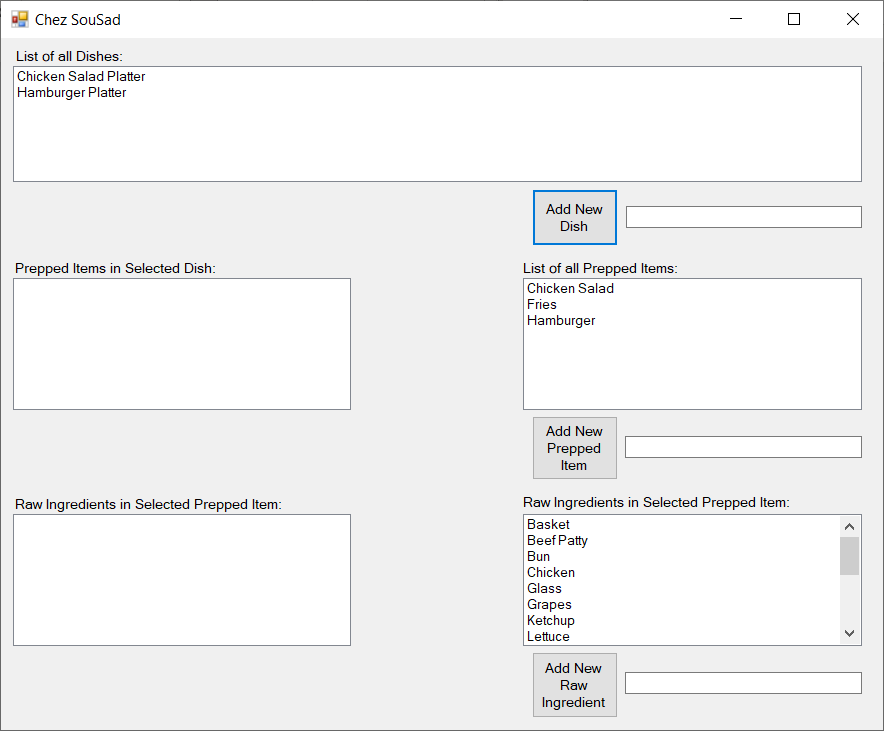
Program Cover Sheet

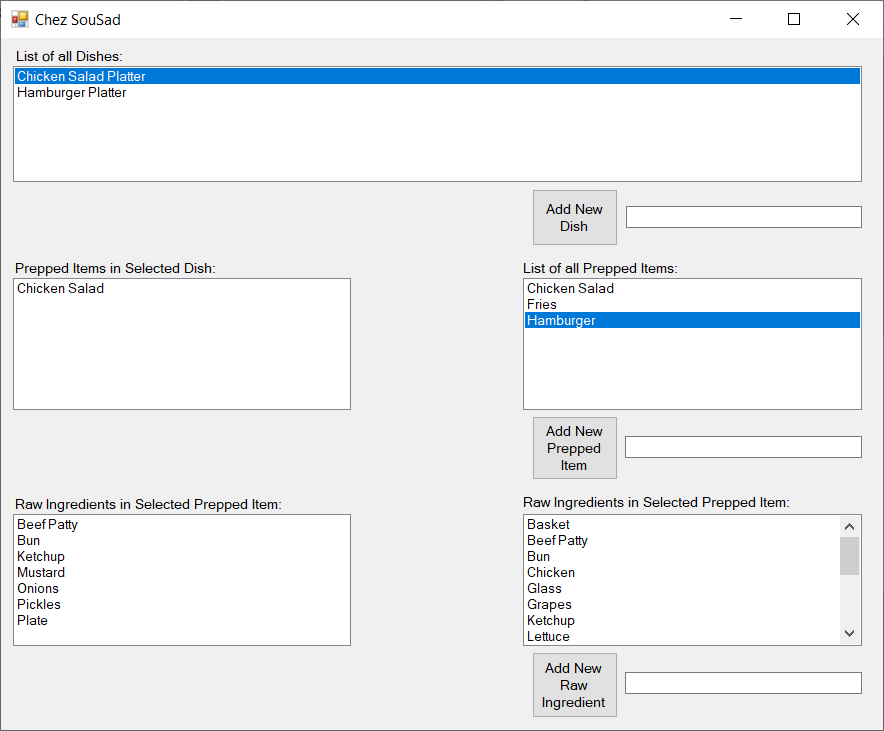
|  |
| --- |
| Name: Nathan VanSnepson |
| Assignment: Assignment 7 |
| List any parts of the assignment that do not work/were not completed: (None) |

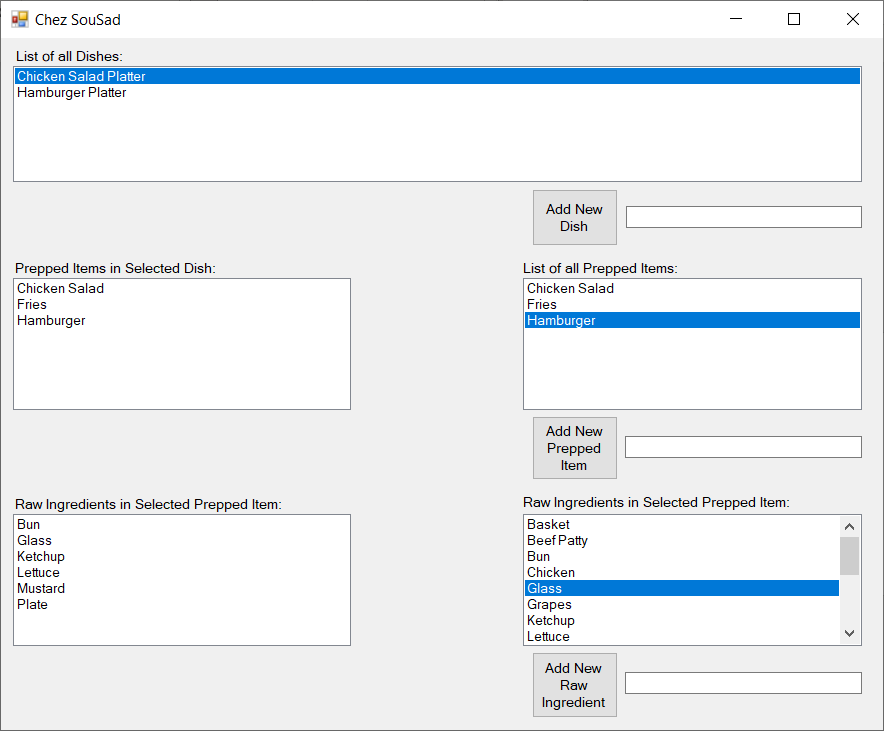
|  |
| --- |
| Instructor’s Comments: |
| Grade: |

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 : Form1.frm -

'- Part of Project: Assignment4 -

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

' Written On: February 16, 2022 -

' Written By: Nathan VanSnepson -

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

'- File Purpose: -

'- -

'- This file contains the main form for the entire applic- -

'- ation. It interacts with the user and displays dish -

'- information, prepped items information, and raw items -

'- information. It allows the user to connect raw items to -

'- prepped items and prepped items to dishes. -

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

'- Program Purpose: -

'- -

'- The purpose of this program is to display the breakdown -

'- of dishes, showing the prepped items and raw items that -

'- are attached. Also, it allows the user to build new -

'- dishes.

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

'- Global Variable Dictionary (alphabetically): -

'- dictDishes - Dictionary that has a key of String(Dish) -

'- and value of dictionary wich has a key of String(Prepped)-

'- and a value of dictionary which has a key of String(Raw) -

'- and a value of String(Raw) -

'- dictPrepped - Dictionary that has a key of String(Prepped-

'- ) and a value of dictionary that has key of String(Raw) -

'- and a value of String(Raw) -

'- dictRaw - Dictionary that has a key of String(Raw) and a -

'- value of String(Raw) -

'- dictTemp - Dictionary holding value to be added to an -

'- existing dictionary. This is used for drag and drop -

'- udtUpdatedListView - Enum used to control what draganddr--

'- op is being used. -

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

Public Class Form1

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

'--- GLOBAL CONSTANTS --- GLOBAL CONSTANTS --- GLOBAL CONSTANTS --- GLOBAL CONSTANTS ---

'--- GLOBAL CONSTANTS --- GLOBAL CONSTANTS --- GLOBAL CONSTANTS --- GLOBAL CONSTANTS ---

'--- GLOBAL CONSTANTS --- GLOBAL CONSTANTS --- GLOBAL CONSTANTS --- GLOBAL CONSTANTS ---

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

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

'--- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES ---

'--- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES ---

'--- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES ---

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

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

'----- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS ----

'----- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS ----

'----- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS --- GLOBAL ENUMS ----

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

'Enum for drag and drop action being done

Enum updatedListView

AddPrepped

RemovePrepped

AddRaw

RemoveRaw

End Enum

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

'--- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES ---

'--- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES ---

'--- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES ---

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

Dim dictDishes As New SortedDictionary(Of String, SortedDictionary(Of String, SortedDictionary(Of String, String)))(StringComparer.InvariantCultureIgnoreCase)

Dim dictPrepped As New SortedDictionary(Of String, SortedDictionary(Of String, String))(StringComparer.InvariantCultureIgnoreCase)

Dim dictRaw As New SortedDictionary(Of String, String)(StringComparer.InvariantCultureIgnoreCase)

Dim dictTemp As New Dictionary(Of String, Object)(StringComparer.InvariantCultureIgnoreCase)

Dim udtUpdatedListView As updatedListView

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

'--- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS ---

'--- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS ---

'--- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS ---

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

Private Sub populateDictionaries()

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

'- Subprogram Name: populateDictionaries -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to prepopulate the -

'- dictionaries.

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

'- Parameter Dictionary (in parameter order): -

'- (none) -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Put default ingredients into dictRaw

dictRaw.Add("Basket", "Basket")

dictRaw.Add("Beef Patty", "Beef Patty")

dictRaw.Add("Bun", "Bun")

dictRaw.Add("Chicken", "Chicken")

dictRaw.Add("Glass", "Glass")

dictRaw.Add("Grapes", "Grapes")

dictRaw.Add("Ketchup", "Ketchup")

dictRaw.Add("Lettuce", "Lettuce")

dictRaw.Add("Mayonnaise", "Mayonnaise")

dictRaw.Add("Mustard", "Mustard")

dictRaw.Add("Oil", "Oil")

dictRaw.Add("Onions", "Onions")

dictRaw.Add("Pickles", "Pickles")

dictRaw.Add("Plate", "Plate")

dictRaw.Add("Potatoes", "Potatoes")

dictRaw.Add("Sugar", "Sugar")

dictRaw.Add("Syrup", "Syrup")

dictRaw.Add("Water", "Water")

'Insert default prepped items into dictPrepped

dictPrepped.Add("Hamburger", New SortedDictionary(Of String, String)(StringComparer.InvariantCultureIgnoreCase))

dictPrepped.Item("Hamburger").Add("Beef Patty", "Beef Patty")

dictPrepped.Item("Hamburger").Add("Bun", "Bun")

dictPrepped.Item("Hamburger").Add("Ketchup", "Ketchup")

dictPrepped.Item("Hamburger").Add("Mustard", "Mustard")

dictPrepped.Item("Hamburger").Add("Onions", "Onions")

dictPrepped.Item("Hamburger").Add("Pickles", "Pickles")

dictPrepped.Item("Hamburger").Add("Plate", "Plate")

dictPrepped.Add("Fries", New SortedDictionary(Of String, String)(StringComparer.InvariantCultureIgnoreCase))

dictPrepped.Item("Fries").Add("Potatoes", "Potatoes")

dictPrepped.Item("Fries").Add("Basket", "Basket")

dictPrepped.Add("Chicken Salad", New SortedDictionary(Of String, String)(StringComparer.InvariantCultureIgnoreCase))

dictPrepped.Item("Chicken Salad").Add("Chicken", "Chicken")

dictPrepped.Item("Chicken Salad").Add("Lettuce", "Lettuce")

'Insert default dishes into dictDishes

dictDishes.Add("Chicken Salad Platter", New SortedDictionary(Of String, SortedDictionary(Of String, String))(StringComparer.InvariantCultureIgnoreCase))

dictDishes.Item("Chicken Salad Platter").Add("Chicken Salad", dictPrepped.Item("Chicken Salad"))

dictDishes.Add("Hamburger Platter", New SortedDictionary(Of String, SortedDictionary(Of String, String))(StringComparer.InvariantCultureIgnoreCase))

dictDishes.Item("Hamburger Platter").Add("Hamburger", dictPrepped.Item("Hamburger"))

dictDishes.Item("Hamburger Platter").Add("Fries", dictPrepped.Item("Fries"))

End Sub

Private Sub populateRaw()

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

'- Subprogram Name: populateRaw -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate listbox -

'- lstRaw with all the raw items in dictRaw -

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

'- Parameter Dictionary (in parameter order): -

'- (none) -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears the listbox

lstRaw.Items.Clear()

'Loops through raw items to fill listbox

For Each raw As String In dictRaw.Keys

lstRaw.Items.Add(raw)

Next

End Sub

Private Sub populatePrepped()

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

'- Subprogram Name: populatePrepped -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate listbox -

'- lstPrepped with all the prepped items in dictPrepped -

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

'- Parameter Dictionary (in parameter order): -

'- (none) -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears the listbox

lstPrepped.Items.Clear()

'Loops through prepped items to fill listbox

For Each prepped As String In dictPrepped.Keys

lstPrepped.Items.Add(prepped)

Next

End Sub

Private Sub populateDishes()

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

'- Subprogram Name: populateDishes -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate listbox -

'- lstDishes with all the dishes in dictDishes -

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

'- Parameter Dictionary (in parameter order): -

'- (none) -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears the listbox

lstDishes.Items.Clear()

'Loops through dishes to fill listbox

For Each dishes As String In dictDishes.Keys

lstDishes.Items.Add(dishes)

Next

End Sub

Private Sub populateSelectedRaw(preppedItem As String)

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

'- Subprogram Name: populateSelectedRaw -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate listbox -

'- lstSelectedRaw with all the raw items associated with a -

'- given prepped item. -

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

'- Parameter Dictionary (in parameter order): -

'- preppedItem - A prepped item that comes from dictPrepped -

'- to access its raw items -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears the listbox

lstSelectedRaw.Items.Clear()

'Loops through raw items for a prepped item to fill listbox

For Each rawItem As String In dictPrepped.Item(preppedItem).Keys

lstSelectedRaw.Items.Add(rawItem)

Next

End Sub

Private Sub populateSelectedPrepped(dish As String)

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

'- Subprogram Name: populateSelectedPrepped -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate listbox -

'- lstSelectedPrepped with all the prepped items associated -

'- with a given dish. -

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

'- Parameter Dictionary (in parameter order): -

'- dish - A dish that comes from dictDishes to access its -

'- prepped items -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears the listbox

lstSelectedPrepped.Items.Clear()

'Loops through raw items for a prepped item to fill listbox

For Each preppedItem As String In dictDishes.Item(dish).Keys

lstSelectedPrepped.Items.Add(preppedItem)

Next

End Sub

Private Sub Form1\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

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

'- Subprogram Name: Form1\_Load -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to prepopulate all the -

'- dictionaries, then populate the listboxes that show all -

'- the associated items. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Prepopulates dictionaries

populateDictionaries()

'Populates lstRaw

populateRaw()

'Populates lstPrepped

populatePrepped()

'Populates lstDishes

populateDishes()

'//New

lstSelectedPrepped.AllowDrop = True

End Sub

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

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

'- Subprogram Name: btnAddRaw\_Click -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to add new raw items to-

'- dictRaw and repopulate lstRaw. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears error, removes it from screen

erpAddRaw.Clear()

'Checks for valid raw item entry

If dictRaw.ContainsKey(txtAddRaw.Text) Or txtAddRaw.Text.Length = 0 Then

'displays error to use if invalid entry

erpAddRaw.SetError(txtAddRaw, "Raw ingredient already exists or is empty")

Else

'add raw item to dictRaw

dictRaw.Add(txtAddRaw.Text, txtAddRaw.Text)

'repopulate lstRaw

populateRaw()

End If

End Sub

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

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

'- Subprogram Name: btnAddPrepped\_Click -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to add new prepped -

'- items to dictPrepped and repopulate lstPrepped -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears error, removes it from screen

erpAddPrepped.Clear()

'Checks for valid prepped item entry

If dictPrepped.ContainsKey(txtAddPrepped.Text) Or txtAddPrepped.Text.Length = 0 Then

'displays error to use if invalid entry

erpAddPrepped.SetError(txtAddPrepped, "Raw ingredient already exists or is empty")

Else

'add prepped item to dictPrepped

dictPrepped.Add(txtAddPrepped.Text, New SortedDictionary(Of String, String)(StringComparer.InvariantCultureIgnoreCase))

'repopulate lstPrepped

populatePrepped()

End If

End Sub

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

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

'- Subprogram Name: btnAddDish\_Click -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to add new dishes to -

'- dictDishes and repopulate lstDishes -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Clears error, removes it from screen

erpAddDish.Clear()

'Checks for valid prepped item entry

If dictDishes.ContainsKey(txtAddDish.Text) Or txtAddDish.Text.Length = 0 Then

'displays error to use if invalid entry

erpAddDish.SetError(txtAddDish, "Raw ingredient already exists or is empty")

Else

'add dish to dictDishes

dictDishes.Add(txtAddDish.Text, New SortedDictionary(Of String, SortedDictionary(Of String, String))(StringComparer.InvariantCultureIgnoreCase))

'repopulate lstDishes

populateDishes()

End If

End Sub

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

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

'- Subprogram Name: lstPrepped\_SelectedIndexChanged -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate -

'- lstSelectedRaw with the associated raw items for a -

'- selected prepped item -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'If there is 1 prepped item selected in lstPrepped

If lstPrepped.SelectedItems.Count = 1 Then

'Populate lstSelectedRaw with associated raw items

populateSelectedRaw(lstPrepped.SelectedItem.ToString())

End If

End Sub

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

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

'- Subprogram Name: lstDishes\_SelectedIndexChanged -

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

'- Written By: Nathan VanSnepson -

'- Written On: February 16, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to populate -

'- lstSelectedprepped with the associated prepped items for -

'- a selected dish -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'If there is 1 dish selected

If lstDishes.SelectedIndex > -1 Then

'Populate lstSelectedPrepped with associated prepped items

populateSelectedPrepped(lstDishes.SelectedItem.ToString())

End If

End Sub

Private Sub lstPrepped\_MouseMove(sender As Object, e As MouseEventArgs) Handles lstPrepped.MouseMove

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

'- Subprogram Name: lstPrepped\_MouseMove -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is initiate the process of-

'- drag and drop. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- dropEffect - defines the dropEffect displayed -

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

Dim dropEffect As DragDropEffects 'Instantiates a new DragDropEffects

dictTemp = New Dictionary(Of String, Object) 'Sets dictTemp to a new value

'If the left mouse button was selected and there is a dish selected

If e.Button = MouseButtons.Left And lstDishes.SelectedItems.Count > 0 Then

'Set the value of the list view being updated

udtUpdatedListView = updatedListView.AddPrepped

'Add item to dictionary

dictTemp.Add(lstPrepped.SelectedItem.ToString(), dictPrepped.Values(lstPrepped.SelectedIndex()))

'set dropEffect to a new DoDragDrop event

dropEffect = lstSelectedPrepped.DoDragDrop(lstPrepped.SelectedItem.ToString(), DragDropEffects.All Or DragDropEffects.Link)

End If

End Sub

Private Sub lstSelectedPrepped\_DragEnter(sender As Object, e As DragEventArgs) Handles lstSelectedPrepped.DragEnter

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

'- Subprogram Name: lstSelectedPrepped\_DragEnter -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to watch for when an -

'- object is dragged into the control -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

e.Effect = DragDropEffects.Copy

End Sub

Private Sub lstSelectedPrepped\_DragDrop(sender As Object, e As DragEventArgs) Handles lstSelectedPrepped.DragDrop

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

'- Subprogram Name: lstSelectedPrepped\_DragDrop -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to update prepped items-

'- for a selected dish. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Checks if the drag drop is being started from the correct control

If udtUpdatedListView = updatedListView.AddPrepped Then

'Check if the key does not exits

If Not dictDishes.Item(lstDishes.SelectedItem.ToString()).ContainsKey(e.Data.GetData("Text").ToString()) Then

'add item to dishes

dictDishes.Item(lstDishes.SelectedItem.ToString()).Add(e.Data.GetData("Text").ToString(), dictPrepped.Item(e.Data.GetData("Text").ToString()))

'Repopulate lstSelectedPrepped

populateSelectedPrepped(lstDishes.SelectedItem.ToString())

End If

End If

End Sub

Private Sub lstSelectedPrepped\_MouseMove(sender As Object, e As MouseEventArgs) Handles lstSelectedPrepped.MouseMove

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

'- Subprogram Name: lstSelectedPrepped\_MouseMove -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is initiate the process of-

'- drag and drop. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- dropEffect - defines the dropEffect displayed -

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

Dim dropEffect As DragDropEffects 'Instantiates a new DragDropEffects

dictTemp = New Dictionary(Of String, Object) 'Sets dictTemp to a new value

'If the left mouse button was selected and there is a dish selected

If e.Button = MouseButtons.Left And lstDishes.SelectedItems.Count > 0 Then

'Set the value of the list view being removed

udtUpdatedListView = updatedListView.RemovePrepped

'set dropEffect to a new DoDragDrop event

dropEffect = lstPrepped.DoDragDrop(lstSelectedPrepped.SelectedItem.ToString(), DragDropEffects.All Or DragDropEffects.Link)

End If

End Sub

Private Sub lstPrepped\_DragEnter(sender As Object, e As DragEventArgs) Handles lstPrepped.DragEnter

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

'- Subprogram Name: lstPrepped\_DragEnter -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to watch for when an -

'- object is dragged into the control -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

e.Effect = DragDropEffects.Copy

End Sub

Private Sub lstPrepped\_DragDrop(sender As Object, e As DragEventArgs) Handles lstPrepped.DragDrop

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

'- Subprogram Name: lstPrepped\_DragDrop -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to update prepped items-

'- for a selected dish. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Checks if the drag drop is being started from the correct control

If udtUpdatedListView = updatedListView.RemovePrepped Then

'Check if the key does not exits

If dictDishes.Item(lstDishes.SelectedItem.ToString()).ContainsKey(e.Data.GetData("Text").ToString()) Then

'Removes prepped item from dish

dictDishes.Item(lstDishes.SelectedItem.ToString()).Remove(e.Data.GetData("Text").ToString())

'Repopulate lstSelectedPrepped

populateSelectedPrepped(lstDishes.SelectedItem.ToString())

End If

End If

End Sub

Private Sub lstRaw\_MouseMove(sender As Object, e As MouseEventArgs) Handles lstRaw.MouseMove

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

'- Subprogram Name: lstRaw\_MouseMove -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is initiate the process of-

'- drag and drop. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- dropEffect - defines the dropEffect displayed -

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

Dim dropEffect As DragDropEffects 'Instantiates a new DragDropEffects

'If the left mouse button was selected and there is a prepped item selected

If e.Button = MouseButtons.Left And lstPrepped.SelectedItems.Count > 0 Then

'Set the value of the list view being added

udtUpdatedListView = updatedListView.AddRaw

'set dropEffect to a new DoDragDrop event

dropEffect = lstSelectedRaw.DoDragDrop(lstRaw.SelectedItem.ToString(), DragDropEffects.All Or DragDropEffects.Link)

End If

End Sub

Private Sub lstSelectedRaw\_DragEnter(sender As Object, e As DragEventArgs) Handles lstSelectedRaw.DragEnter

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

'- Subprogram Name: lstSelectedRaw\_DragEnter -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to watch for when an -

'- object is dragged into the control -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

e.Effect = DragDropEffects.Copy

End Sub

Private Sub lstSelectedRaw\_DragDrop(sender As Object, e As DragEventArgs) Handles lstSelectedRaw.DragDrop

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

'- Subprogram Name: lstSelectedRaw\_DragDrop -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to update prepped items-

'- for a selected dish. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Checks if the drag drop is being started from the correct control

If udtUpdatedListView = updatedListView.AddRaw Then

'Checks if the key does not exist

If Not dictPrepped.Item(lstPrepped.SelectedItem.ToString()).ContainsKey(e.Data.GetData("Text").ToString()) Then

'Add item to prepped item

dictPrepped.Item(lstPrepped.SelectedItem.ToString()).Add(e.Data.GetData("Text").ToString(), e.Data.GetData("Text").ToString())

'Repopulate lstSelectedRaw

populateSelectedRaw(lstPrepped.SelectedItem.ToString())

End If

End If

End Sub

Private Sub lstSelectedRaw\_MouseMove(sender As Object, e As MouseEventArgs) Handles lstSelectedRaw.MouseMove

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

'- Subprogram Name: lstSelectedRaw\_MouseMove -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is initiate the process of-

'- drag and drop. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- dropEffect - defines the dropEffect displayed -

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

Dim dropEffect As DragDropEffects 'Instantiates a new DragDropEffects

'If the left mouse button was selected and there is a prepped item selected

If e.Button = MouseButtons.Left And lstPrepped.SelectedItems.Count > 0 Then

'Set the value of the list view being removed

udtUpdatedListView = updatedListView.RemoveRaw

'set dropEffect to a new DoDragDrop event

dropEffect = lstRaw.DoDragDrop(lstSelectedRaw.SelectedItem.ToString(), DragDropEffects.All Or DragDropEffects.Link)

End If

End Sub

Private Sub lstRaw\_DragEnter(sender As Object, e As DragEventArgs) Handles lstRaw.DragEnter

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

'- Subprogram Name: lstRaw\_DragEnter -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to watch for when an -

'- object is dragged into the control -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

e.Effect = DragDropEffects.Copy

End Sub

Private Sub lstRaw\_DragDrop(sender As Object, e As DragEventArgs) Handles lstRaw.DragDrop

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

'- Subprogram Name: lstSelectedRaw\_DragDrop -

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

'- Written By: Nathan VanSnepson -

'- Written On: March 28, 2022 -

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

'- Subprogram Purpose: -

'- The purpose of this subprogram is to update prepped items-

'- for a selected dish. -

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

'- Parameter Dictionary (in parameter order): -

'- sender - reference to the object that raised the event -

'- e - reference to EventArgs class that contains any addi- -

'- tional information about the event. -

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

'- Local Variable Dictionary (alphabetically): -

'- (None) -

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

'Checks if the drag drop is being started from the correct control

If udtUpdatedListView = updatedListView.RemoveRaw Then

'Checks if the item is in the prepped item

If dictPrepped.Item(lstPrepped.SelectedItem.ToString()).ContainsKey(e.Data.GetData("Text").ToString()) Then

'removes item from prepped item

dictPrepped.Item(lstPrepped.SelectedItem.ToString()).Remove(e.Data.GetData("Text").ToString())

'Repopulate lstSelectedRaw

populateSelectedRaw(lstPrepped.SelectedItem.ToString())

End If

End If

End Sub

End Class