***Program Cover Sheet***

|  |
| --- |
| Name: |
| Assignment: |
| List any parts of the assignment that do not work/were not completed: |

|  |
| --- |
| Instructor’s Comments: |
| Grade: |

Program Submission Requirements: (1) all files, zipped and uploaded to Canvas and (2) a completed cover sheet, program execution screenshots and source code printed, **stapled** and turned in during class. Failure to follow the submission requirements will result in points lost on that particular assignment.

'------------------------------------------------------------

'- File Name : frmChezSouSad.frm -

'- Part of Project: Assign4 -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- File Purpose: -

'- This file contains the main application form where the -

'- user will decide their menu for the their meal. The user

' can add specific raw ingredients to be added to their

' selected prepped item which then can be added to the dish.

'------------------------------------------------------------

'- Program Purpose: -

'- -

'- This program uses dictionaries to create an active menu

' selection for users. These dictionaries are nested and

' communicate with eachother to effectively while adding,

' removing, adn sorting the list items and selected items.

' This program also error handles bad input.

'------------------------------------------------------------

'- Global Variable Dictionary (alphabetically): -

'- gdicDishes - global dictionary to hold the dish dictionary

' that holds a string, as well as another

' dictionary(prepped) that also holds a dictionary(raw)

'– gdicPreppedItems - global dictionary to hold the prepped

' items and also another dictionary

' to hold the raw dictionary

'- gdicRawItems - global dictionary to hold the raw items

'- PATTERN - a regex pattern to validate user input from the

' txt boxes, does not allow special characters to be added

'- STRDUPLICATEERROR - a simple string that holds an error message

'- STRREGEXERROR - a simple string that holds an error message

'------------------------------------------------------------

Imports System.Text.RegularExpressions

Public Class frmChezSouSad

'global dictionary to hold the raw items

Public gdicRawItems As New SortedDictionary(Of String, String)

'global dictionary to hold the prepped items and also another dictionary to hold the raw dictionary

Public gdicPreppedItems As New SortedDictionary(Of String, SortedDictionary(Of String, String))

'global dictionary to hold the dish dictionary that holds a string, as well as another dictionary(prepped) that also holds a dictionary(raw)

Public gdicDishes As New SortedDictionary(Of String, SortedDictionary(Of String, SortedDictionary(Of String, String)))

'a regex pattern to validate user input from the txt boxes, does not allow special characters to be added

Const PATTERN As String = "^[a-zA-Z]+$"

'a simple string that holds an error message

Const STRDUPLICATEERROR As String = "Selection already added."

'a simple string that holds an error message

Const STRREGEXERROR As String = "No special characters allowed. Please enter a valid food item."

Private Sub frmChezSouSad\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

'------------------------------------------------------------

'- Subprogram Name: frmChezSouSad\_Load -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever form loads. The form

'– will have some preloaded data for user convienience, and

' and that information is stored and created here.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- dicChickenSalad- Creating a dictionary to hold some selected

' raw chicken salad ingredients

'- dicChickenSaladPlatter - creating a dictionary to hold the

' prepped dictionary anda string

'- dicHamburger-Creating a dictionary to hold some selected

' raw hamburger ingredients

'- dicHamburgerPlatter- creating a dictionary to hold the

' prepped dictionary anda string

'- dicSoda- Creating a dictionary to hold some selected raw

' soda ingredients

'------------------------------------------------------------

'adding hardcoded random raw ingredients to the list like you had mentioned in the lecture

gdicRawItems.Add("beef patty", "beef patty")

gdicRawItems.Add("bun", "bun")

gdicRawItems.Add("chicken", "chicken")

gdicRawItems.Add("glass", "glass")

gdicRawItems.Add("lettuce", "lettuce")

gdicRawItems.Add("basket", "basket")

gdicRawItems.Add("mayo", "mayo")

gdicRawItems.Add("sugar", "sugar")

gdicRawItems.Add("water", "water")

'Creating a dictionary to hold some hard coded values

Dim dicHamburger As New SortedDictionary(Of String, String)

'adding some raw ingredients to this prepped ingredient

dicHamburger.Add("bun", gdicRawItems("bun"))

dicHamburger.Add("beef patty", gdicRawItems("beef patty"))

dicHamburger.Add("basket", gdicRawItems("basket"))

'adding the prepped item to the prepped item dictionary

gdicPreppedItems.Add("hamburger", dicHamburger)

'Creating a dictionary to hold some hard coded values

Dim dicSoda As New SortedDictionary(Of String, String)

'adding some raw ingredients to this prepped ingredient

dicSoda.Add("sugar", gdicRawItems("sugar"))

dicSoda.Add("water", gdicRawItems("water"))

dicSoda.Add("glass", gdicRawItems("glass"))

'adding the prepped item to the prepped item dictionary

gdicPreppedItems.Add("soda", dicSoda)

'Creating a dictionary to hold some hard coded values

Dim dicChickenSalad As New SortedDictionary(Of String, String)

'adding some raw ingredients to this prepped ingredient

dicChickenSalad.Add("lettuce", gdicRawItems("lettuce"))

dicChickenSalad.Add("chicken", gdicRawItems("chicken"))

dicChickenSalad.Add("mayo", gdicRawItems("mayo"))

'adding the prepped item to the prepped item dictionary

gdicPreppedItems.Add("chicken salad", dicChickenSalad)

'Creating a specific dish dictionary to hold some hard coded values

Dim dicChickenSaladPlatter As New SortedDictionary(Of String, SortedDictionary(Of String, String))

'adding the prepped item to the dish

dicChickenSaladPlatter.Add("chicken salad", gdicPreppedItems("chicken salad"))

'adding the dish to the dish dictionary

gdicDishes.Add("chicken salad platter", dicChickenSaladPlatter)

'Creating a specicic dish dictionary to hold some hard coded values

Dim dicHamburgerPlatter As New SortedDictionary(Of String, SortedDictionary(Of String, String))

'adding the prepped items to the dish

dicHamburgerPlatter.Add("hamburger", gdicPreppedItems("hamburger"))

dicHamburgerPlatter.Add("soda", gdicPreppedItems("soda"))

'adding the dish to the dish dictionary

gdicDishes.Add("hamburger platter", dicHamburgerPlatter)

'loop through each item added to the dictionary

For Each item In gdicRawItems.Values

'show them in the list box

lstRaw.Items.Add(item.ToLower)

Next

'loop through each item added to the dictionary

For Each item In gdicPreppedItems.Keys

'show them in the list box

lstPreppedItems.Items.Add(item.ToLower)

Next

'loop through each item added to the dictionary

For Each item In gdicDishes.Keys

'show them in the list box

lstDishes.Items.Add(item.ToLower)

Next

End Sub

Private Sub btnAddPreppedItem\_Click(sender As Object, e As EventArgs) Handles btnAddPreppedItem.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- add prepped item button. This function will check to see

'– if the user input is valid, and add the new information

'- to its own dictionary. Proper error handling is also in

' place.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- dicNewPrepped- creating a new dictionary to hold

' the selected prepped items

'- strText- a variable to hold the value of the text box

' so I dont have to type it so much

'------------------------------------------------------------

'creating a new dictionary to hold the selected information

Dim dicNewPrepped As New SortedDictionary(Of String, String)

'set a variable to hold the text so I dont have to type it so much

Dim strText As String = txtAddPreppedItem.Text.ToLower

'if the infromation added does not match the regex pattern

If Regex.Match(strText, PATTERN).Success = False Then

'give the user the error message

MessageBox.Show(STRREGEXERROR, "Error")

'clear the text box

txtAddPreppedItem.Clear()

'if the entered information is already added

ElseIf gdicRawItems.ContainsKey(strText) Then

'give the user the error message

MessageBox.Show(STRDUPLICATEERROR, "Error")

'clear the text box

txtAddPreppedItem.Clear()

Else

'if it does not fall into the two error catagories, then add it to the dictionary

gdicPreppedItems.Add(strText, dicNewPrepped)

'clear the text box

txtAddPreppedItem.Clear()

'show the info to the user in the selected text box

lstPreppedItems.Items.Add(strText.ToLower)

End If

End Sub

Private Sub btnAddRaw\_Click(sender As Object, e As EventArgs) Handles btnAddRaw.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- add raw item button. This function will check to see

'– if the user input is valid, and add the new information

'- to its own dictionary. Proper error handling is also in

' place.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- strAddingRaw - a variable to hold the value of the text box

' so I dont have to type it so much

'------------------------------------------------------------

'set a variable to hold the text so I dont have to type it so much

Dim strAddingRaw As String = txtRaw.Text.ToUpper

'if the infromation added does not match the regex pattern

If Regex.Match(strAddingRaw, PATTERN).Success = False Then

'give the user the error message

MessageBox.Show(STRREGEXERROR, "Error")

'clear the text box

txtRaw.Clear()

'if the entered information is already added

ElseIf gdicRawItems.ContainsKey(strAddingRaw) Then

'give the user the error message

MessageBox.Show(STRDUPLICATEERROR, "Error")

'clear the text box

txtRaw.Clear()

Else

'if it does not fall into the two error catagories, then add it to the dictionary

gdicRawItems.Add(strAddingRaw, strAddingRaw)

'clear the text box

txtRaw.Clear()

'loop through all of the items added

For Each item In gdicRawItems.Values

'show the info to the user in the selected text box

lstRaw.Items.Add(item.ToLower)

Next

End If

End Sub

Private Sub btnAddDish\_Click(sender As Object, e As EventArgs) Handles btnAddDish.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- add Dish button. This function will check to see

'– if the user input is valid, and add the new information

'- to its own dictionary. Proper error handling is also in

' place. –

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- dicNewDish- creating a new dictionary to hold the selected

' information from the list as well as the information

' from the other dictionary

'- strDishText- a variable to hold the value of the text box

' so I dont have to type it so much

'------------------------------------------------------------

'creating a new dictionary to hold the selected information from list as well as the information from the other dictionary

Dim dicNewDish As New SortedDictionary(Of String, SortedDictionary(Of String, String))

'set a variable to hold the text so I dont have to type it so much

Dim strDishText As String = txtAddDish.Text.ToLower

'if the infromation added does not match the regex pattern

If Regex.Match(strDishText, PATTERN).Success = False Then

'give the user the error message

MessageBox.Show(STRREGEXERROR, "Error")

'clear the text box

txtAddDish.Clear()

'if the entered information is already added

ElseIf gdicDishes.ContainsKey(strDishText) Then

'give the user the error message

MessageBox.Show(STRDUPLICATEERROR, "Error")

'clear the text box

txtAddDish.Clear()

Else

'if it does not fall into the two error catagories, then add it to the dictionary

gdicDishes.Add(strDishText, dicNewDish)

'clear the text box

txtAddDish.Clear()

'show the info to the user in the selected text box

lstDishes.Items.Add(strDishText.ToLower)

End If

End Sub

Private Sub btnRightRaw\_Click(sender As Object, e As EventArgs) Handles btnRightRaw.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- right arrow button next to the list of raw items.

'– When clicked, the selected item will be removed from the

' selected raw items list and dictionary. Proper error handling

' is also in effect.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- strSelected - a variable to hold the value of the text box

' so I dont have to type it so much

'------------------------------------------------------------

'set a variable to hold the selected item so I dont have to type it so much

Dim strSelected As String = lstSelectedRaw.SelectedItem

'if the user tries to click the right arrow without selecting an option

If lstSelectedRaw.SelectedIndex = -1 Then

'yell at the user to select an option

MessageBox.Show("Please select a prepped raw item.", "Error")

Else

'if they do select an option first

'locate the selected option

lstRaw.Items.Contains(strSelected)

'remove the selected option from the dictionary

gdicPreppedItems(lstPreppedItems.SelectedItem).Remove(strSelected)

'remove the selected option from the selected side

lstSelectedRaw.Items.Remove(strSelected)

End If

End Sub

Private Sub btnRightPrepped\_Click(sender As Object, e As EventArgs) Handles btnRightPrepped.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- right arrow button next to the list of prepped items.

'– When clicked, the selected item will be removed from the

' selected prepped items list and dictionary.

' Proper error handling is also in effect.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- strSelectedPrepped - a variable to hold the value of

' the text box so I dont have to type it so much

'------------------------------------------------------------

'set a variable to hold the selected item so I dont have to type it so much

Dim strSelectedPrepped As String = lstSelectedPrep.SelectedItem

'if the user tries to click the right arrow without selecting an option

If lstSelectedPrep.SelectedIndex = -1 Then

'yell at the user to select an option

MessageBox.Show("Please select a prepped item in the selected dish.", "Error")

Else

'if they do select an option first

'locate the selected option

lstPreppedItems.Items.Contains(strSelectedPrepped)

'remove the selected option from the dictionary

gdicDishes(lstDishes.SelectedItem).Remove(strSelectedPrepped)

'remove the selected option from the selected side

lstSelectedPrep.Items.Remove(strSelectedPrepped)

End If

End Sub

Private Sub btnLeftRaw\_Click(sender As Object, e As EventArgs) Handles btnLeftRaw.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- left arrow button next to the list of raw items.

'- The selected item(s) will be added to the new dictionary

'- which is linked to prepped item and be shown in the

'- list box on the left. Proper error handling is also being

'– used.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- arrTemp - Creating a temp array to hold the multiple

'- selected options from the list

'------------------------------------------------------------

'Creating a temp array to hold the multiple selected options from the list

Dim arrTemp(lstRaw.SelectedItems.Count - 1) As String

'looping through each of the selected options in the list

For i As Integer = 0 To arrTemp.Length - 1

'adding them to the temp array, and then showing the selected item in the selected list

arrTemp(i) = lstRaw.SelectedItems(i)

Next

'if the user did not select an option

If lstRaw.SelectedIndex = -1 Then

'yell at them to pick an option

MessageBox.Show("Please select a raw item.", "Error")

' if the user did select an option, but they picked one that was already added

ElseIf lstSelectedRaw.Items.Contains(lstRaw.SelectedItem) Then

'yell at them again with the global error message

MessageBox.Show(STRDUPLICATEERROR, "Error")

'if the user selected multiple options

ElseIf lstRaw.SelectedItems.Count > 1 Then

'but did not select a dish

If lstPreppedItems.SelectedIndex = -1 Then

'yell at them to pick a dish to add the prepped items too

MessageBox.Show("Please select a prepped item.", "Error")

Else

'loop through all of the items selected

For Each ingredient In arrTemp

'add each one to the global dictionary for dishes

gdicPreppedItems(lstPreppedItems.SelectedItem).Add(ingredient, gdicRawItems(ingredient))

'show the elements in the list box

lstSelectedRaw.Items.Add(ingredient)

Next

End If

Else

Try

'add the item to the global dictionary

gdicPreppedItems(lstPreppedItems.SelectedItem).Add(lstRaw.SelectedItem, gdicRawItems(lstRaw.SelectedItem))

'show the elements in the list box

lstSelectedRaw.Items.Add(lstRaw.SelectedItem)

'If the raw item is not attached to the prepped item, then yell at the user

Catch ex As Exception

MessageBox.Show("Raw item is not attached to a prepped item. Please select a prepped item.", "Error")

End Try

End If

End Sub

Private Sub btnLeftPrepped\_Click(sender As Object, e As EventArgs) Handles btnLeftPrepped.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user clicks the -

'- left arrow button next to the list of prepped items.

'- The selected item(s) will be added to the new dictionary

'- which is linked to dishes and be shown in the

'- list box on the left. Proper error handling is also being

'– used.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- arrTempPrep - Creating a temp array to hold the multiple

' selected options from the list

'------------------------------------------------------------

'Creating a temp array to hold the multiple selected options from the list

Dim arrTempPrep(lstPreppedItems.SelectedItems.Count - 1) As String

'looping through each of the selected options in the list

For i As Integer = 0 To arrTempPrep.Length - 1

'adding them to the temp array, and then showing the selected item in the selected list

arrTempPrep(i) = lstPreppedItems.SelectedItems(i)

Next

'if the user did not select an option

If lstPreppedItems.SelectedIndex = -1 Then

'yell at them to pick an option

MessageBox.Show("Please select a prepped item.", "Error")

' if the user did select an option, but they picked one that was already added

ElseIf lstSelectedPrep.Items.Contains(lstPreppedItems.SelectedItem) Then

'yell at them again with the global error message

MessageBox.Show(STRDUPLICATEERROR, "Error")

'if the user selected multiple options

ElseIf lstPreppedItems.SelectedItems.Count > 1 Then

'but did not select a dish

If lstDishes.SelectedIndex = -1 Then

'yell at them to pick a dish to add the prepped items too

MessageBox.Show("Please select a dish.", "Error")

Else

'loop through all of the items selected

For Each ingredient In arrTempPrep

'add each one to the global dictionary for dishes

gdicDishes(lstDishes.SelectedItem).Add(ingredient, gdicPreppedItems(ingredient))

'show the elements in the list box

lstSelectedPrep.Items.Add(ingredient)

Next

End If

Else

Try

'add the item to the global dictionary

gdicDishes(lstDishes.SelectedItem).Add(lstPreppedItems.SelectedItem, gdicPreppedItems(lstPreppedItems.SelectedItem))

'show the elements in the list box

lstSelectedPrep.Items.Add(lstPreppedItems.SelectedItem)

'If the prepped item is not attached to the dish, then yell at the user

Catch ex As Exception

MessageBox.Show("Prepped item is not attached to a dish. Please select a dish.", "Error")

End Try

End If

End Sub

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

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user selects an

'- option on the dishes box. Once selected, the dictionary

'- that holds the prepped items that belongs the specific

'- selected item is shown in the list box. This also has

'- proper error handling.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- dicPreparedDishes - Declaring a sorted dictionary that is

' going to hold the selected values

'------------------------------------------------------------

'Declaring a sorted dictionary that is going to hold the selected values

Dim dicPreparedDishes As SortedDictionary(Of String, SortedDictionary(Of String, String))

'Clearing the items that are in the prepped ingredients in selected prepped item

lstSelectedPrep.Items.Clear()

'Setting the index of the selected item to false

lstSelectedPrep.SelectedIndex = -1

'added a try catch to handle if the user accidentally clicks on a blank selection

Try

'set the prepared dictionary to the selected item in the option list

dicPreparedDishes = gdicDishes(lstDishes.SelectedItem)

'If the selected item does not have any ingredients attached (count = 0)

If dicPreparedDishes.Count = 0 Then

'handle the error with a message box

MessageBox.Show("That dish has no ingredients.", "Error")

'clear the text field

txtAddDish.Clear()

Else 'If the selected item does have any ingredients attached (count = not 0)

'loop through all of the attached ingredients

For Each item In dicPreparedDishes.Keys

'show the ingredients that are in the dicionary in the list box

lstSelectedPrep.Items.Add(item.ToLower)

Next

End If

Catch ex As Exception

'left intentionally blank because if the user does accidentally click on a blank space there really should be

'no error and the program should keep running

End Try

End Sub

Private Sub lstPreppedItems\_Click(sender As Object, e As EventArgs) Handles lstPreppedItems.Click

'------------------------------------------------------------

'- Subprogram Name: btnQuit\_Click -

'------------------------------------------------------------

'- Written By: Madison Kell -

'- Written On: February 11, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subroutine is called whenever the user selects an

'- option on the prepped box. Once selected, the dictionary

'- that holds the selecred raw items that belongs the specific

'- selected item is shown in the list box. This also has

'- proper error handling.

'------------------------------------------------------------

'- Parameter Dictionary (in parameter order): -

'- sender – Identifies which particular control raised the –

'- click event -

'- e – Holds the EventArgs object sent to the routine -

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- dicPreppedItem - Declaring a sorted dictionary that is

' going to hold the selected values

'------------------------------------------------------------

'Declaring a sorted dictionary that is going to hold the selected values

Dim dicPreppedItem As New SortedDictionary(Of String, String)

'Clearing the items that are in the raw ingredients in selected prepped item

lstSelectedRaw.Items.Clear()

'Setting the index of the selected item to false

lstSelectedRaw.SelectedIndex = -1

'added a try catch to handle if the user accidentally clicks on a blank selection

Try

'set the prepped item dictionary to the selected item in the option list

dicPreppedItem = gdicPreppedItems.Item(lstPreppedItems.SelectedItem)

'If the selected item does not have any ingredients attached (count = 0)

If dicPreppedItem.Count = 0 Then

'handle the error with a message box

MessageBox.Show("No Ingredients attached", "Error")

'clear the text field

txtAddDish.Clear()

'If the selected item does have any ingredients attached (count = not 0)

Else

'loop through all of the attached ingredients

For Each rawIngred In dicPreppedItem.Keys

'show the ingredients that are in the dicionary in the list box

lstSelectedRaw.Items.Add(rawIngred)

Next

End If

Catch ex As Exception

'left intentionally blank because if the user does accidentally click on a blank space there really should be

'no error and the program should keep running

End Try

End Sub

End Class













