

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 radio 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 text. These 'ico' files are usually installed in the following folder:
C:\Program Files\Microsoft Visual Studio .NET 2003\Common7\Graphics\Icons'

Another property introduced here is the 'Start Position' of the form. Setting this to 'CenterScreen' saves manually positioning the form.

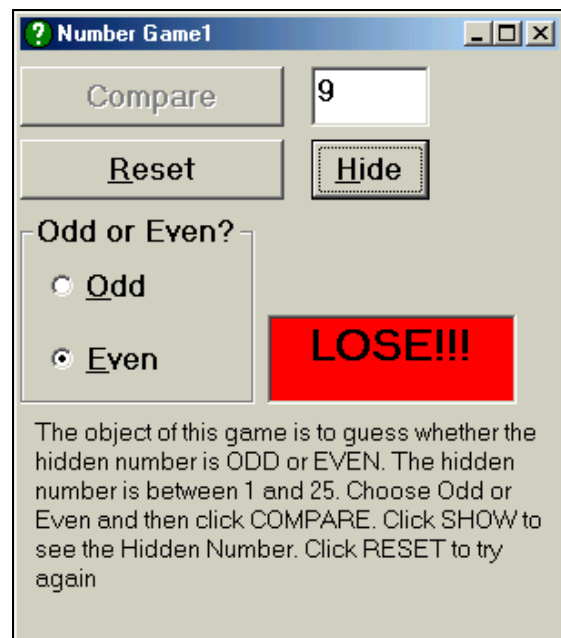
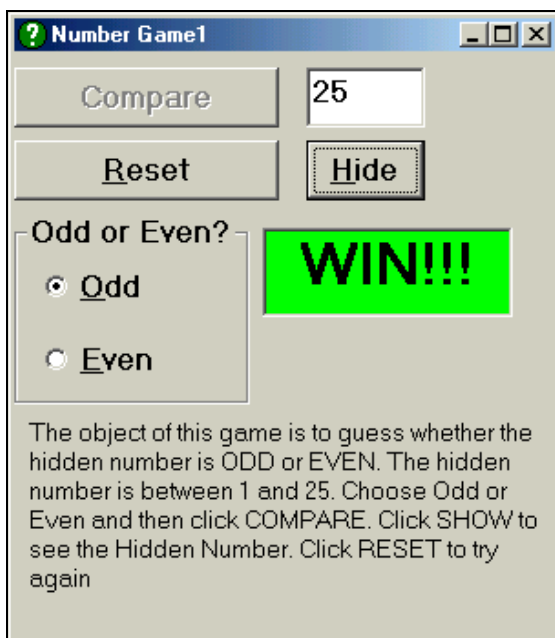
The initial properties are very important in this project. Pay particular attention to the 'Enabled', 'Visible' and 'Value' properties.

It is not necessary to name or label a groupbox if they serve no other purpose than displaying a message or to contain radio 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 button 'btnHide' should also display the text '&Show'. This will change during the execution of the program.



Names of Objects

Type of Object	Number	Names of Objects
Form	1	Form1
TextBox	1	txtDisplay
Button	3	btnCompare, btnReset, btnHide
Groupbox	1	No name – Groupbox1
Radio Button	2	RdbOdd, rdbEven
Label	3	LblWin, lblLose, Label1

Initial Properties of Objects

Object	Property	Initial Value
Form	Text	Number Game 1
	Start Position	Center Screen
	Icon	Browse for an "ico" file: see Design Notes above
txtDisplay	Text	""
	Visible	False
btnCompare	Text	&Compare
	Text	&Reset
	Text	&Show
btnHide	Enabled	False
rdbEven	Text	&Even
rdbOdd	Text	&Odd
	Value	True
Label1	Text	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.
	Text	WIN!!
	BackColor	GREEN
	Font	18 Bold
	Text Align	Middle Center
	Visible	False
	Text	LOSE!!
	BackColor	RED
Font	18 Bold	
lblLose	Text Align	Middle Center
	Visible	False
Groupbox1	Text	Odd or Even?

Events – Code

Private Sub Form1_Load(ByVal sender As ...)

*'generate a new random number between 1 and 25
'and display it in the textbox*

```
Randomize()  
txtDisplay.Text = Int(Rnd() * 25 + 1)
```

End Sub

Private Sub btnHide_Click(ByVal sender As ...)

*'if the text box is displayed it hides it
'otherwise it shows it*

```
If txtDisplay.Visible = True Then  'Hide it  
    btnHide.Text = "&Show"  
    txtDisplay.Visible = False  
Else  
    btnHide.Text = "&Hide"          'Show it  
    txtDisplay.Visible = True  
End If
```

End Sub

Private Sub btnCompare_Click(ByVal sender As ...)

*'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 rdbEven.Checked = True Then
```

```
    'we picked even and it was even  
    lblWin.Visible = True
```

```
ElseIf Even = False And rdbOdd.Checked = True Then
```

```
    'we picked odd and it was odd  
    lblWin.Visible = True
```

```
Else
```

```
    'we picked wrongly  
    lblLose.Visible = True
```

```
End If
```

'disable the compare button so it can't be clicked until restart

```
btnCompare.Enabled = False  
btnHide.Enabled = True
```

End Sub

Events – Code (continued)

Private Sub btnReset_Click(ByVal sender As ...)

'reset all the buttons and boxes

```
txtDisplay.Visible = False  
btnHide.Text = "&Show"  
lblWin.Visible = False  
lblLose.Visible = False  
btnCompare.Enabled = True  
btnHide.Enabled = False
```

*'generate a new random number between 1 and 25
'and displays it in the textbox*

```
Randomize()  
txtDisplay.Text = Int(Rnd() * 25 + 1)
```

End Sub

Further Design Notes

- Comments are used to display information for the programmer e.g.

'work out whether or not the number is even.

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 'Option Strict' under the 'Tools' – 'Options' – 'Projects' menu. This forces variable declaration.
- To save a copy of a solution involves saving three files: the solution file (.sln), the project file (.vbproj) and the form file (.vb).