This is a structure created by the programmer. I put it in a module.
Now I am using the structure to define the array.
```vbc
Public Class frmSoup
    Inherits System.Windows.Forms.Form

    Dim itemArray As itemInfo

    Private Sub frmSoupCreate_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
        itemArray.itemNo(0) = 3
        itemArray.itemNo(1) = 12
        itemArray.itemNo(2) = 17
        itemArray.itemNo(3) = 24
        itemArray.itemNo(4) = 25
        itemArray.itemNo(5) = 27
        itemArray.itemNo(6) = 28
        itemArray.itemNo(7) = 45
        itemArray.itemName(0) = "SEAFOOD CHOWDER"
        itemArray.itemName(1) = "CHOW CHOWDER"
        itemArray.itemName(2) = "CLAM CHOWDER"
        itemArray.itemName(3) = "TOMATO SOUP"
        itemArray.itemName(4) = "CHICKEN SOUP"
        itemArray.itemName(5) = "VEGETABLE SOUP"
        itemArray.itemName(6) = "ONION SOUP"
        itemArray.itemName(7) = "GREEN PEA SOUP"
        itemArray.itemName(8) = "VYNTON SOUP"
        itemInfo = OpenFileDialog1.OpenFile("soup", OpenMode.Input)
    End Sub

    Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtOrderNo.Click, txtItemNo.Click, txtNumOrd.Click
        txtOrderNo.Clear()
        txtItemNo.Clear()
        txtNumOrd.Clear()
    End Sub
```
Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
    FileClose(1)
    Me.Close()
End Sub

Private Sub btnRead_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRead.Click
    Dim wkOrderNo As Integer, wkItemNo As Integer
    Dim wkNumOrd As Integer
    If Not EOF(1) Then
        Input(1, wkOrderNo)
        Input(1, wkItemNo)
        wkNumOrd = wkItemNo
        txtnoOrdNo.Text = wkOrderNo
        txtnoItemNo.Text = wkItemNo
        txtnoNumOrd.Text = wkNumOrd
        txtItemName.Text = SearchArray(wkItemNo)
        Else
        MessageBox.Show("EOF reached")
        btnRead.Visible = False
        End If
    End Sub

Function SearchArray(ByVal wkItemNo) As Integer
    Dim itemsUB As Integer = 0
    Dim searchInd As String = "NO"
    Do Until itemsUB > 0 Or searchInd = "YES"
        If wkItemNo = ItemArray(itemsUB).ItemNo Then
            searchInd = "YES"
        Else
            itemsUB = itemsUB + 1
        End If
    Loop
Private Sub btnRead_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRead.Click
    MsgBox("SOF reached")
    btnRead.Visible = False
End Sub

Function SearchArray(ByVal wItemNo As Integer) As Boolean
    Dim itemsSub As Integer = 0
    Dim matchInd As String = "NO"
    Do Until itemsSub > 0 Or matchInd = "YES"
        If wItemNo = itemarray(itemsSub).ItemNo Then
            matchInd = "YES"
        Else
            itemsSub = itemsSub + 1
        End If
    Loop
Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
    FileClose(1)
    Me.Close()
End Sub

Private Sub btnRead_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRead.Click
    Dim wkOrderNo As Integer, wkItemNo As Integer
    Dim wkNumOrd As Integer
    If Not EOF(1) Then
        Input(1, wkOrderNo)
        Input(1, wkItemNo)
        Input(1, wkNumOrd)
        txtOrderNo.Text = wkOrderNo
        txtItemNo.Text = wkItemNo
        txtNumOrd.Text = wkNumOrd
        txtItemName.Text = SearchArray(wkItemNo)
    Else
        MessageBox.Show("EOF reached")
        btnRead.Visible = False
    End If
End Sub

Function SearchArray(ByVal wkItemNo As Integer) As Boolean
    Dim itemsSub As Integer = 0
    Dim matchInd As String = "NO"
    Do Until itemsSub > 0 Or matchInd = "YES"
        If wkItemNo = itemarray(itemsSub).ItemNo Then
            matchInd = "YES"
        Else
            itemsSub = itemsSub + 1
        End If
    Loop
End Function
```vbnet
End If
Loop

1111, 12, 1
1212, 27, 2
1224, 45, 4
1225, 28, 2
2003, 17, 1
2027, 28, 4
3003, 3, 3
3010, 34, 3
4012, 15, 5
4078, 50, 2
5000, 19, 1
5012, 24, 4
```
I am executing the function that does the search. I am sending wkItemNo. The return puts results in txtItemName.text.
This is an alternate approach that allows the two things: item number and item name to be stored together (we looked at this before).
This approaches access through functions such as substring.
Looking at ReDim of the array.

Public Class frmTestReDim
    Inherits System.Windows.Forms.Form
    Dim accuarray() As Integer
    Dim ct As Integer
    Dim vtOriginal As Integer
    Dim vtCount As Integer

    Windows Form Designer generated code

    Private Sub btnStartGather_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnStartGather.Click
        vtOriginal = InputBox("Enter number of original entries")
        ReDim accuarray(vtOriginal - 1)
        lstOriginal.Items.Add("Entries: " & CStr(accuarray.Length)
        ct = 0
        btnGatherAmount.Enabled = True
        btnClear.Enabled = True
        btnStartGather.Enabled = False
        txtAmount.Focus()
    End Sub

    Private Sub btnGatherAmount_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGatherAmount.Click
        accuarray(ct) = CInt(txtAmount.Text)
        lstOriginal.Items.Add(accuarray(ct))
        ct = ct + 1
        btnGatherAmount.Enabled = False
    End Sub

    Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
        txtAmount.Clear()
        If ct <= accuarray.Length Then
            btnGatherAmount.Enabled = True
            txtAmount.Focus()
        End If
    End Sub
Without using preserve with ReDim. Preserve would keep the first four numbers.
Private Sub btnGather_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    accarray(ct) = CInt(txtAmountSecond.Text)
    lstSecond.Items.Add(accarray(ct))
    ct = ct + 1
    btnGather.Enabled = False
End Sub

Private Sub btnClearSecond_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    If ct < accarray.Length Then
        btnGather.Enabled = True
        txtAmountSecond.Clear()
    Else
        btnShowContents.Enabled = True
    End If
End Sub

Private Sub btnShowContents_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim i As Integer
    For i = 0 To accarray.Length - 1
        lstShowArray.Items.Add(accarray(i))
    Next
End Sub
Looking at two forms modal.
Looking at menu building.
Private Sub mnuAdd_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuAdd.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 + num2
End Sub

Private Sub mnuSubtract_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuSubtract.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 - num2
End Sub

Private Sub mnuMultiply_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuMultiply.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 * num2
End Sub

Private Sub mnuDivide_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuDivide.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 / num2
End Sub

Private Sub mnuAnsRed_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuAnsRed.Click
    txtAns.ForeColor = Color.Red
End Sub

Private Sub mnuısRed_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuı.Click
    txtıs.ForeColor = Color.Red
End Sub

Private Sub mnuPopupAdd_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuPopupAdd.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 + num2
End Sub

Private Sub mnuPopupSubtract_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuPopupSubtract.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 - num2
End Sub

Private Sub mnuPopupMultiply_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuPopupMultiply.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 * num2
End Sub

Private Sub mnuPopupDivide_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuPopupDivide.Click
    Dim num1, num2 As Integer
    num1 = CInt(txtFirst.Text)
    num2 = CInt(txtSecond.Text)
    txtAns.Text = num1 / num2
End Sub