**NJ Custom Chair Company**

Nam Jun Lee

Washington State University

MIS 325

December 14, 2021

**Background:**

A manager of a custom chair company called NJ has requested a prototype to replace the current transaction recording system. The company has stores in five states in the United States. The company's main product has four types of chairs and customizes the materials, colors, and additional options of the chairs as customers want.

In order to motivate employees, the CEO of NJ created a system that provides incentives when sales of each store achieve $50,000 every month. In addition, for the new lineup, the manager had been asked to provide specific information on the company's sales as to which chairs had the most impact on sales. However, the transaction recording system currently used by the manager can only check customer information and total sales, so the manager said a more detailed transaction recording system is needed.

The NJ manager asked to make it easier to identify which store achieved this goal and which store had the most impact on total sales through sales at each store. It also asked what kind of chairs were sold a lot to introduce new products, what materials customers want for each chair, what colors are popular, and how many of the people who bought the chair signed up for the NJ membership.

To solve this problem, the program is automatically updated to the transaction history data table when the manager enters the name of the customer who purchased the chair, store area, chair type and material, color, additional options, number of purchased chairs, and membership status. There are five summary tables, including color, chair material, chair type, each store, and total sales. It also has the ability to automatically update information to make it easier to understand by visualizing it through graphs.

As such, it is difficult for a manager to check the total sales of a company and sales of each store when operating a canteen in several regions. Above all, if you want to know only specific information (type, color, store, etc.) for each subject, you have to check the transaction records one by one and do statistics. So, the program is designed to help managers of companies with multiple stores release statistics.

**Overview of the functionality:**

Three multi-views were used to reduce screen confusion on this website. First, there is a text box that requires accessing the website and then entering a password. When entering a password and pressing the click button, output an error message if the password is wrong through the label. If the password is entered correctly, output a welcome message and show three link buttons.

**1. Invoice (first page)**

This page requires the manager to enter the input control and list control of the necessary information to store transaction records. There are two input controls, five list controls, and a check box.

* Text box for entering the customer's name.
* List of store area radio buttons: These include five states (California, New York, Oregon, Washington, Lowa). If the manager selects one store, it shows the flag image for each state (using select case). In addition, since the tax rates vary from state to state, investigate the tax rates and indicate where information was obtained. (using hyper link)
* Chair type radio button list: 4 chairs are included (Wingback, Club, Folding, Sofa). If the manager selects one chair type, it shows an image of each chair and displays the basic price of the chair.
* Chair material radio button list: There are a total of three materials (Fabric, Leather, Mesh). Since the price of each chair material is inevitably different, there are a total of 12 items in the radio button list (4 types \* 3 materials). If the manager selects a material for each chair type, each material image is shown, and the added material price is shown. However, if the material of each chair type is not the same, it shows an error message (using select case).
* Chair Color Radio button list: 6 colors exist (black, gray, burgundy, beige, white, brown). Each color image is shown if the manager selects one color (using select case). There is no additional amount for the color.
* Additional Options Checkbox List & Additional Options Text Box: There are 5 options (wheel, mass function, tilting, automatic height setting function). If an option is selected using the for statement, it shows the option name and total option price added to the additional option text.
* Text box for entering the number of chairs.
* Check box for membership.

Invoice button: Pressing this button displays a row of data in the gridview of the ASP.NET web page after the record is stored in the data structure. However, if nothing is entered in the customer’s name input box or the number of chairs input box, highlight each box border and display an error message in the resulting text box (if-else). In the case of the number of chairs, if a value other than a number is included, an error message is displayed. In the case of a store area, chair type, and chair-based radio button list, if not one is selected, each list border is emphasized, and an error message is displayed in the resulting text box (if-else). In addition, in the case of a radio button list of chair materials, an error message is displayed if the material for each chair type is not correct. Additional options, membership check boxes, are factors that do not matter if you do not select them.

Clear button: Pressing this button initializes each list control, input control, and everything displayed in the check box to be entered (blank).

In this way, when all records are written in the right format, clicking on the invoice button shows a label that increases the transaction record and stores each new transaction record in a new row in the data structure.

The transaction history data table has a total of 16 columns (Transaction ID, Order date, Customer name, Store location, Chair Color, Chair Cost, Extra Cost, Quantity, Total cost of chair, Member, Discount price, Tax rate, Tax price, Total cost before tax, Total cost). It starts with an automatic number field called Transaction ID, where the data type is an integer.

**2. Summaries (second page)**

Five summary data tables are included. Use a data storage approach when a new transaction record of data is added to the transaction table (transaction history) and update the cumulative sum or metric column of some dimensional tables.

* Color: Indicate the number of chairs sold by color.
* Chair material: Set the radio button list control for chair Material. Import text information into columns in the summary data table (12 in total). Here, the number of chairs sold for each chair material, the total price, and the number of options added are included. This data table further subdivides the chair type data table and shows the results.
* Chair type: Set the chair type radio button list control. Import text information into columns in the summary data table (4 in total). It has the same row as the chair material data table.
* Store: Set the store area radio button list control. Import text information into columns in the summary data table (5 in total). This shows the current status of each region's tax rate, pre-tax total sales, post-tax total sales, number of distributors, number of members, designated target price ($ 50000), target price remaining until target, achievement rate, and string variable. The current status remaining until the goal is divided into a total of 4 sections using the select case (0 to 30, 30.1 to 60, 60.1 to 99.9, 1 to 199.9, more 2).
* Grand total: It holds total measurements (Total customers, Total cost, Total quantity, Member count).

**3. Visualization (third page)**

There is a total of three graphs here.

* Number of sold by chair type: Bring up the chair type data table and set the x-axis to Chair type and the y-axis to Quantity.
* Total sales by store: Bring up the store data table and set the x-axis to Store and y-axis to Total Cost.
* The number of sold by chair color: The color data table is retrieved, and x value is set to Color Type and y value is Quantity to show the results as a pie chart.

**Aspx.vb code:**

Imports System.Data

Partial Class FinalProject\_FinalProject

Inherits System.Web.UI.Page

'The project was created by managers to check sales of each store at a glance, and it shows total sales and which products for each chair type had more impact on sales and which stores had the most impact on sales.

'This global variable tables is used as a data structure to store transaction data rows and 5 summary data tables

Public Shared dtTransaction As New DataTable

Public Shared dtChairType As New DataTable

Public Shared dtChairInd As New DataTable

Public Shared dtStore As New DataTable

Public Shared dtGrandTotal As New DataTable

Public Shared dtColor As New DataTable

'two global integer variables updated to keep a total of member and count of sold quantity of chair

Public Shared soldAll, member As Integer

'two global decimal variables updated to keep total include tax, total not include tax

Public Shared taxAdd, taxPre As Decimal

#Region "Set up dattable for data storage"

Private Sub FinalProject\_FinalProject\_Init(sender As Object, e As EventArgs) Handles Me.Init

'if columns of either table were already created then exit, do not create them again

If dtTransaction.Columns.Count > 0 OrElse dtChairType.Columns.Count > 0 OrElse dtChairInd.Columns.Count > 0 OrElse dtStore.Columns.Count > 0 OrElse dtGrandTotal.Columns.Count > 0 OrElse dtColor.Columns.Count > 0 Then Exit Sub

'This is an operation in which columns are added to the global data table. Names and data types for each column are specified. The first column is defined as an integer and an automatic number. There are 16 additional columns

With dtTransaction.Columns

.Add("TransactionID", GetType(Integer))

.Add("OrderDate", GetType(String))

.Add("CustomerName", GetType(String))

.Add("StoreLocation", GetType(String))

.Add("Chair", GetType(String))

.Add("Color", GetType(String))

.Add("ChairCost", GetType(Decimal))

.Add("ExtraCost", GetType(Decimal))

.Add("Quantity", GetType(Decimal))

.Add("TotalChairCost", GetType(Decimal))

.Add("Member", GetType(String))

.Add("DiscountPrice", GetType(Decimal))

.Add("%TaxRate", GetType(Decimal))

.Add("TaxPrice", GetType(Decimal))

.Add("TotalCostPreTax", GetType(Decimal))

.Add("TotalCost", GetType(Decimal))

End With

'The first column is defined as an integer and an automatic number

With dtTransaction.Columns("TransactionID")

.AutoIncrement = True

.AutoIncrementSeed = 1

.AutoIncrementStep = 1

End With

'This is an operation in which columns are added to the global summary data table. Names and data types for each column are specified. There are 4 additional columns

With dtChairType.Columns

.Add("ChairType", GetType(String))

.Add("Quantity", GetType(Decimal))

.Add("ExtraQuantity", GetType(Decimal))

.Add("TotalCost", GetType(Decimal))

End With

'if calculations are going to be made in columns (such as running totals) then the values need to start at 0

With dtChairType

.Columns("Quantity").DefaultValue = 0

.Columns("ExtraQuantity").DefaultValue = 0

.Columns("TotalCost").DefaultValue = 0

End With

'row to store data in type of each chair data table (4 rows)

For Each typeT As ListItem In typeRadioButtonList2.Items

Dim drT As DataRow = dtChairType.NewRow

drT.Item("ChairType") = typeT.Text

dtChairType.Rows.Add(drT)

Next

'This is an operation in which columns are added to the global summary data table. Names and data types for each column are specified. There are 4 additional columns

With dtChairInd.Columns

.Add("ChairMaterial", GetType(String))

.Add("Quantity", GetType(Decimal))

.Add("ExtraQuantity", GetType(Decimal))

.Add("TotalCost", GetType(Decimal))

End With

'if calculations are going to be made in columns (such as running totals) then the values need to start at 0

With dtChairInd

.Columns("Quantity").DefaultValue = 0

.Columns("ExtraQuantity").DefaultValue = 0

.Columns("TotalCost").DefaultValue = 0

End With

'row to store data in material of each chair data table (12 rows)

For Each typeM As ListItem In materialRadioButtonList3.Items

Dim drM As DataRow = dtChairInd.NewRow

drM.Item("ChairMaterial") = typeM.Text

dtChairInd.Rows.Add(drM)

Next

'This is an operation in which columns are added to the global summary data table. Names and data types for each column are specified. There are 10 additional columns.

With dtStore.Columns

.Add("Store", GetType(String))

.Add("%TaxRate", GetType(Decimal))

.Add("TotalCostPreTax", GetType(Decimal))

.Add("TotalCost", GetType(Decimal))

.Add("TotalQuantity", GetType(Decimal))

.Add("MemberCount", GetType(Decimal))

.Add("Goal", GetType(Decimal))

.Add("RemainGoal", GetType(Decimal))

.Add("%Goal", GetType(String))

.Add("StatusOfGoal", GetType(String))

End With

'if calculations are going to be made in columns (such as running totals) then the values need to start at 0

With dtStore

.Columns("TotalCostPreTax").DefaultValue = 0

.Columns("TotalCost").DefaultValue = 0

.Columns("TotalQuantity").DefaultValue = 0

.Columns("MemberCount").DefaultValue = 0

.Columns("%Goal").DefaultValue = 0

.Columns("RemainGoal").DefaultValue = 0

.Columns("StatusOfGoal").DefaultValue = 0

End With

'row to store data in type of each store data table (5 rows)

For Each typeStore As ListItem In storeRadioButtonList1.Items

Dim drS As DataRow = dtStore.NewRow

drS.Item("Store") = typeStore.Text

dtStore.Rows.Add(drS)

Next

'This is an operation in which columns are added to the global summary data table. Names and data types for each column are specified. There are 2 additional columns.

With dtColor.Columns

.Add("ColorType", GetType(String))

.Add("Quantity", GetType(Decimal))

End With

'if calculations are going to be made in columns (such as running totals) then the values need to start at 0

With dtColor

.Columns("Quantity").DefaultValue = 0

End With

'row to store data in type of each chair color data table (6 rows)

For Each typeColor As ListItem In colorRadioButtonList.Items

Dim drC As DataRow = dtColor.NewRow

drC.Item("ColorType") = typeColor.Text

dtColor.Rows.Add(drC)

Next

'This is an operation in which columns are added to the global summary data table. Names and data types for each column are specified. There are 4 additional columns

With dtGrandTotal.Columns

.Add("TotalCustomers", GetType(Decimal))

.Add("TotalCost", GetType(Decimal))

.Add("TotalQuantity", GetType(Decimal))

.Add("MemberCount", GetType(Decimal))

End With

'if calculations are going to be made in columns (such as running totals) then the values need to start at 0

With dtGrandTotal

.Columns("TotalCustomers").DefaultValue = 0

.Columns("TotalCost").DefaultValue = 0

.Columns("TotalQuantity").DefaultValue = 0

.Columns("MemberCount").DefaultValue = 0

End With

'row to store data in grand total data table only one

Dim drG As DataRow = dtGrandTotal.NewRow

dtGrandTotal.Rows.Add(drG)

'used to see if the arrays are created and available (5 summary data tables; nothing recorded)

With colorGridView

.DataSource = dtColor

.DataBind()

End With

With chairTypeGridView

.DataSource = dtChairType

.DataBind()

End With

With chairGridView

.DataSource = dtChairInd

.DataBind()

End With

With totalGridView

.DataSource = dtGrandTotal

.DataBind()

End With

With storeGridView

.DataSource = dtStore

.DataBind()

End With

End Sub

#End Region

#Region "Security"

Protected Sub ImageButton1\_Click(sender As Object, e As ImageClickEventArgs) Handles ImageButton1.Click

'This functions to allow the program to run only when a correct password is entered

MaintainScrollPositionOnPostBack = True

If passwordTextBox1.Text = "0000" Then 'If you enter the password correctly, it shows linkbutton 1 and link buttion 2 and linkbutton 3 and welcome label

LinkButton1.Visible = True

LinkButton2.Visible = True

LinkButton3.Visible = True

passwordLabel1.Visible = False

welcomeLabel.Visible = True

MultiView1.Visible = True

Else 'If you enter the password incorrectly, it shows that error message

passwordLabel1.Visible = True

passwordLabel1.Text = "Please enter the correct number!!!"

welcomeLabel.Visible = False

End If

End Sub

Protected Sub LinkButton1\_Click(sender As Object, e As EventArgs) Handles LinkButton1.Click

'these next procedures allow the human to switch between sections of the program first link button

MultiView1.ActiveViewIndex = 0

End Sub

Protected Sub LinkButton2\_Click(sender As Object, e As EventArgs) Handles LinkButton2.Click

'these next procedures allow the human to switch between sections of the program second link button

MultiView1.ActiveViewIndex = 1

End Sub

Protected Sub LinkButton3\_Click(sender As Object, e As EventArgs) Handles LinkButton3.Click

'these next procedures allow the human to switch between sections of the program third link button

MultiView1.ActiveViewIndex = 2

DrawCharts() 'when the program user switches to view2 - refresh the charts which should be quick since the data is already pre-aggregated

End Sub

#End Region

#Region "Set index changed"

Protected Sub storeRadioButtonList1\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles storeRadioButtonList1.SelectedIndexChanged

'Image control in which the picture changes depending on the selected store

MaintainScrollPositionOnPostBack = True

Select Case storeRadioButtonList1.SelectedIndex

Case 0 'if you choose california then show the california flag image

storeImage.ImageUrl = "california.png"

storeImage.Visible = True

Case 1 'if you choose newyork then show the new york flag image

storeImage.ImageUrl = "newyork.png"

storeImage.Visible = True

Case 2 'if you choose oregon then show the oregon flag image

storeImage.ImageUrl = "oregon.png"

storeImage.Visible = True

Case 3 'if you choose washington then show the washington flag image

storeImage.ImageUrl = "washington.jpg"

storeImage.Visible = True

Case 4 'if you choose lowa then show the lowa flag image

storeImage.ImageUrl = "lowa.png"

storeImage.Visible = True

End Select

End Sub

Protected Sub typeRadioButtonList2\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles typeRadioButtonList2.SelectedIndexChanged

'Image control in which the picture changes depending on the selected chair material and if material and type does not match, show the error message.

MaintainScrollPositionOnPostBack = True

Select Case typeRadioButtonList2.SelectedIndex

Case 0 'if you choose wingback then show the wingback image

typeImage.ImageUrl = "wingback.jpg"

typeImage.Visible = True

Case 1 'if you choose club then show the club image

typeImage.ImageUrl = "club.jpg"

typeImage.Visible = True

Case 2 'if you choose folding then show the folding image

typeImage.ImageUrl = "folding.jpg"

typeImage.Visible = True

Case 3 'if you choose sofa then show the sofa image

typeImage.ImageUrl = "sofa.jpg"

typeImage.Visible = True

End Select

'It shows the basic price for the of the selected chair.

TypeLa.Visible = True

TypeLa.Text = "Basic Price: " & typeRadioButtonList2.SelectedValue

End Sub

Protected Sub materialRadioButtonList3\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles materialRadioButtonList3.SelectedIndexChanged

'If the chair type and the chair material does not match, it functions to show errors through the label text.

MaintainScrollPositionOnPostBack = True

'if chair type does not selected, then show the error label and return empty material list checked

If typeRadioButtonList2.SelectedIndex < 0 Then

errorChairLa.Visible = True

materialRadioButtonList3.SelectedIndex = -1

Exit Sub

Else 'show nothing

errorChairLa.Visible = False

End If

Select Case materialRadioButtonList3.SelectedIndex

Case 0 To 2 'if you choose wingback but the material of chair is not wingback, show the error message

If typeRadioButtonList2.SelectedIndex > 0 Then

materialLabel.Visible = True

Else 'show nothing

materialLabel.Visible = False

End If

Case 3 To 5 'if you choose club but the material of chair is not club, show the error message

If typeRadioButtonList2.SelectedIndex = 0 OrElse typeRadioButtonList2.SelectedIndex = 2 OrElse typeRadioButtonList2.SelectedIndex = 3 Then

materialLabel.Visible = True

Else 'show nothing

materialLabel.Visible = False

End If

Case 6 To 8 'if you choose folding but the material of chair is not folding, show the error message

If typeRadioButtonList2.SelectedIndex = 0 OrElse typeRadioButtonList2.SelectedIndex = 1 OrElse typeRadioButtonList2.SelectedIndex = 3 Then

materialLabel.Visible = True

Else 'show nothing

materialLabel.Visible = False

End If

Case 9 To 11 'if you choose sofa but the material of chair is not sofa, show the error message

If typeRadioButtonList2.SelectedIndex = 0 OrElse typeRadioButtonList2.SelectedIndex = 1 OrElse typeRadioButtonList2.SelectedIndex = 2 Then

materialLabel.Visible = True

Else 'show nothing

materialLabel.Visible = False

End If

End Select

Select Case materialRadioButtonList3.SelectedIndex

'if you choose a fabric then show the fabric image

Case 0

materialImage.Visible = True

materialImage.ImageUrl = "fabric.jpg"

Case 3

materialImage.Visible = True

materialImage.ImageUrl = "fabric.jpg"

Case 6

materialImage.Visible = True

materialImage.ImageUrl = "fabric.jpg"

Case 9

materialImage.Visible = True

materialImage.ImageUrl = "fabric.jpg"

'if you choose a leather then show the leather image

Case 1

materialImage.Visible = True

materialImage.ImageUrl = "leather.jpg"

Case 4

materialImage.Visible = True

materialImage.ImageUrl = "leather.jpg"

Case 7

materialImage.Visible = True

materialImage.ImageUrl = "leather.jpg"

Case 10

materialImage.Visible = True

materialImage.ImageUrl = "leather.jpg"

'if you choose a mesh then show the mesh image

Case 2

materialImage.Visible = True

materialImage.ImageUrl = "mesh.jpg"

Case 5

materialImage.Visible = True

materialImage.ImageUrl = "mesh.jpg"

Case 8

materialImage.Visible = True

materialImage.ImageUrl = "mesh.jpg"

Case 11

materialImage.Visible = True

materialImage.ImageUrl = "mesh.jpg"

End Select

'show the extra charge you selected

materialLa.Visible = True

materialLa.Text = "Extra Charge: " & materialRadioButtonList3.SelectedValue

End Sub

Protected Sub CheckBoxList1\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles CheckBoxList1.SelectedIndexChanged

'if you choose the extra option then show the total amount of extra fee in extra option textbox

MaintainScrollPositionOnPostBack = True

Dim strExtra As String = Nothing 'this variable used to build the list of extra option

Dim decTotalExtra As Decimal = 0 'total extra option fee

If CheckBoxList1.SelectedIndex > -1 Then

Dim extraOpt As ListItem

For Each extraOpt In CheckBoxList1.Items 'for each item on the list of items for the checkbox. Run this loop once for each item on the list of extra option. So 5 times. Use a loop to run code 5 times. Much more on loops another day.

If extraOpt.Selected = True Then 'If extra option is selected, show the type of extra option and price of each extra option.

strExtra &= extraOpt.Text & ": " & extraOpt.Value & vbNewLine

'add local total extra option price

decTotalExtra += extraOpt.Value

End If

Next

End If

'Displays the price of each option for the option selected in the extraOutput multi-line text box.

extraOutput.Text = "[List of Extra Option]" & vbNewLine

extraOutput.Text &= strExtra & vbNewLine

extraOutput.Text &= "Total: " & decTotalExtra.ToString("C2")

End Sub

Protected Sub colorRadioButtonList\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles colorRadioButtonList.SelectedIndexChanged

'Image control in which the picture changes depending on the selected chair color

MaintainScrollPositionOnPostBack = True

Select Case colorRadioButtonList.SelectedIndex

Case 0 'if you choose the black then show the black color image

colorImage.Visible = True

colorImage.ImageUrl = "black.jpg"

Case 1 'if you choose the burgundy then show the burgundy color image

colorImage.Visible = True

colorImage.ImageUrl = "burgundy.png"

Case 2 'if you choose the white then show the white color image

colorImage.Visible = True

colorImage.ImageUrl = "white.jpg"

Case 3 'if you choose the gray then show the gray color image

colorImage.Visible = True

colorImage.ImageUrl = "gray.png"

Case 4 'if you choose the beige then show the beige color image

colorImage.Visible = True

colorImage.ImageUrl = "beige.jpg"

Case 5 'if you choose the brown then show the brown color image

colorImage.Visible = True

colorImage.ImageUrl = "brown.png"

End Select

End Sub

#End Region

#Region "Make transaction & summary data table"

Protected Sub invoiceButton\_Click(sender As Object, e As EventArgs) Handles invoiceButton.Click

'When you click this button, it has the function of storing transactions for each customer in the data table and storing each summary data table

MaintainScrollPositionOnPostBack = True

'11 local decimal variables

Dim discount, tax, priceType, quantity, priceMaterial, priceQuantotal, chairTotal, dectotal, extra, taxPrice, addExtra As Decimal

'2 local integer variables

Dim intExtraCount, intmember As Integer

If nameTextBox.Text = Nothing Then 'If nothing is write, indicate an error in the txtOutput text box and it indicates which item is incorrect

txtOutput.Text = "Please enter the Name!!!"

nameTextBox.BorderColor = Drawing.Color.Red

nameTextBox.BorderWidth = 5

Exit Sub

Else 'If correct, txtOutput does not display any errors

nameTextBox.BorderColor = Drawing.Color.White

nameTextBox.BorderWidth = 1

End If

'error check store list

If storeRadioButtonList1.SelectedIndex < 0 Then 'If nothing is selected, indicate an error in the txtOutput text box and it indicates which item is incorrect

txtOutput.Text = "Please select the store!!!"

storeRadioButtonList1.BorderColor = Drawing.Color.Red

storeRadioButtonList1.BorderWidth = 5

Exit Sub

Else 'If correct, txtOutput does not display any errors

storeRadioButtonList1.BorderColor = Drawing.Color.White

storeRadioButtonList1.BorderWidth = 1

tax = storeRadioButtonList1.SelectedValue \* 0.01

End If

'error check chair type list

If typeRadioButtonList2.SelectedIndex < 0 Then 'If nothing is selected, indicate an error in the txtOutput text box and it indicates which item is incorrect

txtOutput.Text = "Please select the chair type!!!"

typeRadioButtonList2.BorderColor = Drawing.Color.Red

typeRadioButtonList2.BorderWidth = 5

Exit Sub

Else 'If correct, txtOutput does not display any errors

typeRadioButtonList2.BorderColor = Drawing.Color.White

typeRadioButtonList2.BorderWidth = 1

priceType = typeRadioButtonList2.SelectedValue

End If

'error check chair material

If materialRadioButtonList3.SelectedIndex < 0 Then 'If nothing is selected, indicate an error in the txtOutput text box and it indicates which item is incorrect

txtOutput.Text = "Please select the chair material!!!"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'If correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

priceMaterial = materialRadioButtonList3.SelectedValue

End If

'error check chair material match the type

If typeRadioButtonList2.SelectedIndex = 0 And materialRadioButtonList3.SelectedIndex > 2 Then 'if you choose wingback but the material of chair is not wingback, show the error message in txtOuput text box

txtOutput.Text = "You select the " & materialRadioButtonList3.SelectedItem.Text & vbNewLine & vbNewLine

txtOutput.Text &= "Please match the chair type and chair material!!! (Wingback)"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

End If

'error check chair material match the type

If typeRadioButtonList2.SelectedIndex = 1 And materialRadioButtonList3.SelectedIndex < 3 Then 'if you choose club but the material of chair is not club, show the error message in txtOuput text box

txtOutput.Text = "You select the " & materialRadioButtonList3.SelectedItem.Text & vbNewLine & vbNewLine

txtOutput.Text &= "Please match the chair type and chair material!!! (Club)"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

End If

'error check chair material match the type

If typeRadioButtonList2.SelectedIndex = 1 And materialRadioButtonList3.SelectedIndex > 5 Then 'if you choose club but the material of chair is not club, show the error message in txtOuput text box

txtOutput.Text = "You select the " & materialRadioButtonList3.SelectedItem.Text & vbNewLine & vbNewLine

txtOutput.Text &= "Please match the chair type and chair material!!! (Club)"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

End If

'error check chair material match the type

If typeRadioButtonList2.SelectedIndex = 2 And materialRadioButtonList3.SelectedIndex > 8 Then 'if you choose folding but the material of chair is not folding, show the error message in txtOuput text box

txtOutput.Text = "You select the " & materialRadioButtonList3.SelectedItem.Text & vbNewLine & vbNewLine

txtOutput.Text &= "Please match the chair type and chair material!!! (Folding)"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

End If

'error check chair material match the type

If typeRadioButtonList2.SelectedIndex = 2 And materialRadioButtonList3.SelectedIndex < 6 Then 'if you choose folding but the material of chair is not folding, show the error message in txtOuput text box

txtOutput.Text = "You select the " & materialRadioButtonList3.SelectedItem.Text & vbNewLine & vbNewLine

txtOutput.Text &= "Please match the chair type and chair material!!! (Folding)"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

End If

'error check chair material match the type

If typeRadioButtonList2.SelectedIndex = 3 And materialRadioButtonList3.SelectedIndex < 9 Then 'if you choose sofa but the material of chair is not sofa, show the error message in txtOuput text box

txtOutput.Text = "You select the " & materialRadioButtonList3.SelectedItem.Text & vbNewLine & vbNewLine

txtOutput.Text &= "Please match the chair type and chair material!!! (Sofa)"

materialRadioButtonList3.BorderColor = Drawing.Color.Red

materialRadioButtonList3.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

materialRadioButtonList3.BorderWidth = 1

materialRadioButtonList3.BorderColor = Drawing.Color.White

End If

'error check color

If colorRadioButtonList.SelectedIndex < 0 Then 'If nothing is selected, indicate an error in the txtOutput text box and it indicates which itme is incorrect

txtOutput.Text = "Please select the color!!!"

colorRadioButtonList.BorderColor = Drawing.Color.Red

colorRadioButtonList.BorderWidth = 5

Exit Sub

Else 'If correct, txtOutput does not display any errors

colorRadioButtonList.BorderWidth = 1

colorRadioButtonList.BorderColor = Drawing.Color.White

End If

'error check quantity text box

If quanTextBox.Text = Nothing OrElse IsNumeric(quanTextBox.Text) = False Then 'If nothing is write and is not numeric, indicate an error in the txtOutput text box and it indicates which item is incorrect

txtOutput.Text = "Please enter the quantity of chair (Numeric)!!!"

quanTextBox.BorderColor = Drawing.Color.Red

quanTextBox.BorderWidth = 5

Exit Sub

Else 'if correct, txtOutput does not display any errors

quanTextBox.BorderWidth = 1

quanTextBox.BorderColor = Drawing.Color.White

quantity = Convert.ToDecimal(quanTextBox.Text)

End If

'chair price

chairTotal = priceMaterial + priceType

'chair price into addExtra variable

addExtra = chairTotal

If CheckBoxList1.SelectedIndex > -1 Then

Dim extraOpt As ListItem

For Each extraOpt In CheckBoxList1.Items 'for each item on the list of items for the checkbox. Run this loop once for each item on the list of extra option. So 5 times. Use a loop to run code 5 times. Much more on loops another day.

If extraOpt.Selected = True Then

extra += extraOpt.Value 'add price of each extra option selected

intExtraCount += extraOpt.Selected 'add the quantity of extra option selected

End If

Next

End If

'total extra quantity (number of additional options increases with the number of chairs (multiplied)

intExtraCount \*= quantity

'total extra option price

addExtra += extra

'total price measurement

priceQuantotal = addExtra \* quantity

'total price of extra option multiplied by quantity

addExtra = addExtra \* quantity

Dim strmember As String 'member string variable

intExtraCount = Math.Abs(intExtraCount)

If memCheckBox.Checked = True Then 'if member then gives 10 % discount and member count increase

addExtra \*= 0.9 'this gives 10 %discount

discount = priceQuantotal - priceQuantotal \* 0.9

strmember = "Yes" 'if member

'member count

intmember += 1

member += 1

Else 'if not member

strmember = "No" 'if not member

End If

'total transactions price

taxAdd += (addExtra \* tax) + addExtra

dectotal += (addExtra \* tax) + addExtra

'tax price

taxPrice = (addExtra \* tax)

'total transactions price before tax

taxPre += addExtra

'number of all items sold

soldAll += quantity

'Display total amount of chair cost(returned) in txtOutput text box.

txtOutput.Text = "The total amount " & nameTextBox.Text & " has to pay " & dectotal.ToString("C0") & " in " & storeRadioButtonList1.SelectedItem.Text & " store" & vbNewLine & vbNewLine

txtOutput.Text &= nameTextBox.Text & "'s transaction has been added to the data table!!!" & vbNewLine

Dim drT As DataRow = dtTransaction.NewRow 'new row in the transaction data table

'add rows to each column with data (transaction)

'customer name

drT.Item("CustomerName") = nameTextBox.Text

'store location

drT.Item("StoreLocation") = storeRadioButtonList1.SelectedItem.Text

'chair type & material

drT.Item("Chair") = materialRadioButtonList3.SelectedItem.Text

'number of purchases

drT.Item("Quantity") = quantity

'chair cost (only chair)

drT.Item("ChairCost") = chairTotal

'total chair cost (add quantity and extra option)

drT.Item("TotalChairCost") = priceQuantotal

'extra option price

drT.Item("ExtraCost") = extra

'color

drT.Item("Color") = colorRadioButtonList.SelectedItem.Text

'order date (current date)

drT.Item("OrderDate") = Now.ToShortDateString

'member

drT.Item("Member") = strmember

'discount price

drT.Item("DiscountPrice") = discount

'tax rate

drT.Item("%TaxRate") = storeRadioButtonList1.SelectedValue

'tax price

drT.Item("TaxPrice") = taxPrice

'total cost before tax added

drT.Item("TotalCostPreTax") = addExtra

'total cost after tax added

drT.Item("TotalCost") = dectotal

'row to store data

dtTransaction.Rows.Add(drT)

'Increase the total number of reservations when the button is clicked and performed

transactionLabel.Text = "Total of transactions: " & dtTransaction.Rows.Count

transactionLabel.Visible = True

tranGridView.DataSource = dtTransaction

tranGridView.DataBind() 'used to see if the array is created and available

'add rows to each column with chair type summary data table

With dtChairType.Rows(typeRadioButtonList2.SelectedIndex)

.Item("Quantity") += quantity

.Item("ExtraQuantity") += intExtraCount

.Item("TotalCost") += dectotal

End With

'add rows to each column with chair material summary data table

With dtChairInd.Rows(materialRadioButtonList3.SelectedIndex)

.Item("Quantity") += quantity

.Item("ExtraQuantity") += intExtraCount

.Item("TotalCost") += dectotal

End With

'add rows to each column with grand total summary data table

With dtGrandTotal.Rows(0)

'total customer (count)

.Item("TotalCustomers") = dtTransaction.Rows.Count

'total sales price

.Item("TotalCost") = taxAdd

'total chair sold

.Item("TotalQuantity") = soldAll

'total member

.Item("MemberCount") = member

End With

'add rows to each column with chair color summary data table

With dtColor.Rows(colorRadioButtonList.SelectedIndex)

.Item("Quantity") += quantity

End With

'add rows to each column with store summary data table

With dtStore.Rows(storeRadioButtonList1.SelectedIndex)

'each store tax rate

.Item("%TaxRate") = storeRadioButtonList1.SelectedValue

.Item("TotalCostPreTax") += addExtra

.Item("TotalCost") += dectotal

.Item("TotalQuantity") += quantity

'each store member

.Item("MemberCount") += intmember

'set amount of goal

Dim goal As Decimal = 50000

'status of goal string variable

Dim strStatus As String

.Item("Goal") = goal

.Item("RemainGoal") = (.Item("Goal") - .Item("TotalCost")) 'compute remaining price

Dim percent As Decimal 'show the percent of goal status

percent = (.Item("TotalCost") / .Item("Goal"))

.Item("%Goal") = percent.ToString("P2")

Select Case percent

Case Is <= 0.3 'if 30 % made it

strStatus = "Plan to extend efforts"

Case 0.301 To 0.6 'if 30.1 % to 60 % made it

strStatus = "Making Progress"

Case 0.601 To 0.999 'if 60.1 % to 99.9% made it

strStatus = "Almost there"

Case 1 To 1.999 'made it

strStatus = "Made it!!"

Case >= 2 'if exeed 200% made it

strStatus = "Crushed it!!"

End Select

.Item("StatusOfGoal") = strStatus 'show status of goal

End With

'used to see if the arrays are created and available (5 summary data tables)

chairTypeGridView.DataSource = dtChairType

chairTypeGridView.DataBind()

chairGridView.DataSource = dtChairInd

chairGridView.DataBind()

totalGridView.DataSource = dtGrandTotal

totalGridView.DataBind()

colorGridView.DataSource = dtColor

colorGridView.DataBind()

storeGridView.DataSource = dtStore

storeGridView.DataBind()

End Sub

#End Region

#Region "Clear"

Protected Sub clearButton\_Click(sender As Object, e As EventArgs) Handles clearButton.Click

'clear the all output text boxes, list controls, images, text boxes (only enter the transactions setting), and labels

MaintainScrollPositionOnPostBack = True

'clear customer name text box

nameTextBox.Text = Nothing

'clear store selected

storeRadioButtonList1.SelectedIndex = -1

storeImage.Visible = False

'clear chair type selected

typeRadioButtonList2.SelectedIndex = -1

typeImage.Visible = False

TypeLa.Visible = False

'clear chair material selected

materialRadioButtonList3.SelectedIndex = -1

materialLabel.Visible = False

materialLa.Visible = False

errorChairLa.Visible = False

materialImage.Visible = False

'clear color selected

colorRadioButtonList.SelectedIndex = -1

colorImage.Visible = False

'clear extra option selected

Dim ext As ListItem

For Each ext In CheckBoxList1.Items

ext.Selected = False

Next

extraOutput.Text = Nothing 'clear extra option output text box

'clear quantity text box

quanTextBox.Text = Nothing

'clear member checked

memCheckBox.Checked = False

'clear result output

txtOutput.Text = Nothing

End Sub

#End Region

#Region "Visualization"

Protected Sub DrawCharts()

'create a total of each products sold. these are properties of the overall chart control

With chairChart

.DataSource = dtChairType

.DataBind()

.Titles.Add("Total # Products Sold")

End With

'set the x and y axis labels

With chairChart.ChartAreas(0)

.AxisX.Title = "Chair Type"

.AxisY.Title = "# Sold Quantity"

.Area3DStyle.Enable3D = True 'show 3D chart

End With

'withing the chart have series of data.

With chairChart.Series("Series1")

.ChartType = DataVisualization.Charting.SeriesChartType.Column

.XValueMember = "ChairType" 'these are columns in array - a dimension

.YValueMembers = "Quantity" 'this is the data to chart

.IsValueShownAsLabel = True

End With

'create a total sales of each store. these are properties of the overall chart control

With storeChart

.DataSource = dtStore

.DataBind()

.Titles.Add("Total # Sales for Store")

End With

'set the x and y axis labels

With storeChart.ChartAreas(0)

.AxisX.Title = "Store Type"

.AxisY.Title = "# Sales Price"

.Area3DStyle.Enable3D = True 'show 3D chart

End With

'withing the chart have series of data.

With storeChart.Series("Series1")

.ChartType = DataVisualization.Charting.SeriesChartType.Bar

.XValueMember = "Store" 'these are columns in array - a dimension

.YValueMembers = "TotalCost" 'this is the data to chart

.IsValueShownAsLabel = True

End With

'create a total sold of each color (Quantity). these are properties of the overall chart control

With colorChart

.DataSource = dtColor

.DataBind()

.Titles.Add("Each Color Quantity")

.Legends.Add("Legend1") ' the pie chart looks better with a legend

.Legends("Legend1").Enabled = True 'the legend is pulled from the X - dimension field specified

.ChartAreas(0).Area3DStyle.Enable3D = True

End With

'withing the chart have series of data.

With colorChart.Series("Series1")

.ChartType = DataVisualization.Charting.SeriesChartType.Pie

.XValueMember = "ColorType" 'the dimension

.YValueMembers = "Quantity" ' the measure

.IsValueShownAsLabel = True

.IsVisibleInLegend = True

End With

End Sub

#End Region

End Class

**Website URL:**

<https://cb-ot-devst06.ad.wsu.edu/MF13namjun.lee/FinalProject/FinalProject.aspx>