Multidimensional Array Elements

In visual basic, we can access the values of multidimensional arrays by using a row index and column index values.

Following is the example of accessing the elements of multidimensional arrays in visual basic programming language based on our requirements.

' Two Dimensional Array

Dim arr2D As Integer(,) = New Integer(2, 1) {{4, 6}, {5, 0}, {3, 1}}



' Three Dimensional Array

Dim array3D As Integer(,,) = New Integer(1, 1, 2) {{{1, 2, 3}, {4, 5, 6}}, {{7, 8, 9}, {10, 11, 12}}}

Console.WriteLine(arr2D(1, 0)) ' 5

Console.WriteLine(array3D(1, 1, 1)) ' 11

Visual Basic Multidimensional Array Example



Following is the example of using multidimensional arrays in visual basic programming language to represent the elements in an array with multiple dimensions.

' Two Dimensional Array

Dim array2D As Integer(,) = New Integer(2, 1) {{4, 5}, {5, 0}, {3, 1}}

' Three Dimensional Array

Dim array3D As Integer(,,) = New Integer(1, 1, 2) {{{1, 2, 3}, {4, 5, 6}}, {{7, 8, 9}, {10, 11, 12}}}

Console.WriteLine("---Two Dimensional Array Elements---")

For i As Integer = 0 To 3 - 1

For j As Integer = 0 To 2 - 1

Console.WriteLine("a[{0},{1}] = {2}", i, j, array2D(i, j))



Next

Next

Console.WriteLine("---Three Dimensional Array Elements---")

For i As Integer = 0 To 2 - 1



For j As Integer = 0 To 2 - 1



For k As Integer = 0 To 3 - 1

Console.WriteLine("a[{0},{1},{2}] = {3}", i, j, k, array3D(i, j, k))



Next

Next

Next

Console.WriteLine("Press Enter Key to Exit..")

Console.ReadLine()

**Working with String Arrays**

Arrays are lists of data that have a single data type.

Dim s As String

Dim s(9) As String

S(9)=”Raghava”

MessageBox.Show(s(9))

**Defining and using a Simple Array**

'define an array to hold friends in..

Dim friends(5) As String

' store the names of each friend...

friends(0) = "Sasi"

friends(1) = "Sadhana"

friends(2) = "Hyndavi"

friends(3) = "Joshnika"

friends(4) = "Mokshitha"

friends(5) = "Bhavanya"

‘add Sasi to the list…

ListBox1.Items.Add(friends(0))

Set list box integralheight property false. This tells the list box not to resize.

**Using For Each…Next**

'define an array to hold friends in..

Dim friends(5) As String

' store the names of each friend...

friends(0) = "Sasi"

friends(1) = "Sadhana"

friends(2) = "Hyndavi"

friends(3) = "Joshnika"

friends(4) = "Mokshitha"

friends(5) = "Bhavanya"

'add friends to the list

Dim friendName As String



For Each friendName In friends



ListBox1.Items.Add(friendName)



Next

**Initializing arrays with values**

It is possible to create an array in Visual Basic .NET and populate it in one line of code, rather than having to write specific lines to populate each item as we have here

'define an array to hold friends in....

Dim friends() As String = {"Sasi", "Sadhana", "Hyndavi", "Joshnika", "Mokshitha", "Bhavanya"}



'show the friends

For num As Integer = 0 To friends.Length - 1



ListBox1.Items.Add(friends(num))

Next

**Example**

Dim months(11) As String

months(0) = "Jan"

months(1) = "Feb"

months(2) = "Mar"

months(3) = "Apr"

For i As Integer = 0 To 3

Console.WriteLine("(" & i + 1 & ")" & months(i))

Next

**Example**

Dim months() As String = {"Jan", "Feb"}

'Console.WriteLine(months(0))

For i As Integer = 0 To 1

'Console.WriteLine(i)

Console.WriteLine("(" & i + 1 & ")" & months(i))

Next

**Example**

Dim nums(9) As Integer

Console.WriteLine("Enter numbers")

For i = 0 To 9

nums(i) = Console.ReadLine()

Next

Array.Sort(nums)

Console.WriteLine("Numbers Sorted")



For Each element As Integer In nums

Console.WriteLine(element)

Next

**Example**

Dim names(1) As String

Console.WriteLine("Enter names")

For name As Integer = 0 To names.Length - 1

names(name) = Console.ReadLine()

Next

Array.Sort(names)

Console.WriteLine(" Sorted Names are: “)

For name As Integer = 0 To names.Length - 1

Console.WriteLine(names(name))

Next