Objectives

Program Purpose

• Design a simple number guessing game
• The computer generates a hidden number between 1 and 25
• The user guesses whether it is even or odd
• A message displays whether the user guessed correctly or not

Learning Goals

• Use of option buttons
• Use of a Boolean variable
• Use of comments in the code
• Use of ‘If_Then_Else_ElseIf_End If’ statements
• Use of a ‘Randomize’ statement

Design Notes

The icon property of the form is used. This displays the small icon in the top-left corner of the form, besides the form’s caption. These ‘ico’ files are usually installed in the ‘following folder: C:\Program Files\Microsoft Visual Studio\Common\Graphics\Icons’.

Another property introduced here is the ‘Startup Position’ of the form. Setting this to 2 (centre screen) saves manually positioning the form.

The initial properties are very important in this project. Pay particular attention to the ‘Enabled’, ‘Visible’ and ‘Value’ properties.

It is not necessary to name a label or frame if they serve no other purpose than displaying a message or to contain option buttons. Similarly, a form is only named if it is addressed in the code.

A variable is used to store information temporarily. A Boolean variable can have only 2 values (True or False). It is ‘declared’ with a DIM statement in the general section of the program.

Interface

The Interface has 2 hidden labels, one of which is made visible (which one depends on the user’s guess). The command button ‘cmdHide’ should have the caption ‘&Show’. This will change during the execution of the program.

Program Example 4  Number Game 1

The object of this game is to guess whether the hidden number is ODD or EVEN. The hidden number is between 1 and 25. Choose Odd or Even and then click COMPARE. Click SHOW to see the Hidden Number. Click RESET to try again.

The object of this game is to guess whether the hidden number is ODD or EVEN. The hidden number is between 1 and 25. Choose Odd or Even and then click COMPARE. Click SHOW to see the Hidden Number. Click RESET to try again.
## Names of Objects

<table>
<thead>
<tr>
<th>Type of Object</th>
<th>Number</th>
<th>Names of Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form</td>
<td>1</td>
<td>Form1</td>
</tr>
<tr>
<td>TextBox</td>
<td>1</td>
<td>txtDisplay</td>
</tr>
<tr>
<td>Command Button</td>
<td>3</td>
<td>cmdCompare, cmdReset, cmdHide</td>
</tr>
<tr>
<td>Frame</td>
<td>1</td>
<td>No name –Frame1</td>
</tr>
<tr>
<td>Option Button</td>
<td>2</td>
<td>optOdd, optEven</td>
</tr>
<tr>
<td>Label</td>
<td>3</td>
<td>lblWin, lblLose, Label1</td>
</tr>
</tbody>
</table>

## Initial Properties of Objects

<table>
<thead>
<tr>
<th>Object</th>
<th>Property</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form</td>
<td>Caption</td>
<td>Number Game 1</td>
</tr>
<tr>
<td></td>
<td>Startup Position</td>
<td>2-Center Screen</td>
</tr>
<tr>
<td>txtDisplay</td>
<td>Text</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td>Icon</td>
<td>Browse for an &quot;ico&quot; file: see Design Notes above</td>
</tr>
<tr>
<td></td>
<td>Visible</td>
<td>False</td>
</tr>
<tr>
<td>cmdCompare</td>
<td>Caption</td>
<td>&amp;Compare</td>
</tr>
<tr>
<td>cmdReset</td>
<td>Caption</td>
<td>&amp;Reset</td>
</tr>
<tr>
<td>cmdHide</td>
<td>Caption</td>
<td>&amp;Show</td>
</tr>
<tr>
<td></td>
<td>Enabled</td>
<td>False</td>
</tr>
<tr>
<td>optEven</td>
<td>Caption</td>
<td>&amp;Even</td>
</tr>
<tr>
<td>optOdd</td>
<td>Caption</td>
<td>&amp;Odd</td>
</tr>
<tr>
<td></td>
<td>Value</td>
<td>True</td>
</tr>
<tr>
<td>Label1</td>
<td>Caption</td>
<td>The object of this game is to guess whether the hidden number is ODD or EVEN. The hidden number is between 1 and 25. Choose Odd or Even and then click COMPARE. Click SHOW to see the Hidden Number. Click RESET to try again</td>
</tr>
<tr>
<td>lblWin</td>
<td>Caption</td>
<td>WIN!!</td>
</tr>
<tr>
<td></td>
<td>BackColor</td>
<td>GREEN</td>
</tr>
<tr>
<td></td>
<td>Font</td>
<td>18 Bold</td>
</tr>
<tr>
<td></td>
<td>Visible</td>
<td>False</td>
</tr>
<tr>
<td>lblLose</td>
<td>Caption</td>
<td>LOSE!!</td>
</tr>
<tr>
<td></td>
<td>BackColor</td>
<td>RED</td>
</tr>
<tr>
<td></td>
<td>Font</td>
<td>18 Bold</td>
</tr>
<tr>
<td></td>
<td>Visible</td>
<td>False</td>
</tr>
<tr>
<td>Frame1</td>
<td>Caption</td>
<td>Odd or Even?</td>
</tr>
</tbody>
</table>
Events – Code

Private Sub cmdCompare_Click()
'work out whether or not the number is even
'MOD gives the remainder after division
'VAL converts text to a number
'The next line declares a boolean variable
'This stores a value of true or false

Dim Even As Boolean

'first determine if the number is even or not
If Val(txtDisplay.Text) Mod 2 = 0 Then 'its even
    Even = True
Else
    Even = False
End If

'now compare the User's Guess with ..
'the value of the variable Even
If Even = True And optEven.Value = True Then
    'we picked even and it was even
    lblWin.Visible = True
ElseIf Even = False And optOdd.Value = True Then
    'we picked odd and it was odd
    lblWin.Visible = True
Else
    'we picked wrong
    lblLose.Visible = True
End If

'disable compare button so it can't be clicked until
'reset
cmdCompare.Enabled = False
End Sub

Private Sub cmdHide_Click()
'if the text box is displayed it hides it
'otherwise it shows it

If txtDisplay.Visible = True Then 'Hide it!
    cmdHide.Caption = "&Show"
    txtDisplay.Visible = False
Else
    'Show it
    cmdHide.Caption = "&Hide"
    txtDisplay.Visible = True
End If

End Sub

Private Sub cmdReset_Click()
'reset all the buttons and boxes

txtDisplay.Visible = False
    cmdHide.Caption = "&Show"
    lblWin.Visible = False
    lblLose.Visible = False
    cmdCompare.Enabled = True
    cmdHide.Enabled = False

'generate a new random number
'randomly generates a number between 1 and 25
'and displays it in the textbox
Randomize
    txtDisplay.Text = Int(Rnd * 25 + 1)

End Sub

Private Sub Form_Load()
Randomize
    txtDisplay.Text = Int(Rnd * 25 + 1)

End Sub

Further Design Notes

- Comments are used to display information for the programmer. The use of an apostrophe at the start of the line indicates it is a comment and not code. It will be ignored at runtime.
- The MOD operator gives a remainder. This is useful for checking if a number is even or not.
- The VAL function converts text to a number. Its use prevents odd results where strings are converted into their underlying ASCII value.
- Mistakes are easily made if variables are spelt or typed differently. To prevent this, the programmer should check ‘Require Variable Declaration’ under the ‘Tools’ – ‘Options’ menu.