End Sub

Sub demo8()

'Incremento de valores

For j = 2 To 10 Step 2

Total = Total + j

MsgBox "El total parcial es: " & Total

Next

MsgBox "El total final es: " & Total

End Sub

Estructura: For each ... In ... Next

Sub demo9()

‘Escribir un valor en rangos de celdas

Dim **hoja** As Worksheet

Dim **celda** As Range

For Each **hoja** In ActiveWorkbook.Worksheets

hoja.Select

Range("iv1").Select

For Each **celda** In Range("iv1:iv5")

celda = 25

Next

Stop

Next

End Sub

Sub demo17()

'Borrar datos anteriores

Stop

Sheets(Array("Hojas", "VBA", "temp")).Select

Range("IV1:IV5").Select

Selection.ClearContents

Sheets("Hojas").Select

Range("A1").Select

Sheets("VBA").Select

Range("A1").Select

Sheets("temp").Select

Range("A2").Select

End Sub

Estructura:

For each ... In ...

...

Next

Invocado desde otra
subrutina

```
Sub demo10()  
    Call demo10a("ens", "nro. 8")    '1ra forma  
    'demo10a "ens", "nro. 8"        '2da forma  
End Sub  
  
Sub demo10a(ByVal rn As String, vr As String)  
    'Buscar el valor vr en rango rn  
    Dim cel As Range  
    'Dim rn As Range                'rango dónde buscar  
  
    Set rcol = Range(rn)  
    For Each cel In rcol  
        If cel.Value = vr Then  
            t5s = cel.Address  
            MsgBox "Celda      : " & t5s & Chr(10) & "Contenido: " & vr  
        End If  
    Next  
End Sub
```



Bucle : Do While condición ... Loop

- ✓ Comprobar la **condición antes** de entrar al bucle.
- ✓ Si **miNum** vale **9** en vez de 20, las instrucciones contenidas en el bucle **nunca se ejecutarán**.

Sub demo11()

'Bucle Do While: comprobar 1ro. la condición.

contador = 0

miNum = 20

Do While **miNum > 10**

miNum = miNum - 1

contador = contador + 1

MsgBox "miNum : " & miNum & Chr(10) & "Contador : " & contador

Loop

MsgBox "La operación se efectuó " & contador & " veces."

End Sub

Bucle : Do ... Loop While condición

- ✓ Comprobar la **condición después** de que el bucle se haya ejecutado al menos una vez.
- ✓ Las instrucciones sólo se ejecutarán antes de que la condición llegue a ser False.

Sub demo12()

'Bucle Loop While: comprobar la condición después de ejecutarlo cada vez

contador = 0

miNum = 10 **‘probar y luego cambiar 10 a 12**

Do

miNum = miNum - 1

contador = contador + 1

MsgBox "miNum : " & miNum & Chr(10) & "Contador : " & contador

Loop While miNum > 10

MsgBox "El proceso se realizó " & contador & " veces."

End Sub

Bucle : Do Until condición ... Loop

- ✓ Comprobar la **condición antes** de entrar al bucle.
- ✓ El bucle se ejecutará mientras la **CONDición** sea **False**, o sea hasta cuando **miNum = 10**.

Sub demo13()

'Bucle Do Until: comprobar 1ro. la condición

contador = 0

miNum = 20

Do Until miNum = 10

MsgBox "miNum : " & miNum & Chr(10) & "Contador : " & contador

miNum = miNum - 1

contador = contador + 1

Loop

MsgBox "Se repitió " & contador & " veces."

End Sub

Bucle: Do ... Loop Until condición

- ✓ Comprobar la **condición después** de que el bucle se haya ejecutado al menos una vez.
- ✓ El bucle se ejecutará mientras la CONDición sea **False**, o sea hasta cuando **miNum = 10**.

Sub demo14()

'Bucle Loop Until: comprobar la condición después de ejecutarlo cada vez

contador = 0

miNum = 1

Do

MsgBox "miNum : " & miNum & Chr(10) & "Contador : " & contador

miNum = miNum + 1

contador = contador + 1

Loop Until miNum = 10

MsgBox "Se repitió " & contador & " veces."

End Sub

Bucle : While condición ... Wend

- ✓ Comprobar la **condición antes** de ejecutar el bucle.
- ✓ El bucle se ejecutará si se cumple la **condición**

Sub demo15()

'Bucle While...Wend : evaluación previa de la condición

Dim bingo As Integer

bingo = 0

cont1 = 0

While bingo <> 850

'otra opción: Do While o Do Until

'se ejecuta mientras bingo sea diferente de 850

bingo = Int(1000 * Rnd())

'Range("a1") = bingo

cont1 = cont1 + 1

Wend ' otra opción: Loop con Do previo

MsgBox "Su número " & bingo & " ganó después de " & cont1 & " jugadas !!"

End Sub

Ejemplo : Llamada a una función

Sub demo16()

'Quitar espacios y capturar iniciales

```
t9s = InputBox("Inserte texto con espacios innecesarios:", "Correcciones...",  
               "Esto es un ensayo con muchos espacios.")
```

```
t3s = ""
```

```
palabra = Split(qesp(t9s))
```

'separación de palabras

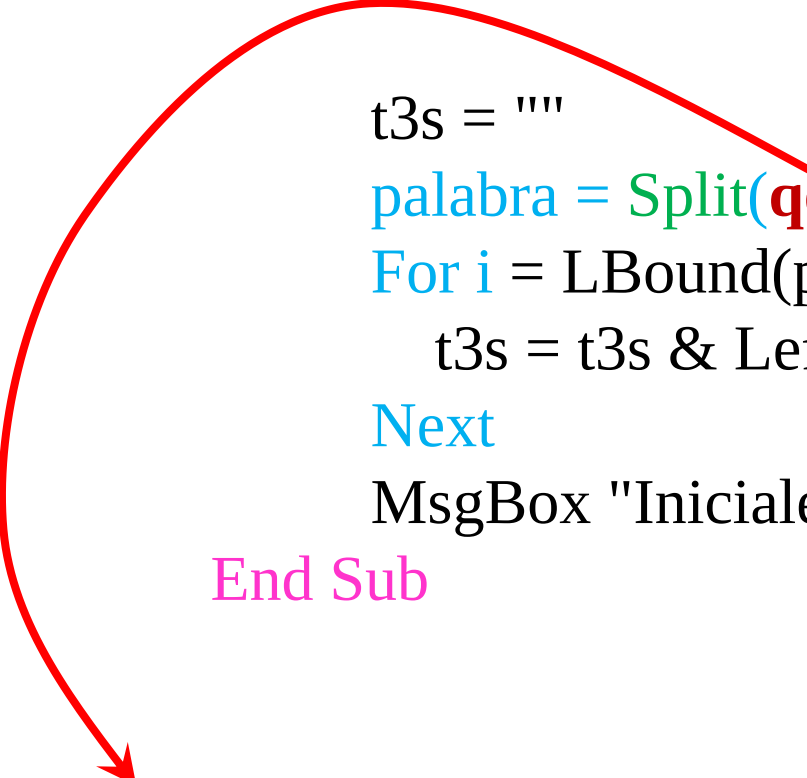
```
For i = LBound(palabra) To UBound(palabra)
```

```
    t3s = t3s & Left(palabra(i), 1)    'acumula inicial de c/palabra
```

```
Next
```

```
MsgBox "Iniciales del texto: " & t3s
```

End Sub



Ejemplo : *Function nombre() ... End Function*

Sub demo16()

'Quitar espacios y capturar iniciales

t9s = InputBox("Inserte texto con espacios
innecesarios:", "Correcciones.", "Esto es un
ensayo con muchos espacios.")

t3s = ""

palabra = Split(qesp(t9s))

For i = LBound(palabra) To UBound(palabra)

t3s = t3s & Left(palabra(i), 1) 'acumula inicial de
c/palabra

Next

MsgBox "Iniciales del texto: " & t3s

End Sub

Function **qesp**(ByVal t9s As String)

'Quitar ESPacios: dejar un solo espacio entre palabras
'ivdo: demo16

Dim t3s As String

t3s = Trim(t9s)

Do While InStr(t3s, " ")

t3s = Replace(t3s, " ", " ")

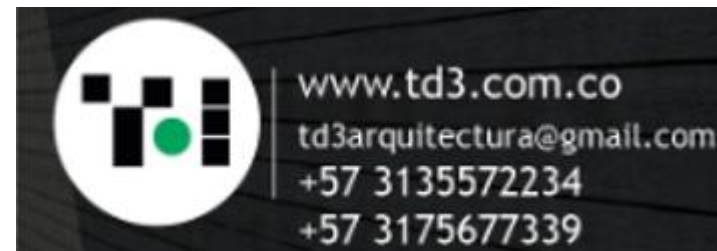
Loop

MsgBox "Texto sin espacios: " & Chr(10) & t3s

qesp = t3s 'necesario para captura iniciales

End Function

GRACIAS...



Experiencia sobre aplicaciones VBA:

<http://dgltda.com/CieTie/IndCti.html>

<http://www.dgltda.com/> → Mapa del sitio → Aplicaciones

Víctor Julio Alvarez Q.

Tel 301 6658

Cel 310 330 6003