

1. Introducción a la programación

Fundamentos de Informática

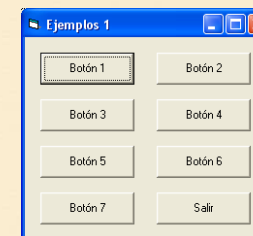
Dpto. Lenguajes y Sistemas Informáticos

Curso 2012 / 2013



Índice

1. Introducción a la programación



1. Ej01: Vacío
2. Ej02: Saludo
3. Ej03: Asignación
4. Ej04: Lectura
5. Ej05: Concatenación
6. Ej06: Conversión
7. Resumen
8. Notación DdF vs VB
9. Ejercicio



2

Introducción a la programación

Ej01: Vacío

1. Ejemplo 01

- **Título**
 - Subprograma vacío (Botón 1)
- **Nombre**
 - cmd1_Click
- **Descripción**
 - Subprograma que no hace nada
- **Observaciones**
 - Comienzo y fin de un subprograma
 - Codificación VB
 - Patrón básico

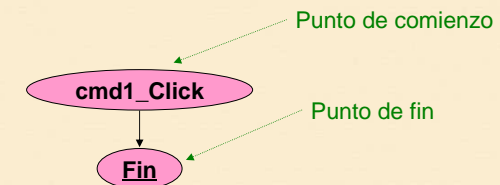


3

Introducción a la programación

Ej01: Vacío

Ej01: Diagrama de Flujo



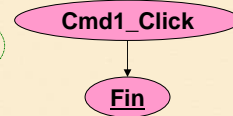
- **Punto de comienzo**
 - Nombre del subprograma
 - Nombre del control (cmd1) + evento (Click)
- **Punto de fin**
 - Único para cada diagrama de flujo
 - Nomenclatura: **Fin**



4

Ej01: Codificación VB

```
Private Sub cmd1_Click()  
End Sub
```



Nos pone automáticamente Private.
No es necesario ponerlo aunque puede quedarse

El entorno VB nos propone el esqueleto del subprograma asociado a hacer clic sobre el botón cmd1 en modo diseño

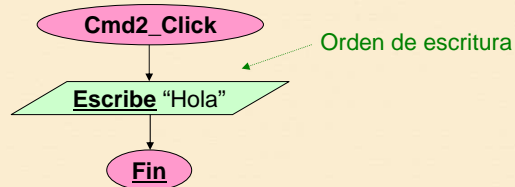


2. Ejemplo 02

- **Título**
 - Saludo (Botón 2)
- **Nombre**
 - cmd2_Click
- **Descripción**
 - Subprograma que saluda
- **Observación**
 - Escritura en pantalla de un texto literal



Ej02: Diagrama de Flujo

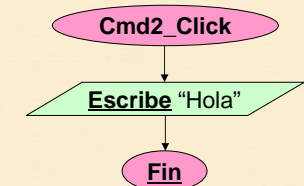


- La orden de escritura vendrá expresada en el DdF como Escribe
- En general no detallaremos todo lo que se escribe en el diagrama de flujo. Para ello puede haber anotaciones aparte junto al diagrama de flujo.



Ej02: Codificación VB

```
Sub cmd2_Click()  
    MsgBox "Hola"  
End Sub
```



- Para escritura en VB utilizaremos la instrucción **MsgBox**

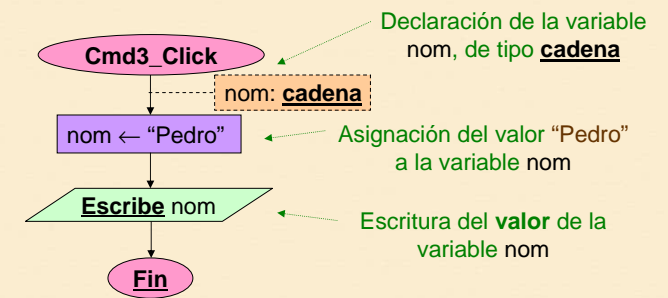


3. Ejemplo 03

- **Título**
 - Asignación (Botón 3)
- **Nombre**
 - cmd3_Click
- **Descripción**
 - Definir una variable de tipo cadena, asignarle un valor cualquiera y mostrarlo en pantalla
- **Observaciones**
 - Declaración de variables
 - Asignación de un valor a una variable
 - Escritura del valor de una variable



Ej03: Diagrama de Flujo



- Hay que declarar todas las variables del programa, indicando de qué *tipo* son, por ejemplo, cadena
- Subrayamos las palabras clave (e.g. cadena, Fin) para diferenciarlos de los nombres inventados (e.g. nom)
- Mediante la *asignación* una variable recibe (←) un valor

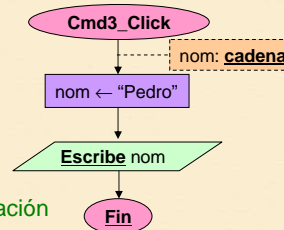


Ej03: Codificación VB

```

Sub Cmd3_Click()
    Dim nom As String
    nom = "Pedro"
    MsgBox nom
End Sub
    
```

Annotations: 'Declaración' points to 'Dim nom As String'; 'Asignación/Inicialización' points to 'nom = "Pedro"'; 'Mostrar valor' points to 'MsgBox nom'.

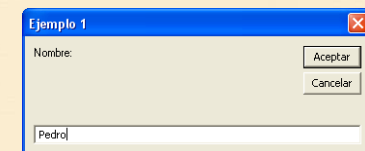


- Para declarar una variable en VB, tras la palabra reservada Dim especificaremos el **nombre**, por ejemplo nom y luego As seguido del **tipo**, por ejemplo String
- La asignación en VB se expresa mediante el símbolo =
- En la **parte izquierda** de la asignación siempre irá una variable y en la **derecha** una expresión que se evaluará
- ¡No hay que **confundir** asignación con igualdad!

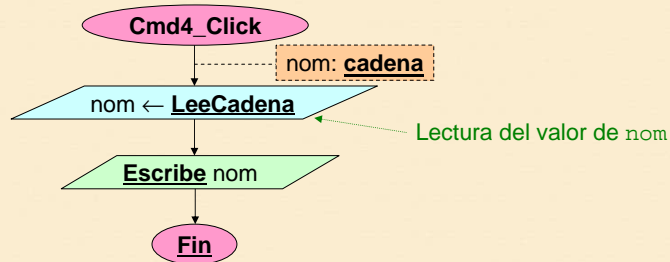


4. Ejemplo 04

- **Título**
 - Lectura (Botón 4)
- **Nombre**
 - cmd4_Click
- **Descripción**
 - Leer un nombre y mostrarlo en pantalla
- **Observaciones**
 - Lectura de un valor



Ej04: Diagrama de Flujo



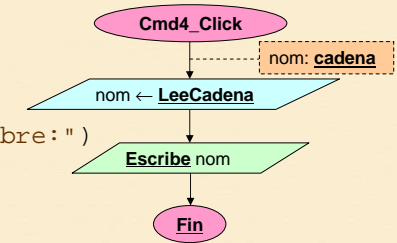
- En vez de asignarle un valor fijo, como en el subprograma Ej03, ahora leemos el valor del teclado
- La orden de lectura de una cadena vendrá expresada como **LeeCadena** en el diagrama de flujo
- Luego escribimos el valor leído en pantalla
- Nótese que no detallamos cómo lo escribimos



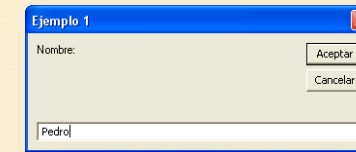
Ej04: Codificación VB

```

Sub Cmd4_Click()
  Dim nom As String
  nom = InputBox ("Nombre: ")
  MsgBox nom
End Sub
  
```



- Para la lectura en VB utilizaremos la instrucción **InputBox**, que nos devuelve siempre una cadena

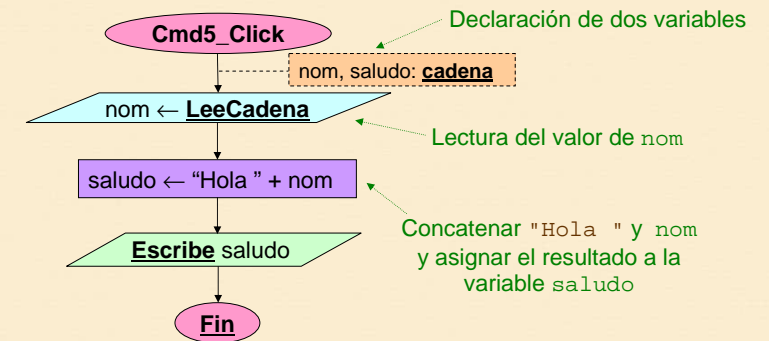


5. Ejemplo 05

- **Título**
 - Concatenación (Botón 5)
- **Nombre**
 - cmd5_Click
- **Descripción**
 - Leer un nombre y obtener una cadena de saludo concatenándole una cadena constante literal
- **Observaciones**
 - Declaración múltiple de variables
 - Expresión: concatenación



Ej05: Diagrama de Flujo



Ej05: Codificación VB

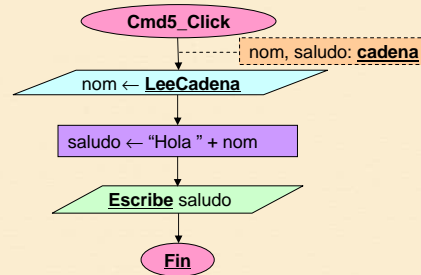
```
Sub cmd5_Click()
    Dim nom As String
    Dim saludo As String
    nom = InputBox("Nombre:")
    saludo = "Hola " & nom
    MsgBox saludo
End Sub
```

Concatenar "Hola " y nom y asignar el resultado a la variable saludo

Declaración múltiple

```
Dim nom As String, saludo As String
```

Pueden declararse varias variables en VB separadas por comas pero hay que volver a especificar el tipo

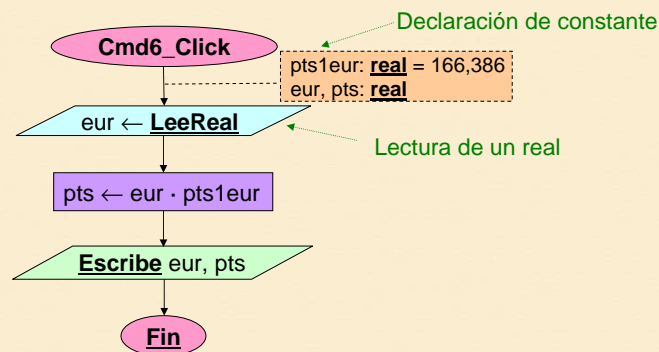


6. Ejemplo 06

- **Título**
 - Conversión (Botón 6)
- **Nombre**
 - cmd6_Click
- **Descripción**
 - Pedir una cantidad en euros, pasarla a pesetas y mostrar el resultado
- **Observaciones**
 - Lectura y escritura de números reales
 - Expresión: multiplicación
 - Constantes



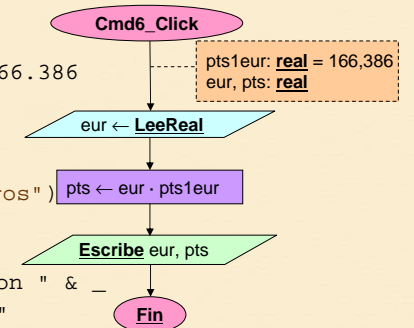
Ej06: Diagrama de Flujo



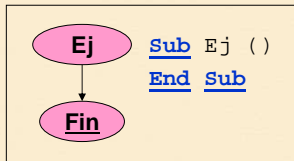
Ej06: Codificación VB

```
Sub cmd6_Click()
    Const pts1Eur As Double = 166.386
    Dim eur As Double
    Dim pts As Double
    Dim s As String
    s = InputBox("Cantidad en euros")
    eur = Cdbl(s)
    pts = eur * pts1Eur
    MsgBox CStr(eur) & " euros son " & _
           CStr(pts) & " pesetas"
End Sub
```

Se utiliza el guión de subrayado para indicar que la instrucción continúa



7. Resumen



```

sum ← x1+x2
sum = x1 + x2
    
```

```

Escribe x
MsgBox "x: " & CStr(x)
    
```

```

FinPrograma
End
    
```

```

n1, n2: entero
sum: real
ptas1Eur: real = 166,386

Dim n1 As Integer, n2 As Integer
Dim sum As Double
Const ptas1Eur As Double = 166.386
    
```

```

nbr ← LeeCadena
n1 ← LeeEntero
d1 ← LeeReal

nbr = InputBox ("Nombre")
s = InputBox ("Nº de elementos")
n1 = CInt (s)
s = InputBox ("Cantidad")
n1 = CDb1 (s)
    
```



8. Notación DdF vs notación VB (I)

Diagrama de flujo	Visual Basic	Comentario
←	=	Asignación
+	+	Suma
-	-	Resta y cambio de signo
.	*	Producto
<u>Div</u>	\	Cociente división entera
<u>Mod</u>	<u>Mod</u>	Resto división entera
$x^{2,53}$	$x^{2.53}$	Potencia
/	/	División real
+	&	Concatenación
<u>entero</u>	<u>Integer</u>	Entero (2 bytes)
<u>real</u>	<u>Double</u>	Real (doble precisión)
<u>cadena</u>	<u>String</u>	Cadena de caracteres
,	.	Coma (punto) decimal



8. Notación DdF vs notación VB (II)

Diagrama de flujo	Visual Basic	Comentario
<u>LeeCadena</u>	InputBox	Lectura de una cadena
<u>LeeEntero</u>	InputBox, <u>CInt</u>	Lectura de un entero
<u>LeeReal</u>	InputBox, <u>CDbl</u>	Lectura de un real
<u>FinPrograma</u>	<u>End</u>	Finalizar programa
<u>Aentero</u>	<u>CInt</u>	Convertir a entero
<u>Areal</u>	<u>CDbl</u>	Convertir a real (doble)
<u>Acadena</u>	<u>CStr</u>	Convertir a cadena
<u>SaltoLinea</u>	<u>vbCrLf</u>	Salto de línea
<u>Escribe</u>	MsgBox	Escritura de un mensaje
	<u>Option Explicit</u>	Forzar declaración de variables



9. Ejercicio

- **Título**
 - Intereses (Botón 7)
- **Nombre**
 - Ejr01_Click
- **Descripción**
 - Diseña y codifica un programa en VB para calcular el **interés** de una determinada **cantidad** a un **tipo anual** dado para un cierto **número de días** suponiendo que el año tiene 365 días exactos. Mostrar tanto el interés como lo que finalmente cobraremos si nos retienen un 18%.

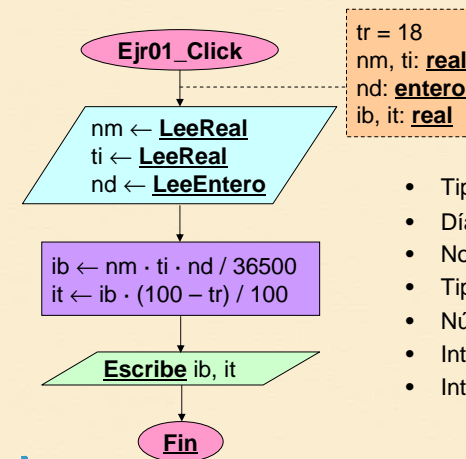


Ejr01: Análisis

- Datos conocidos
 - Días del año = 365
 - Tipo de retención = 18% (tr)
- Datos a leer
 - Nominal (nm)
 - Tipo de interés (ti)
 - Número de días (nd)
- Datos a calcular
 - Interés bruto (ib)
 - Interés neto (it)



Ejr01: Diagrama de flujo



- Tipo de retención = 18% (tr)
- Días del año = 365
- Nominal (nm)
- Tipo de interés (ti)
- Número de días (nd)
- Interés bruto (ib)
- Interés neto (it)



Ejr01: Codificación VB

```

Sub Ejr01_Click ()
    Const tr As Double = 18
    Dim s As String
    Dim nm As Double, ti As Double
    Dim nd As Integer
    Dim ib As Double, it As Double
    s = InputBox ("Nominal: ")
    nm = Cdbl (s)
    s = InputBox ("Tipo de interés:")
    ti = Cdbl (s)
    s = InputBox ("Días:")
    nd = Cint (s)
    ib = nm * ti * nd / 36500
    it = ib * (100 - tr) / 100
    MsgBox "Interés bruto: " & Cstr (ib) & vbCrLf & _
        "Interés neto: " & Cstr (it)
End Sub
    
```

