**CLASS: OM 424-01-22413 ASSIGNMENT NO. 2**

Riyasha Baidya

**Overview of the assignment:**

This assignment aims to expand the business analytics system developed in Assignment 1. The main objective of assignment 2 is to generate a Metric Measurement Report in PDF format. The report will include details such as total cost, production cost, inventory cost, and production schedule. Additionally, generating line charts that show the historical and the forecasted values for production unit/demand and Inventory levels. There will also be a “generate report” button created along with a message box that will display the saved PDF file’s name and directory.

**How the assignment was implemented:**

1. **Understanding and preparation:**

First, I reviewed the assignment that was provided and identified the steps I needed to take to start this assignment. I reviewed assignment 1 feedback changed the date format to month-year and changed the heading title for all the columns. Then I went through the assignment description and took notes on the steps I would need to follow. Then, I had to create 2 new worksheets so that the new data could go there, I named the worksheets AUX data and AUX report.

1. **First step:**

After creating formatting and creating a new worksheet, the first step was to calculate the inventory which we had done last assignment. I had to copy the calculate subroutine from the previous module and paste it into the new module, so the inventory column is calculated. For assignment 2, there were 6 more months added to the datasheet.

1. Second step: The second step was to put all the data needed into the AUX data, which is the first new subroutine that I created, this was recording the macro for setting up AUX data. I started by copy-pasting the headers for the column from the data sheet along with the months. Since the PDF file just asked for the past month starting from June 2023, I copy-paste the values starting from row 43 onwards. Then, I created a new column that was called Demand (Forecast) copied the last 4 months from the Demand column, and pasted it there, deleting the last 3 months from the Demand column. I did the same by creating a new column called production unit (planned) and inventory (Projected) I copy pasted the corresponding to columns for the 4 months and deleted the last 3 months. This was the first sub routine that was created.
2. **Sub** SetUpAUXDataMacro()
3. '
4. 'SetUpAUXData Macro
5. After setting up the AUX data, the next part was to create the line chart for production units and the demand for the past year along with the future horizon. This was my second subroutine which was recording a macro for these columns and making a line chart.
6. **Sub** Createproductiondemandlinechart()
7. '
8. ' Createproductiondemandlinechart Macro
9. '
11. '
12. Range("A1:C16,G1:H16").**Select**
13. Range("G1").Activate
14. ActiveSheet.Shapes.AddChart2(332, xlLineMarkers).**Select**

I had to select the columns for Data, Demand, Demand (forecast), and production unit/production unit planned. Then I generated the line chart with markers along with cutting and pasting it to the AUX report sheet. After, I had to edit the line chart formatting to distinguish between the historical and forecast values.

1. Next, the subroutine was for the other line chart which was for the inventory levels for the past year along with the future planning horizon. To do this, I did the same as the subroutine before recording macro by choosing the columns of dates, inventory, and inventory (projected). Then I generated the line chart with markers, cutting and pasting it to the AUX report. Then, formatting the graph to distinguish between the history and forecast values.
2. **Sub** createinvetorylevelslinechart()
3. '
4. ' createinvetorylevelslinechart Macro
5. '
7. '
8. Range("A1:A16,I1:J16").**Select**
9. Range("I1").Activate
10. ActiveSheet.Shapes.AddChart2(332, xlLineMarkers).**Select**
11. After creating the graph, I created the button “generate report” by going to the develop tab and assigning a subroutine to it so it activates the AUX report sheet.
12. **Sub** OpenAUXReportSheet()
13. ' Go to Aux report
14. Sheets("AUX Report").Activate
15. **End** **Sub**

***g.*** Next, was to generate the AUX report sheet to pdf. I created another subroutine called saving as pdf to generate the file name with the current date and time.

1. **Sub** savingaspdf()
2. ' Generate the file name with date and time
3. **Dim** fileName **As** **String**
4. fileName = "PPP\_Report\_" & Format(Now, "mmddyyyy\_hhmm") & ".pdf"
6. ' Export the sheet as PDF
7. Sheets("AUX Report").ExportAsFixedFormat Type:=xlTypePDF, \_
8. fileName:=fileName, Quality:=xlQualityMinimum, \_
9. IncludeDocProperties:=**True**, IgnorePrintAreas:=**False**, OpenAfterPublish:=**False**
11. ' Display message box with file information
12. MsgBox "This report has been saved to the following directory:" & vbCrLf & Application.ThisWorkbook.Path & "\" & fileName, vbInformation, "Report Name and Directory"
13. **End** **Sub**

This helped the “generate report” button to save as a pdf in the computer with the time and date you saved along with the name PPP\_Report\_

***h.*** Lastly, I needed to hide the AUX data sheet in Excel. The following code hid the AUX data sheet so that the user does not mess with the data.

1. **Sub** HideSheet()
2. ' Hiding Aux data sheet
3. ThisWorkbook.Sheets("AUX Data").Visible = xlSheetHidden
4. **End** **Sub**

**A short tutorial on how to use the .xlsm file.**

This spreadsheet allows the managers in the production planning to see the following visualization of the raw data as follows:

o *A line chart of the historical inventory levels*o *A line chart of the historical production cost along with the holding cost*

o *A line chart of the historical production units along with the demand*

Please click the buttons below to see the charts corresponding charts:

A blue rectangle with black text

Description automatically generated

Click “Generate Report” to save a pdf file of the production planning report.

A blue rectangle with black text

Description automatically generated

**VBA: Appendix**

1. **Sub** SetUpAUXDataMacro()
2. '
3. 'SetUpAUXData Macro
4. '
6. '
7. Sheets("Data").**Select**
8. Range("A1:G1").**Select**
9. Selection.Copy
10. Sheets("AUX Data").**Select**
11. Range("A1").**Select**
12. ActiveSheet.Paste
13. Sheets("Data").**Select**
14. Range("A43:A57").**Select**
15. Application.CutCopyMode = **False**
16. Selection.Copy
17. Sheets("AUX Data").**Select**
18. Range("A2").**Select**
19. ActiveSheet.Paste
20. Sheets("Data").**Select**
21. Range("B43:G57").**Select**
22. Application.CutCopyMode = **False**
23. Selection.Copy
24. Sheets("AUX Data").**Select**
25. Range("B2").**Select**
26. Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks \_
27. :=**False**, Transpose:=**False**
28. Range("C1").**Select**
29. Application.CutCopyMode = **False**
30. Selection.EntireColumn.Insert , CopyOrigin:=xlFormatFromLeftOrAbove
31. Range("C1").**Select**
32. ActiveCell.FormulaR1C1 = "Demand (Forecast)"
33. Range("B13:B16").**Select**
34. Selection.Copy
35. Range("C13").**Select**
36. ActiveSheet.Paste
37. Range("B14:B16").**Select**
38. Application.CutCopyMode = **False**
39. Selection.ClearContents
40. Range("H1").**Select**
41. Selection.EntireColumn.Insert , CopyOrigin:=xlFormatFromLeftOrAbove
42. Range("H1").**Select**
43. ActiveCell.FormulaR1C1 = "Production Units (Planned)"
44. Range("G13:G16").**Select**
45. Selection.Copy
46. Range("H13").**Select**
47. ActiveSheet.Paste
48. Range("G14:G16").**Select**
49. Application.CutCopyMode = **False**
50. Selection.ClearContents
51. ActiveWindow.SmallScroll Down:=0
52. Range("J1").**Select**
53. ActiveCell.FormulaR1C1 = "Inventory (Projected)"
54. Range("I13:I16").**Select**
55. Selection.Copy
56. Range("J13").**Select**
57. ActiveSheet.Paste
58. Range("I14:I16").**Select**
59. Application.CutCopyMode = **False**
60. Selection.ClearContents
61. Range("I24").**Select**
62. **End** **Sub**

65. **Sub** Createproductiondemandlinechart()
66. '
67. ' Createproductiondemandlinechart Macro
68. '
70. '
71. Range("A1:C16,G1:H16").**Select**
72. Range("G1").Activate
73. ActiveSheet.Shapes.AddChart2(332, xlLineMarkers).**Select**
74. ActiveChart.SetSourceData Source:=Range( \_
75. "'AUX Data'!$A$1:$C$16,'AUX Data'!$G$1:$H$16")
76. ActiveChart.Parent.Cut
77. Sheets("AUX Report").**Select**
78. Range("D21").**Select**
79. ActiveSheet.Paste
80. ActiveSheet.ChartObjects("Chart 5").Activate
81. ActiveChart.PlotArea.**Select**
82. ActiveChart.ChartArea.**Select**
83. ActiveChart.ChartTitle.**Select**
84. Selection.Delete
85. ActiveSheet.ChartObjects("Chart 5").Activate
86. ActiveChart.FullSeriesCollection(1).**Select**
87. ActiveChart.FullSeriesCollection(2).**Select**
88. ActiveChart.FullSeriesCollection(3).**Select**
89. ActiveChart.FullSeriesCollection(4).**Select**
90. Selection.MarkerStyle = 1
91. ActiveChart.FullSeriesCollection(3).**Select**
92. ActiveChart.FullSeriesCollection(4).**Select**
94. **End** **Sub**
95. **Sub** colorchangelinechart()
96. '
97. ' color change line Macro
98. '
100. '
101. ActiveSheet.ChartObjects("Chart 5").Activate
102. ActiveChart.FullSeriesCollection(3).**Select**
103. ActiveChart.PlotArea.**Select**
104. ActiveChart.FullSeriesCollection(3).**Select**
105. ActiveChart.FullSeriesCollection(3).Points(1).**Select**
106. ActiveChart.FullSeriesCollection(1).**Select**
107. Selection.MarkerStyle = 8
108. ActiveChart.FullSeriesCollection(2).**Select**
109. ActiveChart.FullSeriesCollection(3).**Select**
110. Selection.MarkerStyle = 1
112. **End** **Sub**
114. **Sub** leftoverediting()
115. '
116. ' leftoverediting Macro
117. '
119. '
120. ActiveSheet.ChartObjects("Chart 5").Activate
121. ActiveChart.FullSeriesCollection(4).**Select**
122. Range("N27").**Select**
123. **End** **Sub**

126. **Sub** createinvetorylevelslinechart()
127. '
128. ' createinvetorylevelslinechart Macro
129. '
131. '
132. Range("A1:A16,I1:J16").**Select**
133. Range("I1").Activate
134. ActiveSheet.Shapes.AddChart2(332, xlLineMarkers).**Select**
135. ActiveChart.SetSourceData Source:=Range( \_
136. "'AUX Data'!$A$1:$A$16,'AUX Data'!$I$1:$J$16")
137. Range("G30").**Select**
138. **End** **Sub**
139. **Sub** editinglinechartsecond()
140. '
141. ' editinglinechartsecond Macro
142. '
144. '
145. ActiveSheet.ChartObjects("Chart 1").Activate
146. ActiveSheet.ChartObjects("Chart 1").Activate
147. ActiveChart.Parent.Cut
148. Sheets("AUX Report").**Select**
149. ActiveWindow.SmallScroll Down:=28
150. Range("C40").**Select**
151. ActiveSheet.Paste
152. ActiveSheet.ChartObjects("Chart 1").Activate
153. ActiveSheet.ChartObjects("Chart 1").Activate
154. ActiveSheet.ChartObjects("Chart 1").Activate
155. ActiveSheet.ChartObjects("Chart 1").Activate
156. ActiveChart.FullSeriesCollection(1).**Select**
157. ActiveChart.FullSeriesCollection(1).Points(1).**Select**
158. ActiveChart.FullSeriesCollection(2).**Select**
159. **End** **Sub**
161. **Sub** OpenAUXReportSheet()
162. ' Go to Aux report
163. Sheets("AUX Report").Activate
164. **End** **Sub**

167. **Sub** savingaspdf()
168. ' Generate the file name with date and time
169. **Dim** fileName **As** **String**
170. fileName = "PPP\_Report\_" & Format(Now, "mmddyyyy\_hhmm") & ".pdf"
172. ' Export the sheet as PDF
173. Sheets("AUX Report").ExportAsFixedFormat Type:=xlTypePDF, \_
174. fileName:=fileName, Quality:=xlQualityMinimum, \_
175. IncludeDocProperties:=**True**, IgnorePrintAreas:=**False**, OpenAfterPublish:=**False**
177. ' Display message box with file information
178. MsgBox "This report has been saved to the following directory:" & vbCrLf & Application.ThisWorkbook.Path & "\" & fileName, vbInformation, "Report Name and Directory"
179. **End** **Sub**
181. **Sub** HideSheet()
182. ' Hiding Aux data sheet
183. ThisWorkbook.Sheets("AUX Data").Visible = xlSheetHidden
184. **End** **Sub**

Citation

When searching how to copy VBA code with highlights (syntax highlighter)  
Reference:  
Caluori, N. (n.d.). Syntax Highlighter for Word - K26. <https://syntax-highlighter.k26.ch/#>

When prompted with “How to make this code save to pdf to directory ” (OpenAI, 2024)  
Reference:  
Chatgpt. (n.d.). <https://chat.openai.com>

When prompted with “How to hide sheet in excel vba” (OpenAI, 2024)  
Reference:  
Chatgpt. (n.d.). <https://chat.openai.com>