**Select Case**

Select Case is a conditional statement, that helps us to test a variable for equality against a set of values. Each value is referred to as a case, and a variable that is being switched on should be checked for all the select cases.

The Select Case statement provides us with an easy way of testing for the contents of a variable. However, it is only suitable for use when a variable in question has only a limited number of options.

**Syntax of Select Case**

Select [ Case ] your\_expression

[ Case expression\_list



[ statement(s) ] ]

[ Case Else



[ else statement(s) ] ]

End Select

your\_expression: this denotes an expression which evaluates to one of the elementary data types supported in Microsoft VB.NET.

expression\_list: expression clauses that denote the match values for the expression. For the case of multiple clauses, separate them using a comma (,).

statement(s): statements that follow the Case and they execute after the select expression has matched any clause in expression\_list.

else statements: statements that follow the Case Else and run once the select expression fails to match any of the clauses in the expression\_list for any Case statement.

Create a new Windows Application project. Name it Select Demo.

1. From the tool box, add a ListBox control to the form that appears and resize it so that it takes up the entire form.
2. Set its IntegralHeight property to False.
3. To make the Listbox stretches with the form, So we have to set its Anchor Property to Top,Bottom, Left, Right.
4. Look at the properties window and select the Items property and select the String Collection Editor that appears and add the names on separate lines
5. Click ok to save changes.

**VB Code**

Dim name, fcolor As String

‘ what did we choose

name=ListBox1.Items(ListBox1.SelectedIndex)

‘use a select case to do something

Select Case name



Case “Anchal”



fcolor = ” Yellow”



Case “Saheli”

fcolor = ” Purple”

Case “Nainika”

fcolor=”Green”

Case “Saswati”

fcolor = ”Maroon”

Case “Monalisa”



fcolor = ”Pink ”

Case “Sucharita”

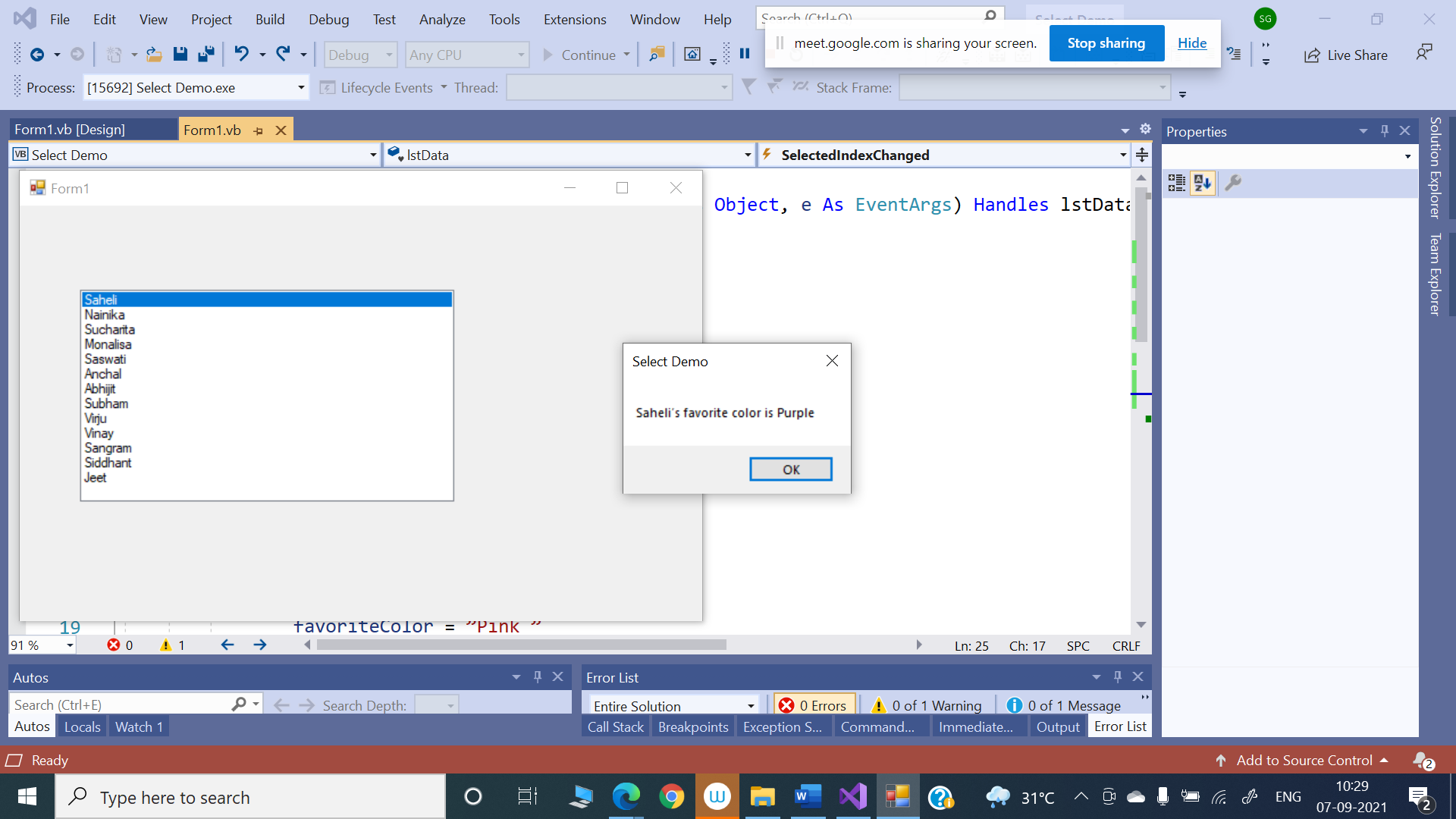
fcolor = ”Red”

End Select

‘display a message box…

MessageBox.Show(name & “’s favorite color is” & fcolor,-“Select Case Demo”)







**Case Insensitive Select Case**



‘what did we choose

Dim name As String

name=lstData.Items(lstData.SelectedIndex)

‘use a select case to do something

Dim favoriteColor As String

Select Case name.ToLower

Case “anchal”

favoriteColor=” Yellow”

Case “saheli”

favoriteColor=” Purple”

Case “nainika”

favoriteColor=”Green”

Case “saswati”

favoriteColor=”Maroon”

Case “monalisa”

favoriteColor=”Pink ”

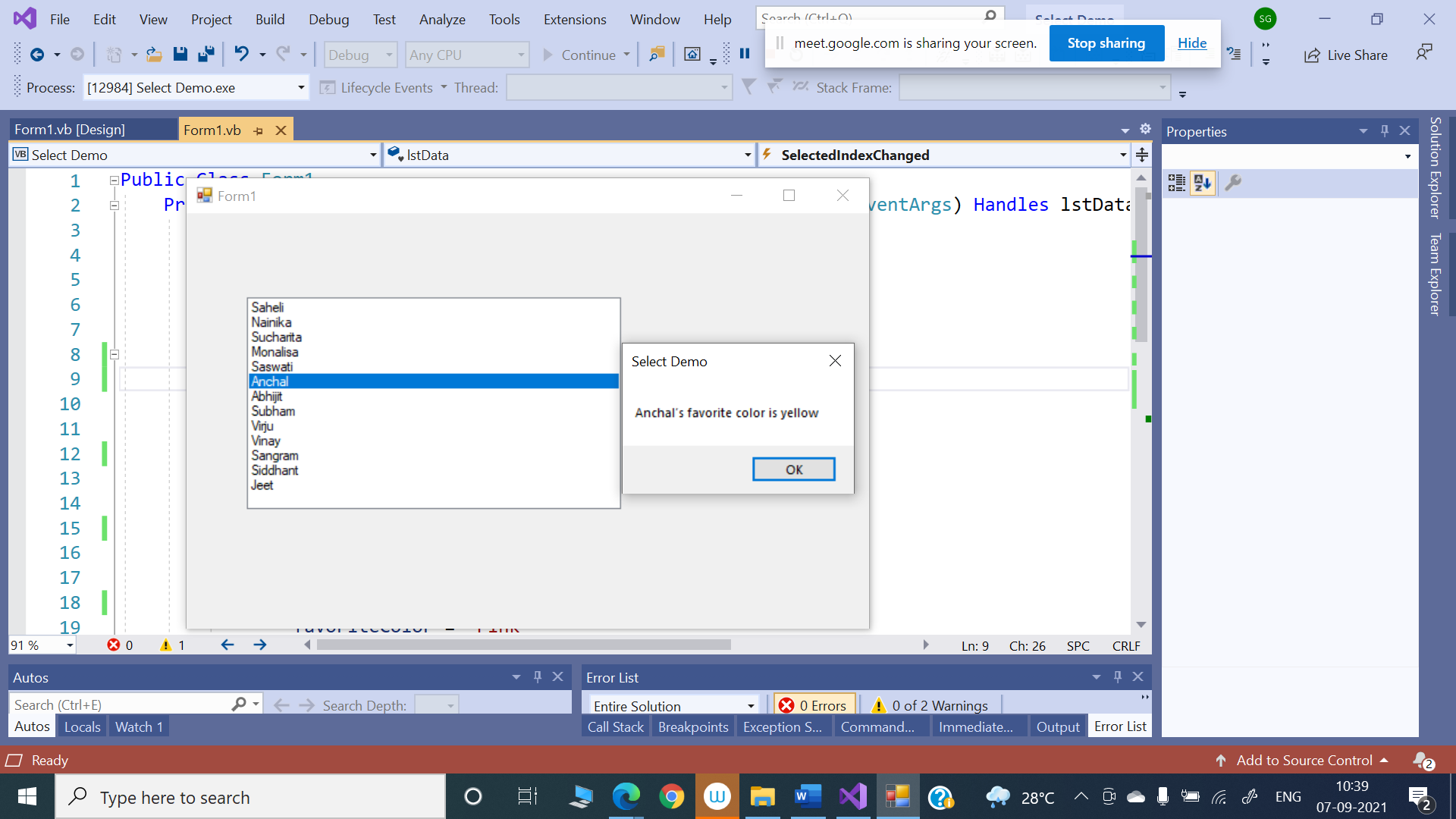
Case “sucharita”

favoriteColor=”Red”

End Select

‘display a message box…

MessageBox.Show(name & “’s favorite color is” & favoriteColor,-“Select Deom”)



**Multiple Selections**

We are not limited to matching one value inside a Select Case…End Select block. We can also match multiple items.

VB Code

Public Class Form1

Private Sub ListBox1\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ListBox1.SelectedIndexChanged

'what did we choose

Dim name As String

name = ListBox1.Items(ListBox1.SelectedIndex)

'use a select case to do something

Dim favoriteColor As String

Select Case name

Case “Anchal”, “Saheli”

'display a message box…

MessageBox.Show(“Female”, “Select Demo”)

Case “Abhijit”, “Vinay”

'display a message box…

MessageBox.Show(“Male”, “Select Demo”)

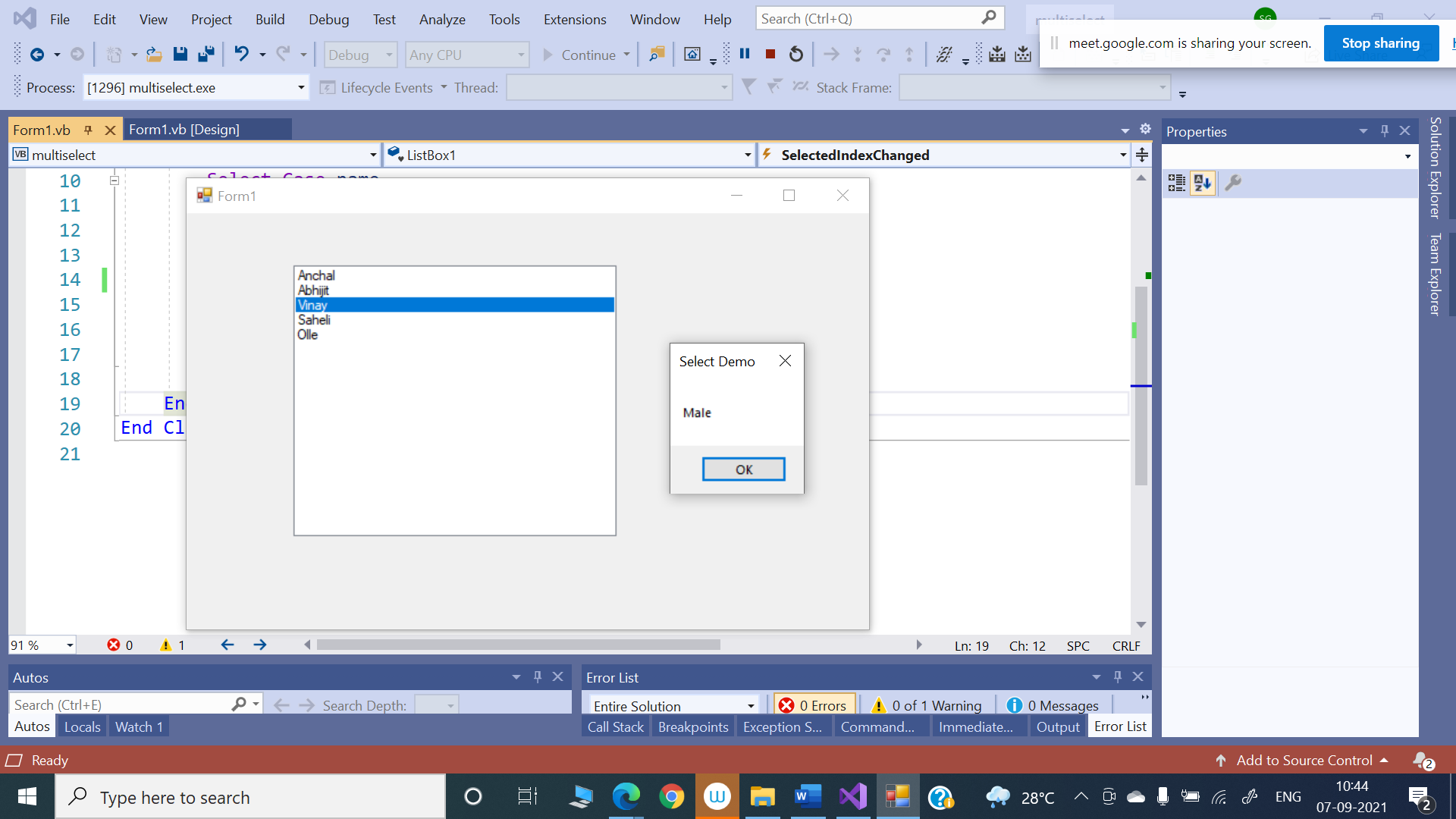
End Select

End Sub

End Class

However, on each Case statement we can provide a list of possible values separated with commas.

In the first one, we look for anchal or Saheli. If either of these matches, we run the code under the case statement as usual.



**The Case Else Statement**

So what happens if none of the Case statements that we have included is matched? The solution for this is Case Else Statement.

‘what did we choose

Dim name As String

name=lstData.Items(lstData.SelectedIndex)

‘use a select case to do something

Select Case name.ToLower

Case “Anchal”, “Saheli”

‘display a message box…

MessageBox.Show(“Female”, “Select Demo”)

Case “Abhijit”, “Vinay”

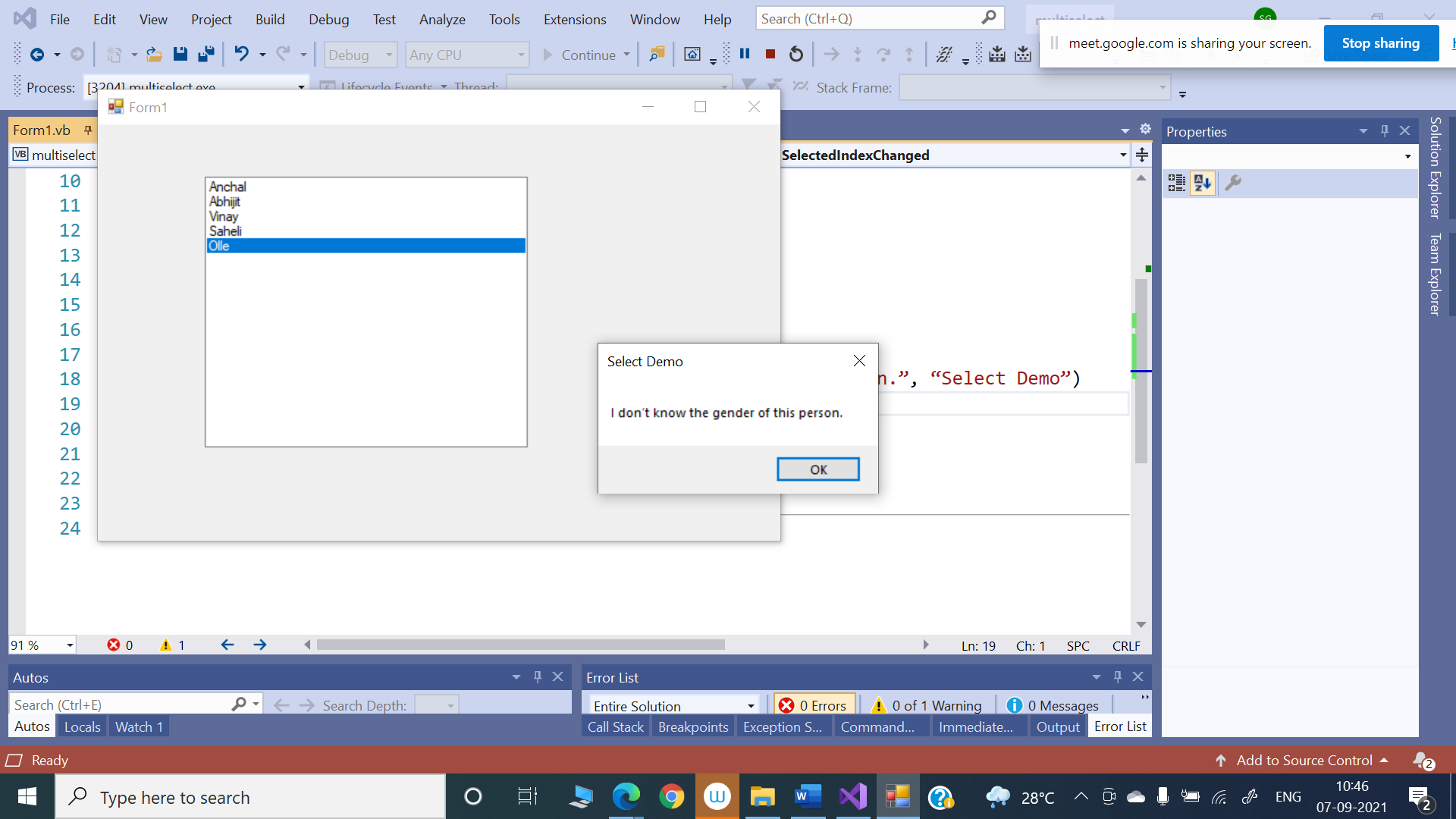
‘display a message box…

MessageBox.Show(“Male”, “Select Demo”)

Case Else

MessageBox.Show(“I don’t know the gender of this person.”, “Select Demo”)

End Select



**Different Data Types with Select case**

**We can use Select case with all of the basic data types in VB.NET such as Integer, Double and Boolean. In day-to-day work, the most common types of Select Case will be based on String and Integer data types.**

Program with Integer Data type

dim n as integer

Console.WriteLine("Enter a value to know a day of a week")

n= console.readline()

select case n

case 1: Console.writeline("Monday")

case 2: Console.writeline("Tuesday")

case 3: console.writeline("Wednesday")

case 4: console.writeline("Thursday")

case 5: console.writeline("Friday")

case 6: console.writeline("Saturday")

case 7: console.writeline("Sunday")

case else

Console.writeline("Week has only 7 days")

end select

