I entered dept aaa, bbb, ccc, ddd.
In this two dimensional table, the first index/subscript number i pointing to the numbers going down the side and the second is pointing to the numbers across the top.
The from code is the first column and the two code the second column. I could have set things up differently but this was the structure I used.
If IsNumeric(txtToF.Text) And CInt(txtToF.Text) < 3 Then

txTfare.Text = “From City is not valid, To City is valid”
Else

txTfare.Text = “Both From City and To City are invalid”
End If
End If
End Sub

Private Sub frmAirFace_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

fromCity[0] = “Boston, MA”
fromCity[1] = “New York, NY”
fromCity[2] = “Austin, TX”
toCity(0) = “Istanbul”
toCity(1) = “Tokyo”
toCity(2) = “London”
priceArray(0, 0) = 800.97
priceArray(0, 1) = 967.98
priceArray(0, 2) = 289.99
priceArray(1, 0) = 745.75
priceArray(1, 1) = 923.45
priceArray(1, 2) = 235.97
priceArray(2, 0) = 967.98
priceArray(2, 1) = 1120.98
priceArray(2, 2) = 496.90
priceArray(3, 0) = 750.75
priceArray(3, 1) = 950.75
priceArray(3, 2) = 250.90

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click

txtFrom.Clear()
End Sub

The price to fly from Austin, TX to Baku is $1,120.98
Private Sub btnGetPrice_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGetPrice.Click
    Dim fromPnt As Integer, toPnt As Integer, wkPrice As Single
    Dim wkXsg As String
    If IsNumeric(txtFrom.Text) And CInt(txtFrom.Text) < 4 Then
        fromPnt = CInt(txtFrom.Text)
    Else
        fromPnt = 0
    End If
    If IsNumeric(txtTo.Text) And CInt(txtTo.Text) < 3 Then
        toPnt = CInt(txtTo.Text)
    Else
        toPnt = 0
    End If
    wkPrice = priceArray(fromPnt, toPnt)
    wkXsg = "The price to fly from " & fromCity(fromPnt) & 
    " to " & toCity(toPnt) & " is " & FormatCurrency(wkPrice)
    txtFare.Text = wkXsg
Else
    txtFare.Text = "From City is valid, To City is not valid"
End If
Else
    If IsNumeric(txtTo.Text) And CInt(txtTo.Text) < 3 Then
        txtFare.Text = "From City is not valid, To City is valid"
    Else
        txtFare.Text = "Both from City and To City are invalid"
    End If
End If
End Sub

Private Sub frmAirfare_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles frmAirfare.Load
    FromCity(0) = "Boston, MA"
    FromCity(1) = "New York, NY"
    FromCity(2) = "Austin, TX"
End Sub
The and crashed when I entered a letter.
Private Sub btnSetPrice_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSetPrice.Click
    Dim fromPtr As Integer, toPtr As Integer, wPrice As Single
    Dim okMsg As String
    If IsNumeric(txtFrom.Text) And CInt(txtFrom.Text) < 1 Then
        fromPtr = CInt(txtFrom.Text)
    End If
    If IsNumeric(txtTo.Text) And CInt(txtTo.Text) < 3 Then
        toPtr = CInt(txtTo.Text)
        wPrice = pricesArray(fromPtr, toPtr)
        okMsg = "The price to fly from " & fromCity(fromPtr) & " to " & toCity(toPtr) & " is " & FormatCurrency(wPrice)
    Else
        okMsg = "From City is valid, To City is valid"
    End If
    Else
        txtFare.Text = "From City is valid, To City is invalid"
    End If
    Else
        txtFare.Text = "Both from City and To City are invalid"
    End If
End Sub

Private Sub frmAirfare_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    FROMCity(0) = "Boston, MA"
    FROMCity(1) = "New York, NY"
    FROMCity(2) = "Austin, TX"
    FROMCity(3) = "Washington, DC"
    toCity(0) = "Istanbul"
    toCity(1) = "Baku"
End Sub
Private Sub btnSetPrice_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim fromPtr As Integer, toPtr As Integer, whPrice As Single
    Dim okMsg As String

    If IsNumeric(txtFrom.Text) And CInt(txtFrom.Text) < 4 Then
        fromPtr = CInt(txtFrom.Text)
    End If
    If IsNumeric(txtTo.Text) And CInt(txtTo.Text) > 3 Then
        toPtr = CInt(txtTo.Text)
    End If
    okPrice = priceArray(fromPtr, toPtr)
    okMsg = "The price to fly from " & fromCity(fromPtr) & ", " & toCity(toPtr) & " is " & FormatCurrency(okPrice)
    txtFare.Text = okMsg
End Sub

Debugging: Worked fine with just Is Numeric

Private Sub frmAirfare_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
    FROMCity(0) = "BOSTON, MA"
    FROMCity(1) = "New York, NY"
    FROMCity(2) = "Austin, TX"
    FROMCity(3) = "Washington, DC"
    toCity(0) = "Istanbul"
    toCity(1) = "Baku"
End Sub
Worked fine when I asked questions separately.

Need to fix this one too!
Private Sub btnSetPrice_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSetPrice.Click
Dim fromPtc As Integer, toPtc As Integer, price As Single
Dim okMsg As String
If IsNumeric(txtFrom.Text) And CInt(txtFrom.Text) < 1 Then
    fromPtr = CInt(txtFrom.Text)
    If IsNumeric(txtTo.Text) And CInt(txtTo.Text) < 3 Then
        toPtr = CInt(txtTo.Text)
        price = pricesArray(fromPtr, toPtr)
        okMsg = "The price to fly from " & fromCity(fromPtr) & " to " & toCity(toPtr) & " is " & FormatCurrency(price)
        txtFare.Text = okMsg
    Else
        txtFare.Text = "From City is valid, To City is not valid"
    End If
Else
    If IsNumeric(txtTo.Text) Then
        If CInt(txtTo.Text) < 9 Then
            txtFare.Text = "From City is not valid, To City is valid"
        Else
            txtFare.Text = "From City and To City are invalid"n
        End If
    Else
        txtFare.Text = "Both From City and To City are invalid"
    End If
End If
End Sub
Parallel arrays

Private Sub formDept_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles formDept.Load
depthNumArray(0) = 15
depthNumArray(1) = 26
depthNumArray(2) = 37
depthNumArray(3) = 56
depthArray(0) = "Books"
depthArray(1) = "Toys"
depthArray(2) = "Gifts"
depthArray(3) = "Cookware"
End Sub

Private Sub btnListDept_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Dim i As Integer
For i = 0 To 3
    listDept.Items.Add(depthNumArray(i) & " " & depthArray(i))
Next
End Sub
End Class
Public Class frmDept
    Inherits System.Windows.Forms.Form
    Dim deptNumArray(3) As Integer
    Dim deptArray(3) As String

    Windows Form Designer generated code

    Private Sub frmDept_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        deptNumArray(0) = 15
        deptNumArray(1) = 26
        deptNumArray(2) = 37
        deptNumArray(3) = 56
        deptArray(0) = "Books"
        deptArray(1) = "Toys"
        deptArray(2) = "Gifts"
        deptArray(3) = "Cookware"
    End Sub

    Private Sub btnListDept_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnListDept.Click
        Dim i As Integer
        For i = 0 To 3
            lstDept.Items.Add(deptNumArray(i) & " " & deptArray(i))
        Next
    End Sub
End Class
Defining the array differently and also including it in a module.
Module modArray
    Public deptArray() As String = {"Books", "Toys", "Gifts", "Cookware"}
End Module
Public Class frmDept
    Inherits System.Windows.Forms.Form

    Windows Form Designer generated code

    Private Sub btnCheckDept_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        If IsNumeric(txtDept.Text) Then
            If CInt(txtDept.Text) <= 3 Then
                txtDeptName.Text = deptArray(CInt(txtDept.Text))
            Else
                txtDeptName.Text = "Invalid Department Number"
            End If
        Else
            txtDeptName.Text = "Non-numeric Department Number"
        End If
    End Sub
End Class
Public Class frmDept

    Inherits System.Windows.Forms.Form

Private Sub btnCheckDept_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCheckDept.Click
    If IsNumeric(txtDept.Text) Then
        If CInt(txtDept.Text) <= 3 Then
            txtDeptName.Text = deptArray(CInt(txtDept.Text))
        Else
            txtDeptName.Text = "Invalid Department Number"
        End If
    Else
        txtDeptName.Text = "Non-numeric Department Number"
    End If
End Sub
End Class
Module modArray
    Public deptarray() As String = {"Books", "Toys", "Gifts", "Cookware"}
End Module
Private Sub formSoupCreate_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles formSoupCreate.Load
    itemNoArray(0) = 1
    itemNoArray(1) = 12
    itemNoArray(2) = 15
    itemNoArray(3) = 17
    itemNoArray(4) = 24
    itemNoArray(5) = 25
    itemNoArray(6) = 28
    itemNoArray(7) = 38
    itemNoArray(8) = 45
    itemNameArray(0) = "SEAFOOD CHOWDER"
    itemNameArray(1) = "CORN CHOWDER"
    itemNameArray(2) = "CLAM CHOWDER"
    itemNameArray(3) = "TOMATO SOUP"
    itemNameArray(4) = "CHICKEN SOUP"
    itemNameArray(5) = "VEGETABLE SOUP"
    itemNameArray(6) = "ONION SOUP"
    itemNameArray(7) = "GREEN PEA SOUP"
    itemNameArray(8) = "WONTON SOUP"
    FileOpen(1, "soup", OpenMode.Input)  
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
    txtOrderNo.Clear()
    txtItemNo.Clear()
    txtNumOrdered.Clear()
End Sub

Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
    FileClose(1)
Private Sub fmSoupCreate_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles sender.Load
    itemNoArray(0) = 1
    itemNoArray(1) = 12
    itemNoArray(2) = 15
    itemNoArray(3) = 17
    itemNoArray(4) = 24
    itemNoArray(5) = 25
    itemNoArray(6) = 17
    itemNoArray(7) = 28
    itemNoArray(8) = 45
    itemNameArray(0) = "SEAFOOD CHOWDER"
    itemNameArray(1) = "CORN CHOWDER"
    itemNameArray(2) = "CLAM CHOWDER"
    itemNameArray(3) = "TOMATO SOUP"
    itemNameArray(4) = "CHICKEN SOUP"
    itemNameArray(5) = "VEGETABLE SOUP"
    itemNameArray(6) = "ONION SOUP"
    itemNameArray(7) = "GREEN PEA SOUP"
    itemNameArray(8) = "WONTON SOUP"
    FileOpen(1, "soup", OpenMode.Input)
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles sender.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)
End Sub
Text data being read. Match being found in top array and then corresponding element being taken from second array.
Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles sender.Click
    txtOrderNo.Clear()
    txtItemNo.Clear()
    txtNumOrd.Clear()
End Sub

Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles sender.Click
    FileClose(1)
    Me.Close()
End Sub

Private Sub btnRead_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles sender.Click
    Dim wOrderNo As Integer, wItemNo As Integer
    Dim wNumOrd As Integer
    If Not EOF(1) Then
        Input(1, wOrderNo)
        Input(1, wItemNo)
        Input(1, wNumOrd)
        txtOrderNo.Text = wOrderNo
        txtItemNo.Text = wItemNo
        txtNumOrd.Text = wNumOrd
        txtname.Text = SearchArray(wItemNo)
End Sub
Search function to search for a match in the first table and then extract the name from the second table.
```
Function SearchArray(ByVal wkItemNo) As String
    Dim itemSub As Integer = 0
    Dim matchInd As String = "NO"
    Do Until itemSub > 6 Or matchInd = "YES"
        If wkItemNo = itemNoArray(itemSub) Then
            matchInd = "YES"
        Else
            itemSub = itemSub + 1
        End If
    Loop
    If matchInd = "YES" Then
        Return itemNoArray(itemSub)
    Else
        Return "Match Not Found"
    End If
End Function
```

```
Function SearchWhile(ByVal wkItemNo) As String
    Dim itemSub As Integer = 0
    Dim matchInd As String = "NO"
    Do While itemSub < 9 And matchInd = "DO"
        If wkItemNo = itemNoArray(itemSub) Then
            matchInd = "YES"
        Else
            itemSub = itemSub + 1
        End If
    Loop
    Return itemNoArray(itemSub)
End Function
```
Dim itemNameArray(9) As String

Private Sub frmSoupCreate_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles frmSoupCreate_Load
    itemNoArray(0) = 3
    itemNoArray(1) = 12
    itemNoArray(2) = 15
    itemNoArray(3) = 17
    itemNoArray(4) = 14
    itemNoArray(5) = 25
    itemNoArray(6) = 17
    itemNoArray(7) = 20
    itemNoArray(8) = 45
    itemNameArray(0) = "SEAFOOD CHOWDER"
    itemNameArray(1) = "CORN CHOWDER"
    itemNameArray(2) = "CLAM CHOWDER"
    itemNameArray(3) = "TOMATO SOUP"
    itemNameArray(4) = "CHICKEN SOUP"
    itemNameArray(5) = "VEGETABLE SOUP"
    itemNameArray(6) = "ONION SOUP"
    itemNameArray(7) = "GREEN PEA SOUP"
    itemNameArray(8) = "WONTON SOUP"
    FileOpen(1, "soup", OpenMode.Input)
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
    txtOrderNo.Clear()
    txtItemNo.Clear()
    txtNumOrd.Clear()
End Sub
Function SearchArray(ByVal wItemNo)
    Dim itemSub As Integer = 0
    Dim matchInd As String = "NO"
    Do Until itemSub > 8 Or matchInd = "YES"
        If wItemNo = itemNumArray(itemSub) Then
            matchInd = "YES"
        Else
            itemSub = itemSub + 1
        End If
    Loop
    If matchInd = "YES" Then
        Return itemNumArray(itemSub)
    Else
        Return "Match Not Found"
    End If
End Function

Function SearchWhile(ByVal wItemNo)
    Dim itemSub As Integer = 0
    Dim matchInd As String = "NO"
    Do While itemSub < 8 And matchInd = "NO"
        If wItemNo = itemNumArray(itemSub) Then
            matchInd = "YES"
        Else
            itemSub = itemSub + 1
        End If
    Loop
    If matchInd = "YES" Then
Private Sub frmSoupCreate_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles frmSoupCreate.Load
    itemNoArray(0) = 1
    itemNoArray(1) = 12
    itemNoArray(2) = 15
    itemNoArray(3) = 17
    itemNoArray(4) = 24
    itemNoArray(5) = 25
    itemNoArray(6) = 17
    itemNoArray(7) = 18
    itemNoArray(8) = 45
    itemNameArray(0) = "SEAFOOD CHOWDER"
    itemNameArray(1) = "CORN CHOWDER"
    itemNameArray(2) = "CLAM CHOWDER"
    itemNameArray(3) = "TOMATO SOUP"
    itemNameArray(4) = "CHICKEN SOUP"
    itemNameArray(5) = "VEGETABLE SOUP"
    itemNameArray(6) = "ONION SOUP"
    itemNameArray(7) = "GREEN PEA SOUP"
    itemNameArray(8) = "WONTON SOUP"
    FileOpen(1, "soup", OpenMode.Input)
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.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)
Private Sub frmSoupCreate_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles frmSoupCreate.Load
    itemNoArray(0) = 1
    itemNoArray(1) = 12
    itemNoArray(2) = 15
    itemNoArray(3) = 17
    itemNoArray(4) = 24
    itemNoArray(5) = 25
    itemNoArray(6) = 17
    itemNoArray(7) = 28
    itemNoArray(8) = 45
    itemNameArray(0) = "SEAFOOD CHOWDER"
    itemNameArray(1) = "CORN CHOWDER"
    itemNameArray(2) = "CLAM CHOWDER"
    itemNameArray(3) = "TOMATO SOUP"
    itemNameArray(4) = "CHICKEN SOUP"
    itemNameArray(5) = "VEGETABLE SOUP"
    itemNameArray(6) = "ONION SOUP"
    itemNameArray(7) = "GREEN PEA SOUP"
    itemNameArray(8) = "WONTON SOUP"
    FileOpen(1, "soup", OpenMode.Input)
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.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)
End Sub
You quit the search for one of two reasons:
1) You pass the number of elements so you have checked everything
2) You found a match.

Using a while loop instead of the until.
Dim itemSub As Integer = 0
Dim matchInd As String = "NO"
Do Until itemSub > 9 Or matchInd = "YES"
   If wkItemNo = itemNameArray(itemSub) Then
      matchInd = "YES"
   Else
      itemSub = itemSub + 1
   End If
Loop
If matchInd = "YES" Then
   Return itemNameArray(itemSub)
Else
   Return "Match Not Found"
End If
'End Function

Function SearchWhile(ByVal wkItemNo) as Integer
   Dim itemSub As Integer = 0
   Dim matchInd As String = "NO"
   Do While itemSub < 9 And matchInd = "NO"
      If wkItemNo = itemNameArray(itemSub) Then
         matchInd = "YES"
      Else
         itemSub = itemSub + 1
      End If
   Loop
   If matchInd = "YES" Then
      Return itemNameArray(itemSub)
   Else
      Return "Match Not Found"
   'End Function
Putting them together in one array. The first two numbers are the thing I will search and from there on is what I will extract.
Using substring to check the first two digits. Using substring to extract after the first two until the end.
Function SearchArray(byVal wkItemNo)
    Dim itemSub As Integer = 0
    Dim matchInd As String = "NO"
    Do Until itemSub > 0 Or matchInd = "YES"
        If wkItemNo = Microsoft.VisualBasic.Left(itemArray(itemSub), 2) Then
            matchInd = "YES"
        Else
            itemSub = itemSub + 1
        End If
    Loop
    If matchInd = "YES" Then
        Return Microsoft.VisualBasic.Mid(itemArray(itemSub), 3)
    Return itemArray(itemSub).Substring(2)
Else
    Return "Match Not Found"
End If
End Function
The numbers start in 0 and go for 2
The words start in 2 and go until end