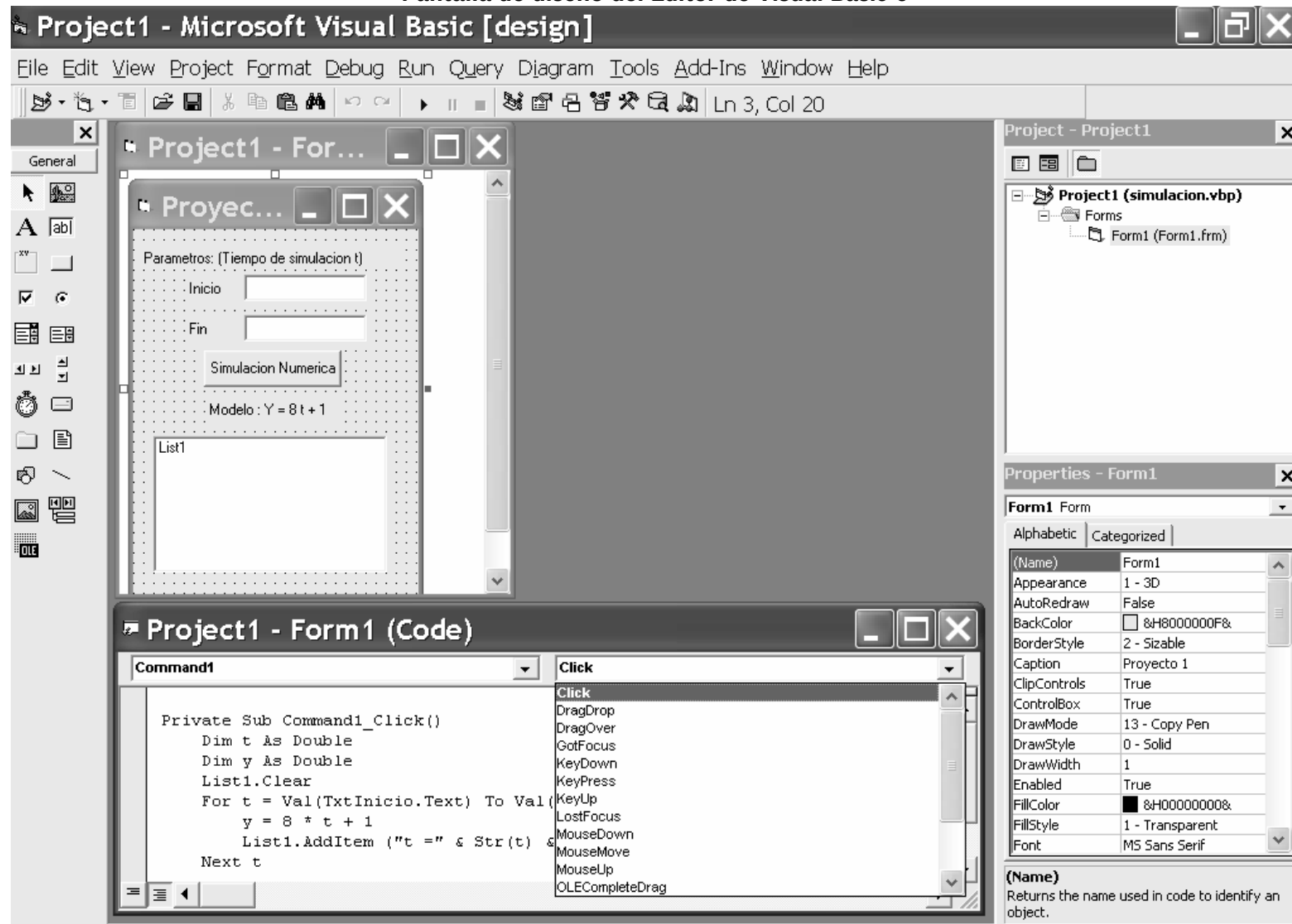


Pantalla de diseño del Editor de Visual Basic 6



NOTAS IMPORTANTES SOBRE PROGRAMACIÓN EN VB 6.0

- ✍ La propiedad "**NAME**" de los objetos (Nombre con el que se hace referencia al objeto) es diferente de la propiedad "**CAPTION**" o "**TEXT**" (Texto que contiene el objeto). La propiedad **NAME** no cambia durante la ejecución del programa. La propiedad **TEXT** o **CAPTION** puede tomar diferentes valores durante la ejecución.

Ejemplo: En un cuadro de texto (TEXTBOX): La propiedad **NAME** puede ser

TxtApellido y la propiedad **TEXT**: **Hernández**

Cambiando el Valor Desde Código: TxtApellido.Text = "LOPEZ"

Tomando su valor a una variable Apellido = TxtApellido.Text

Usando su valor para comparar IF TxtApellido.Text = "LOPEZ" THEN ..

- ✍ La propiedad "**TABINDEX**" permite establecer el orden en que la tecla del tabulador recorre cada objeto. Los valores que puede tomar esta propiedad son de 0 (Primer objeto donde aparece el cursor) hasta el numero total de objetos a recorrer con la tecla TAB. Ejemplos:

TxtNombre.TabIndex = 0

BtnLogin.TabIndex = 1

- ✍ Para hacer referencia al valor de la propiedad de un objeto desde el código del programa, usar el nombre del objeto y la propiedad separados por un punto. Ejemplos:

RFC = TxtRFC.Text

Radio = TxtRadio.Text

TxtAreaExacta.Text = Str(AreaExacta)

Form1.caption =

"Consultas"

PROPIEDADES IMPORTANTES DE LAS FORMAS (FORMS)

- ✍ BACKCOLOR
- ✍ BORDERSTYLE (none, Fixed Single, Sizable, etc)
- ✍ CAPTION
- ✍ CONTROLBOX (True, False)
- ✍ ICON
- ✍ MAXBUTTON (True, False)
- ✍ MINBUTTON (True, False)
- ✍ MOVEABLE (True, False)
- ✍ STARTUP POSITION (manual, centerowner, centerscreen, windows default)
- ✍ VISIBLE (True, False)
- ✍ WINDOWSTATE (normal, minimized, maximized)

PROPIEDADES IMPORTANTES DE LOS BOTONES DE COMANDO (COMMAND BUTTON)

- ✍ CAPTION
- ✍ ENABLED (TRUE, FALSE)
- ✍ FONT
- ✍ DOWN PICTURE
- ✍ PICTURE
- ✍ STYLE (NORMAL, GRAPHICAL)
- ✍ VISIBLE (TRUE, FALSE)

PROPIEDADES IMPORTANTES DE LOS CUADROS DE TEXTO (TEXTBOX)

- ✍ ALIGNMENT (Left, Right, Center)
- ✍ APPEARANCE (3D, Flat)
- ✍ BACKCOLOR
- ✍ BORDERSTYLE (None, Fixed Single)
- ✍ ENABLED
- ✍ FONT
- ✍ LOCKED (True, False)
- ✍ MAXLENGTH (0 - cualquier longitud. Otro numero - Longitud maxima)

- ✍ **MULTILINE** (True, False)
 - ✍ **PASSWORDCHAR**
 - ✍ **SCROLLBARS** (None, Horizontal, Vertical, Both)
 - ✍ **TEXT**
 - ✍ **VISIBLE** (True, False)
 - ✍ **GRABANDO PROYECTOS.-** Al momento de grabar un programa en VB, se debe guardar el proyecto (con extensión .VBP) y todas las formas que se incluyan en él de manera independiente (Con extensión .FRM). Por ejemplo si un proyecto llamado CIRCULO.VBP contiene una forma llamada CIRCULO.FRM , ambos archivos deben guardarse separadamente.
 - ✍ **GENERANDO .EXE.-** Para crear un Archivo . EXE, cargar primero en Visual Basic el programa del que se desea hacer el .EXE, luego, en el menú FILE seleccionar "MAKE projectname.EXE" y elegir la ruta donde se generará el .EXE. El archivo .EXE generado es totalmente independiente del proyecto .VBP y las formas .FRM
 - ✍ **DECLARANDO VARIABLES.-** Una variable es una locación temporal para almacenar datos del programa. Se pueden declarar varias variables al mismo tiempo. Se recomienda utilizar siempre nombres de variables que recuerden la función que tienen en el programa. Ejemplo:

DIM Radio as Integer

DIM X,Y,Z as Double

Algunos tipos válidos son: INTEGER (Entero), LONG (Entero Largo), SINGLE (Simple precisión), DOUBLE (Doble precisión), CURRENCY, STRING, BOOLEAN, DATE y VARIANT
El tipo de Datos Variante es el mas flexible de todos, ya que permite almacenar datos de cualquier tamaño o formato, sin embargo ocupa mas memoria que los otros tipos de datos. Los tipos de Datos VARIANT También pueden declararse sin especificarlo. Ejemplo:

DIM Radio

DIM X,Y,Z
 - ✍ **VAL** .-La función VAL convierte una cadena de caracteres (Texto) a un valor numérico. Es muy útil cuando el usuario introduce datos numéricos en un cuadro de Texto pero internamente necesitamos un número. Ejemplo: Suponer que hay un cuadro de Texto llamado TXTN donde el usuario introduce un valor numerico para N, internamente, no se podrían hacer cálculos matemáticos con el texto que introduzca el usuario (ya que es texto), por esa razón se utiliza VAL de la siguiente manera: N = VAL(TXTN.Text)
Entonces la variable N ya puede ser utilizada para cálculos.
 - ✍ **STR.-** La función STR es inversa a VAL, pues convierte un número a un Texto (cadena de caracteres). Esto es muy útil para mostrar resultados de operaciones matemáticas en cuadros de texto. Ejemplo: Suponga que desea imprimirse la variable AREA = 3.14 * r * r en un cuadro de Texto llamado TXTAREA , Entonces se utiliza la función STR de la siguiente manera: TXTAREA.Text = STR (Area)
 - ✍ **CUANDO USAR FOR ...NEXT O DO..LOOP.-** Ambas instrucciones sirven para repetir ciclos de código. Como regla general, cuando se conoce el número exacto de veces que se repetirá el ciclo, se utiliza FOR...NEXT. Cuando La repetición del ciclo depende del resultado de una condición, se utiliza DO..LOOP.

DECISIONES EN VB

IF condicion THEN

END IF

Ejemplo:

IF X= 0 THEN

Msgbox " X igual a cero "

END IF

IF condicion THEN

...

ELSE

...

END IF

Ejemplo:

IF X= 0 then

Msgbox " X igual a cero "

ELSE

Msgbox " X diferente de Cero "

END IF

SELECT CASE variable

CASE valor1

...

CASE valor2

...

CASE valor 3

...

CASE ELSE

...

END SELECT

SELECT CASE edad

CASE 16

Msgbox " Puedes manejar"

CASE 18

MsgBox "Puedes Votar"

CASE 21

MsgBox "Puedes tomar"

CASE ELSE

MsgBox "Disfruta tu edad"

END SELECT

SELECT CASE variable

CASE valor1 TO valor2

...

CASE valor3 TO valor4

...

CASE ELSE

...

END SELECT

SELECT CASE edad

CASE 0 TO 17

Msgbox " Eres menor de edad"

CASE 18 TO 30

MsgBox "Eres Joven"

CASE 40 TO 99

MsgBox "Eres Adulto"

END SELECT

CICLOS EN VB

```
FOR variable = inicio TO fin
    ....
    (Instrucciones a repetir en el ciclo)
NEXT variable
```

```
FOR X = 1 TO 5
    MsgBox "Repeticion num." & str(X)
NEXT X
```

```
DO WHILE condicion
    ....
    (Instrucciones a repetir en el ciclo)
LOOP
```

```
DO WHILE Nombre <> "FIN"
    Nombre = INPUTBOX("Teclea tu nombre, o FIN para salir")
    IF Nombre <> "FIN" THEN
        MsgBox "Nombre: " & Nombre
    END IF
LOOP
```

```
DO UNTIL condicion
    ....
    (Instrucciones a repetir en el ciclo)
LOOP
```

```
DO UNTIL Nombre = "FIN"
    Nombre = INPUTBOX("Teclea tu nombre, o FIN para salir")
    IF Nombre <> "FIN" THEN
        MsgBox "Nombre: " & Nombre
    END IF
LOOP
```

```
DO
    ....
    (Instrucciones a repetir en el ciclo)
LOOP
```

```
DO
    Nombre = INPUTBOX("Teclea tu nombre, o FIN para salir")
    IF Nombre <> "FIN" THEN
        MsgBox "Nombre: " & Nombre
    ELSE
        EXIT DO
    END IF
LOOP
```