* **Loop Statements**

A loop statement allows us to execute a statement or group of statements multiple times. VB.Net provides following types of loops to handle looping requirements.

|  |  |
| --- | --- |
| **Loop Type** | **Description** |
| [Do Loop](https://www.tutorialspoint.com/vb.net/vb.net_do_loops.htm) | It repeats the enclosed block of statements while a Boolean condition is True or until the condition becomes True. It could be terminated at any time with the Exit Do statement. |
| [For...Next](https://www.tutorialspoint.com/vb.net/vb.net_fornext_loops.htm) | It repeats a group of statements a specified number of times and a loop index counts the number of loop iterations as the loop executes. |
| [While... End While](https://www.tutorialspoint.com/vb.net/vb.net_while_loops.htm) | It executes a series of statements as long as a given condition is True. |
| [Nested loops](https://www.tutorialspoint.com/vb.net/vb.net_nested_loops.htm) | You can use one or more loops inside any another While, For or Do loop. |

1. **For .. Next Loop**

It repeats a group of statements a specified number of times and a loop index counts the number of loop iterations as the loop executes. It could be terminated at any time with the Exit For statement.

The syntax for this loop construct is −

 **For counter [ As datatype ] = start To end [ Step step ]**

 **[ statements ]**

 **Next [ counter ]**

##  *Example*: Design the following figure then write appropriate codes using VB2010

Exit

Form 2

While .. End While

Do .. Loop

For .. Next

 Form

 Label

4 Buttons

Public Class Form2

Private Sub Button1\_Click(… …)

 Label1.text = “ “

 For i As Integer = 1 To 21 Step 2

 Label1.Text = Label1.Text & i & Space(3)

 Next

 End Sub

End Class

When the above code is compiled and executed, it produces the following result –

 1 3 5 7 9 11 13 15 17 19 21

 **Using Exit For**

 Public Class Form2

 Private Sub Button1\_Click( )

 Label1.Text = ""

 For i As Integer = 1 To 21 Step 2

 Label1.Text = Label1.Text & i & Space(3)

 If i >= 15 Then

 Exit For

 End If

 Next

 End Sub

 End Class

1. **Do .. Loop Statement**

It repeats the enclosed block of statements while a boolean condition is True or until the condition becomes True. It could be terminated at any time with the Exit Do statement.

The syntax for this loop construct is −

 Do { While | Until } condition

 [ statements ]

 Loop

 **OR**

 Do

 [ statements ]

 Loop { While | Until } condition

*Example*: Referencing to the For .. Next example. Do the following modification.

Private Sub Button2\_Click()

 Dim i As Integer

 i = 1

 Label1.Text = ""

 Do While i <= 21

 Label1.Text = Label1.Text & i & Space(3)

 i = i + 2

 Loop

End Sub

Private Sub Button2\_Click()

 Dim i As Integer

 i = 1

 Label1.Text = ""

 **Do Until i > 21**

 Label1.Text = Label1.Text & i & Space(3)

 i = i + 2

 Loop

End Sub

Private Sub Button2\_Click()

 Dim i As Integer

 i = 1

 Label1.Text = ""

 Do

 Label1.Text = Label1.Text & i & Space(3)

 i = i + 2

 Loop Until i > 21

End Sub

Private Sub Button2\_Click()

 Dim i As Integer

 i = 1

 Label1.Text = ""

 Do

 Label1.Text = Label1.Text & i & Space(3)

 i = i + 2

 Loop While i <= 21

End Sub

**Using Exit Do**

 Public Class Form2

 Private Sub Button2\_Click( )

 Dim i As Integer

 i = 1

 Label1.Text = ""

 Do While i <= 21

 Label1.Text = Label1.Text & i & Space(3)

 i = i + 2

 If i >= 15 Then

 Exit Do

 End If

 Loop

 End Sub

 End Class

**H.W:** What the expected results for the above code?

1. **While .. End While**

It executes a series of statements as long as a given condition is True.

The syntax for this loop construct is −

 While condition

 [ statements ]

 End While

Here, statement(s) may be a single statement or a block of statements. The condition may be any expression. The loop iterates while the condition is true.

When the condition becomes false, program control passes to the line immediately following the loop.

**Note:** Here, key point of the *While* loop is that the loop might not ever run. When the condition is tested and the result is false, the loop body will be skipped and the first statement after the while loop will be executed.

**H.W:** Rewrite one of the above codes related to Do..Loop statement using While statement instead.