Lacey’s Salon Services

Date: December 12th, 2022

By: Lacey Shivers

The Problem:

This project was created to improve the problem of recording monthly revenue goals for salons and adding the functionality of being able to update each salon when it generates revenue throughout the month. This will allow seeing which salons are meeting their revenue goals, with the addition of identifying what the type of salon is and where it is located. Using this system, one will be able to see which types of salons are doing well in which types of areas and use this information to expand their services. The problem consists of it being hard to manage all the data if a company or someone owns multiple salons, and this would be better than using Excel because you are able to update it throughout the month in a way that gives instant feedback and isn't cluttered. This solution allows for this database to track the progress of revenue goals and see the revenues generated by all the salons they own. Using this project, an owner can solve this issue of business analytics by allowing for an easy-to-use system that updates revenue for their salons, records a running total of revenue, records the total transactions (days open) for a specific salon, and compares each salon to the revenue goals that are set at the beginning of the month.

The Customer:

This targeted customer would be a company or someone who owns multiple salons in different locations and are wanting to create a system that can easily monitor their businesses. This project would provide a stored database that keeps track of each salon using a unique ID number while showing the important details of each salon such as revenue. This would prove to be beneficial for the customer as it allows them to monitor the types of salons they may own and seeing what locations have been successful if they want to expand their business. This system would allow for the owner to spend more time focusing on expanding/improving their business

by having a system that keeps records of the revenue for each salon they own. This includes each of the salon's location, names, transactions with the dates they are recorded, and a unique ID number that is used when updating the revenue whether that is daily, weekly, or monthly.

Website Solution:

This website helps provide analytics to solve the problem of a cluttered and difficult to update Excel sheet. This is done by allowing each salon to be stored in a data table that is easy to read and can be updated daily, weekly, or even monthly. Using this website allows for setting revenue goals each month, updating the amount of revenue as it is generated, and comparing the total revenue to the goal to see what percentage of the goal has been obtained. This system also provides opportunities for other analytics to be constructed on each salons information, including projected monthly revenue and yearly revenue as well. This can be used to determine if more advertising needs to be done for a specific salon and compares each type of salon with the location, they are in to see how well it is doing. This revenue percentage shows how close the salon is to their monthly goal and can be updated daily or weekly to understand how much they need to generate to reach their goal.

Benefits:

The benefits from using this system are having a simplified view of revenue streams, data tables that are updated with each transaction and easy to understand, a chart that compares each type of salon to revenue generated, and charts that compare each specific salon ID to the revenue they are generating. This data is beneficial for new business ideas, possible expansions of business, and seeing which types of salons have excelled in their business. This data creates a

bigger picture of the market and allows for each salon to determine if they are operating efficiently to reach their revenue goal. This system is designed to allow new entries if they open a new salon, updating the revenue stream as often as they would like, and have quick access to how close each salon is to their revenue goals. There are many benefits to this system including being able to update each salon daily, weekly, or monthly while being able to see their total revenue and how close they are to their goals in each transaction.

Functionality:

This system uses a website to incorporate all the analytical elements about each salon and divides this into multiple tabs to create a webpage that is not crowded and easy-to-use. These tabs each serve a different purpose and they are listed as Salon Set-Up, Insert Salon Data, and Management Reporting. First, the Salon Set-Up tab allows the user to enter a unique salon ID into a text box and then they insert their revenue goals into the next text box. This tab also includes a location for the salon so this must be included when they are inserting the data. This system uses error-checking techniques that only allows numbers to be inserted into the unique salon ID and revenue goals to ensure the salon is recorded correctly. There is also an error-checking method used to ensure the location of the salon is recorded. After this data is recorded, it is translated into a data table that includes the unique ID, revenue goals, and the location of the salon. The second tab, Insert Salon Data, is used to update the revenue goals for each salon and assign a name to the salon based on the unique salon ID. This includes inserting the salon name into a text box, recording the revenue generated in a text box, selecting the unique ID number of the salon you are recording for, selecting the type of salon, and recording the date the data is being inserted using a calendar text box. There is error-checking in place to ensure all the data is

correctly inserted and all the textboxes/radio button lists have a selection. This includes insuring only letters are typed into the salon name, only numbers are typed into the revenue total, a date has been selected, and both radio button lists have a selection. This tab includes a text output box that allows the user to see the transaction for what unique salon ID was recorded. Then, this data is transmitted into a data table that updates the total revenue, records the salon name, records the salon type, records the unique salon ID that the data was inserted for, and records the date this transaction was made. Finally, the third tab includes the running total databases and the three charts that were created to compare the revenue with different factors of each salon. After following the steps listed above, the database will output the correct tables and charts that create comparisons in the data and provided business intelligence that refers to each salon. This system has much room for added functionality once it is constructed into a company, and there are many benefits to implementing this type of system.

Final Project Working:

localhost:49512/Homework/Fina

localhost:49512/Homework/Final.aspx

Lacey's Salon Services

Set-Up Salon Insert Salon Data Management Reporting

Set-Up Salon:

Salon Number:

Revenue Goals:

Salon Location: ○ Pullman, WA ○ Tacoma, WA ○ Olympia, WA

Save Data

Clear

SalonID	RevenueGoal	Location
1	5200	Olympia, WA
3	4800	Olympia, WA
7	3900	Tacoma, WA
8	4600	Tacoma, WA

Set-Up Salon Insert Salon Data Management Reporting

Insert Salon Data:

Salon Name:

Revenue Total:

Salon Number: ○ 1 ○ 3 ○ 7 ○ 8 ○ 13 ○ 16

Salon Type: ○ Hair ○ Nail ○ Massage ○ Tanning

Please Select Date: mm/dd/yyyy

Insert Data

Clear

```
Sale Recorded for Salon #: 16
```

TransactionID	SalonID	SalonName	SalonType	Revenue	Date
1	1	Splinters	Nail	5800	12/06/22
2	3	Muscle	Massage	4200	12/06/22
3	7	Perms	Hair	2000	12/06/22
4	8	Sunnys	Tanning	3600	12/06/22
5	13	Finnigans	Nail	4600	12/06/22
6	16	Curlys	Hair	8000	12/06/22

Lacey's Salon Services

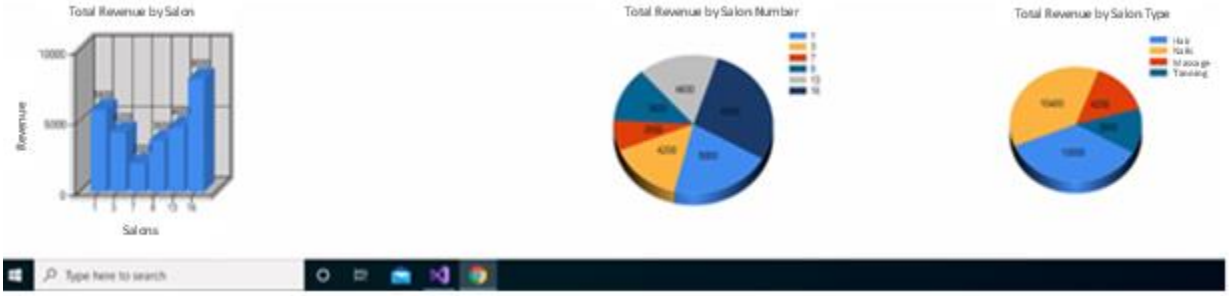
Set-Up Salon Insert Salon Data Management Reporting

Managment Reporting:

SalonID	Location	NumberTA	Revenue	RevenueGoal	PercentofRevenue	Status
1	Olympia, WA	1	5800	5200	112.00%	Awesome! You're Killing It
3	Olympia, WA	1	4200	4800	88.00%	Almost There, Keep Pushing
7	Tacoma, WA	1	2000	3900	51.00%	Efforts Still Need to Increase
8	Tacoma, WA	1	3600	4600	78.00%	Almost There, Keep Pushing
13	Pullman, WA	1	4600	4300	107.00%	Awesome! You're Killing It
16	Pullman, WA	1	8000	3800	211.00%	Amazing! Profits are atleast 2X Expected

SalonType	Revenue	SalonTypeTA
Hair	10000	2
Nail	10400	2
Massage	4200	1
Tanning	3600	1

Final Project Working - Charts:



Final Project Code:

```
1   Imports System.Data
2
    2 references
3   Partial Class Homework_Final
4       Inherits System.Web.UI.Page
5
6       'Defines the datatables that are used
7       Public Shared gdtSalons, gdtSalonDetails, gdtSalonTotals, gdtTypeTotal As New DataTable
8
9       'Creates a Region
10  #Region "Save a Row of Data"
11
12      'Creates Protected Sub procedure to be called later
        1 reference
13      Protected Sub SaveDataRow()
14
15          'Defines Variable
16          Dim decSalonID As Decimal
17
18          'Error Checking for inputs
19          If txtRevenue.Text = Nothing OrElse txtSalonName.Text = Nothing OrElse txtDate.Text = Nothing OrElse IsNumeric
              (txtRevenue.Text) = False Then
20              txtOutput.Text = "Please Check Entry "
21              Exit Sub
22          End If
23
24          'Error Checking for inputs
25          If rblSalonNumber.SelectedIndex = -1 OrElse rblSalonType.SelectedIndex = -1 Then
26              txtOutput.Text &= "Please Select a Salon # and Type"
27              Exit Sub
28          End If
29
30          'Error Checking for inputs
31          If IsNumeric(txtSalonName) = True Then
32              txtOutput.Text &= "Please Ensure the Salon Name is not a Value"
33              Exit Sub
34          End If
35
36          'Used to Create a New Row
37          Dim dr3 As DataRow = gdtSalonDetails.NewRow
38
39          'Defines what variable equals
40          decSalonID = rblSalonNumber.SelectedValue
41
42          'Allows for Columns to be given a value
43          dr3("SalonID") = decSalonID
44          dr3("SalonName") = txtSalonName.Text
45          dr3("Revenue") = Convert.ToDecimal(txtRevenue.Text)
46
47          'Allows for Columns to be given a value
48          dr3("Date") = DateTime.Parse(txtDate.Text).ToString("MM/dd/yy")
49          dr3("SalonType") = rblSalonType.SelectedItem.Text
50
51          'Adds the rows define and binds it to gridview
52          gdtSalonDetails.Rows.Add(dr3)
53          GridView2.DataSource = gdtSalonDetails
54          GridView2.DataBind()
55
56          'Outputs text for recording Salon Number
57          txtOutput.Text = "Sale Recorded for Salon #: " & rblSalonNumber.SelectedItem.Text
58      End Sub
```

```
#End Region

    'Creates a Region
#Region "Update Totals and Record Data"
    0 references
    Protected Sub btnInsert_Click(sender As Object, e As EventArgs) Handles btnInsert.Click

        'Calls Protected Sub that was created
        Call SaveDataRow()

        'Defines Variables
        Dim decPercentGoal As Decimal
        Dim strStatus As String

        'Updates Running Totals
        With gdtSalonTotals.Rows(rblSalonNumber.SelectedIndex)
            .Item("NumberTA") += 1
            .Item("Revenue") += Convert.ToDecimal(txtRevenue.Text)

            'Defines Value for the Variable
            decPercentGoal = FormatNumber(.Item("Revenue") / .Item("RevenueGoal"), 2)

            'Used to Define what should be outputted at each percentage level
            Select Case decPercentGoal

                'Messages to Incourage
                Case Is < 0.3
                    strStatus = "Plan to Promote More"
                Case 0.301 To 0.667
                    strStatus = "Efforts Still Need to Increase"
                Case 0.668 To 0.999
                    strStatus = "Almost There, Keep Pushing"

                    'Messages to Congratulate
                Case 1.0 To 1.999
                    strStatus = "Awesome! You're Killing It"
                Case > 2
                    strStatus = "Amazing! Profits are atleast 2X Expected"
            End Select

            'Updates the Running Total and adds status
            .Item("PercentofRevenue") = decPercentGoal.ToString("P2")
            .Item("Status") = strStatus

            'Binds data to the gridview
            GridView3.DataSource = gdtSalonTotals
            GridView3.DataBind()
        End With

        'Updates Running Totals
        With gdtTypeTotal(rblSalonType.SelectedIndex)
            .Item("SalonTypeTA") += 1
            .Item("Revenue") += Convert.ToDecimal(txtRevenue.Text)

            'Binds data to the gridview
            GridView4.DataSource = gdtTypeTotal
            GridView4.DataBind()
        End With

        'Clears out the rbl lists after button is clicked to store
        rblSalonType.SelectedIndex = -1
        rblSalonNumber.SelectedIndex = -1

        'Clears out the txt boxes after button is clicked to store
        txtRevenue.Text = Nothing
        txtSalonName.Text = Nothing
        txtDate.Text = Nothing
    End Sub
#End Region
```

```
128         'Creates a Region
129     #Region "Set-Up Salon"
            0 references
130         Protected Sub btnSetUp_Click(sender As Object, e As EventArgs) Handles btnSetUp.Click
131
132             'Defines a Variable
133             Dim decSalonNumber, decRevGoals As Decimal
134
135             'Used to Create a New Row
136             Dim dr1 As DataRow = gdtSalons.NewRow
137
138             'Error Checking for Columns
139             If rblSalonNumber.Items.Count = 0 Then
140                 gdtSalonDetails.Rows.Clear()
141                 gdtSalonTotals.Rows.Clear()
142             End If
143
144             'Error Checking for Data Entries
145             If rblLocation.SelectedIndex = -1 OrElse txtSalonNumber.Text = Nothing OrElse IsNumeric(txtSalonNumber.Text) =
                  False OrElse txtRevenueGoal.Text = Nothing OrElse IsNumeric(txtRevenueGoal.Text) = False Then
146                 Response.Write("Please Check Data Entry")
147                 Exit Sub
148             End If
149
150             'Allows for the Salon Number txt box to add an item into the rbl list
151             With rblSalonNumber
152                 .BorderColor = Drawing.Color.DodgerBlue
153                 .Items.Add(txtSalonNumber.Text)
154             End With
155
156             'Defines what table to use
157             With gdtSalons
158
159                 'Defines a Value to a Variable
160                 decRevGoals = Convert.ToDecimal(txtRevenueGoal.Text)
161                 decSalonNumber = Convert.ToDecimal(txtSalonNumber.Text)
162
163                 'Defines values for the new rows added
164                 dr1("SalonID") = decSalonNumber
165                 dr1("RevenueGoal") = decRevGoals
166                 dr1("Location") = rblLocation.SelectedItem.Text
167             End With
168
169             'Binds the new data with the gridview
170             gdtSalons.Rows.Add(dr1)
171             GridView1.DataSource = gdtSalons
172             GridView1.DataBind()
173
174             'Used to Create a New Row
175             Dim dr2 As DataRow = gdtSalonTotals.NewRow
176
177             'Defines values for the new rows added
178             dr2("SalonID") = txtSalonNumber.Text
179             dr2("RevenueGoal") += Convert.ToDecimal(txtRevenueGoal.Text)
180             dr2("Location") = rblLocation.SelectedItem.Text
181
182             'Binds the new data with the gridview
183             gdtSalonTotals.Rows.Add(dr2)
184             GridView3.DataSource = gdtSalonTotals
185             GridView3.DataBind()
186
187             'Clears out rbl List and txt Boxes after button is clicked to store
188             rblLocation.SelectedIndex = -1
189             txtSalonNumber.Text = Nothing
190             txtRevenueGoal.Text = Nothing
191         End Sub
192     #End Region
```

```
193
194         'Creates a region
195     #Region "Load Datatable"
196
                0 references
197         Private Sub Homework_FINAL_Init(sender As Object, e As EventArgs) Handles Me.Init
198
199             'Sets view to -1 so the information is not displayed until linkbtn is clicked
200             MultiView1.ActiveViewIndex = -1
201             Image1.ImageUrl = "~\Images\Salon.jpg"
202
203             'Error checking for total Columns of datatables
204             If gdtSalons.Columns.Count > 0 OrElse gdtSalonDetails.Columns.Count > 0 OrElse gdtSalonTotals.Columns.Count > 0
                  Then Exit Sub
205
206             'Used to create the datatable
207             With gdtSalons
208
209                 'Adds Columns to the datatable
210                 .Columns.Add("SalonID", GetType(Decimal))
211                 .Columns.Add("RevenueGoal", GetType(Decimal))
212                 .Columns.Add("Location", GetType(String))
213             End With
214
215             'Used to create the datatable
216             With gdtSalonDetails
217
218                 'Adds Columns to the datatable
219                 .Columns.Add("TransactionID", GetType(Integer))
220                 .Columns.Add("SalonID", GetType(Decimal))
221                 .Columns.Add("SalonName", GetType(String))
222
223                 'Adds Columns to the datatable
224                 .Columns.Add("SalonType", GetType(String))
225                 .Columns.Add("Revenue", GetType(Decimal))
226                 .Columns.Add("Date", GetType(String))
227
228                 'Zeros out the values in the DataTable
229                 .Columns("SalonName").DefaultValue = Nothing
230                 .Columns("Date").DefaultValue = Nothing
231
232                 'Zeros out the values in the DataTable
233                 .Columns("Revenue").DefaultValue = 0
234                 .Columns("SalonType").DefaultValue = 0
235                 .Columns("SalonID").DefaultValue = 0
236             End With
237
238             'Creates AutoIncrement for the SalonID to be counted as a different transcation
239             With gdtSalonDetails.Columns("TransactionID")
240                 .AutoIncrement = True
241                 .AutoIncrementSeed = 1
242                 .AutoIncrementStep = 1
243             End With
244
245             'Used to create the datatable
246             With gdtSalonTotals
247
248                 'Adds Columns to the datatable
249                 .Columns.Add("SalonID", GetType(Decimal))
250                 .Columns.Add("Location", GetType(String))
251                 .Columns.Add("NumberTA", GetType(Integer))
252
253                 'Adds Columns to the datatable
254                 .Columns.Add("Revenue", GetType(Decimal))
255                 .Columns.Add("RevenueGoal", GetType(Decimal))
256                 .Columns.Add("PercentofRevenue", GetType(String))
257                 .Columns.Add("Status", GetType(String))
258
259                 'Zeros out the values in the DataTable
260                 .Columns("SalonID").DefaultValue = 0
261                 .Columns("NumberTA").DefaultValue = 0
262                 .Columns("Location").DefaultValue = 0
263                 .Columns("Revenue").DefaultValue = 0
264
265                 'Zeros out the values in the DataTable
266                 .Columns("RevenueGoal").DefaultValue = 0
267                 .Columns("PercentofRevenue").DefaultValue = Nothing
268                 .Columns("Status").DefaultValue = Nothing
269             End With
270
```

```
271             'Used to create the datatable
272             With gdtTypeTotal
273
274                 'Adds Columns to the datatable
275                 .Columns.Add("SalonType", GetType(String))
276                 .Columns.Add("Revenue", GetType(Decimal))
277                 .Columns.Add("SalonTypeTA", GetType(Integer))
278
279                 'Zeros out the values in the DataTable
280                 .Columns("SalonType").DefaultValue = Nothing
281                 .Columns("Revenue").DefaultValue = 0
282                 .Columns("SalonTypeTA").DefaultValue = 0
283             End With
284
285             'Deinfes the Clumn SalonType to include EACH option from the rbl list for running totals
286             For Each li As ListItem In rblSalonType.Items
287
288                 'Creates a New Row and Adds it
289                 Dim dr4 As DataRow = gdtTypeTotal.NewRow
290                 dr4.Item("SalonType") = li.Text
291                 gdtTypeTotal.Rows.Add(dr4)
292             Next
293
294             'Binds the DataTable to the gridview selected
295             GridView1.DataSource = gdtSalons
296             GridView1.DataBind()
297
298             'Binds the DataTable to the gridview selected
299             GridView2.DataSource = gdtSalonDetails
300             GridView2.DataBind()
301
302             'Binds the DataTable to the gridview selected
303             GridView3.DataSource = gdtSalonTotals
304             GridView3.DataBind()
305
306             'Binds the DataTable to the gridview selected
307             GridView4.DataSource = gdtTypeTotal
308             GridView4.DataBind()
309         End Sub
310     #End Region
311
312     'Creates a Region
313     #Region "Utilitites"
314
315     'Used to Navigate to a view from the link button
        0 references
316     Protected Sub LinkButton1_Click(sender As Object, e As EventArgs) Handles LinkButton1.Click
317         MultiView1.ActiveViewIndex = 0
318     End Sub
319
320     'Used to Navigate to a view from the link button
        0 references
321     Protected Sub LinkButton2_Click(sender As Object, e As EventArgs) Handles LinkButton2.Click
322         MultiView1.ActiveViewIndex = 1
323     End Sub
324
325     'Used to Navigate to a view from the link button, and draw the charts when clicked
        0 references
326     Protected Sub LinkButton3_Click(sender As Object, e As EventArgs) Handles LinkButton3.Click
327         MultiView1.ActiveViewIndex = 2
328         'DrawChart()
329     End Sub
330
331     'Clears the Information on the first view
        0 references
332     Protected Sub btnClear1_Click(sender As Object, e As EventArgs) Handles btnClear1.Click
333         txtRevenueGoal.Text = Nothing
334         txtSalonNumber.Text = Nothing
335
336         'Clears rbl List and hides the View Index
337         rblLocation.SelectedIndex = -1
338         MultiView1.ActiveViewIndex = -1
339     End Sub
340
```

```
341         'Clears the Information on the second view
            0 references
342         Protected Sub btnClear2_Click(sender As Object, e As EventArgs) Handles btnClear2.Click
343             txtRevenue.Text = Nothing
344             txtSalonName = Nothing
345             txtDate.Text = Nothing
346
347             'Clears rbl Lists and hides the View Index
348             rblSalonType.SelectedIndex = -1
349             rblSalonNumber.SelectedIndex = -1
350             MultiView1.ActiveViewIndex = -1
351         End Sub
352
353     #End Region
354
355         '    'Creates a Region
356         '#Region "Charts"
357
358         '    'Creates Protected sub for chart creation procedure
359         '    Protected Sub DrawChart()
360
361         '        'Defines Chart 1 datasource and title
362         '        With Chart1
363         '            .DataSource = gdtSalonTotals
364         '            .DataBind()
365         '            .Titles.Add("Total Revenue by Salon Salon")
366         '        End With
367
368         '        'Defines X and Y axis for Chart1 and enables 3D view
369         '        With Chart1.ChartAreas(0)
370         '            .AxisX.Title = "Salons"
371         '            .AxisY.Title = "Revenue"
372         '            .Area3DStyle.Enable3D = True
373         '        End With
374
375         '        'Defines Type of Chart to use
376         '        With Chart1.Series("Series1")
377         '            .ChartType = DataVisualization.Charting.SeriesChartType.Column
378
379         '            'Defines Data used in X and Y axis and shows Labels
380         '            .XValueMember = "SalonID"
381         '            .YValueMembers = "Revenue"
382         '            .IsValueShownAsLabel = True
383         '            .IsXValueIndexed = True
384         '        End With
385
386         '        'Defines Chart 2 datasource and title
387         '        With Chart2
388         '            .DataSource = gdtSalonTotals
389         '            .DataBind()
390         '            .Titles.Add("Total Revenue by Salon Type")
391
392         '            'Adds Legend and Enables 3D view
393         '            .Legends.Add("Legend1")
394         '            .Legends("Legend1").Enabled = True
395         '            .ChartAreas(0).Area3DStyle.Enable3D = True
396         '        End With
397
398         '        'Defines Type of Chart to use
399         '        With Chart2.Series("Series1")
400         '            .ChartType = DataVisualization.Charting.SeriesChartType.Pie
401
402         '            'Defines Data used in X and Y axis and shows Labels
403         '            .XValueMember = "SalonID"
404         '            .YValueMembers = "Revenue"
405         '            .IsValueShownAsLabel = True
406         '            .IsVisibleInLegend = True
407         '        End With
```

```
'        'Defines Chart 2 datasource and title
'        With Chart3
'            .DataSource = gdtTypeTotal
'            .DataBind()
'            .Titles.Add("Total Revenue by Salon Type")

'            'Adds Legend and Enables 3D view
'            .Legends.Add("Legend1")
'            .Legends("Legend1").Enabled = True
'            .ChartAreas(0).Area3DStyle.Enable3D = True
'        End With

'        'Defines Type of Chart to use
'        With Chart3.Series("Series1")
'            .ChartType = DataVisualization.Charting.SeriesChartType.Pie

'            'Defines Data used in X and Y axis and shows Labels
'            .XValueMember = "SalonType"
'            .YValueMembers = "Revenue"
'            .IsValueShownAsLabel = True
'            .IsVisibleInLegend = True
'        End With
'    End Sub
'#End Region
End Class
```

**Charts are commented out for URL Purposes since graphs will only work on the Local Host.

URL Link: http://cb-ot-devst06.ad.wsu.edu/MF23lacey.shivers/Homework/Final.aspx