



```

1 Sub AllStocksAnalysisRefactored()
2   'Initialize variables for starting price and ending
  price
3   Dim startTime As Single
4   Dim endTime As Single
5
6   'Input box
7   yearValue = InputBox("What year would you like to r
  un the analysis on?")
8   'Start clock
9   startTime = Timer
10
11  'Format the output sheet on All Stocks Analysis wor
  ksheet
12  Worksheets("All Stocks Analysis").Activate
13  Range("A1").Value = "All Stocks (" + yearValue +
  ")"
14
15  'Create a header row
16  Cells(3, 1).Value = "Ticker"
17  Cells(3, 2).Value = "Total Daily Volume"
18  Cells(3, 3).Value = "Return"
19
20
21
22  'Initialize array of all tickers
23  Dim tickers(11) As String
24  tickers(0) = "AY"
25  tickers(1) = "CSIQ"
26  tickers(2) = "DQ"
27  tickers(3) = "ENPH"
28  tickers(4) = "FSLR"
29  tickers(5) = "HASI"
30  tickers(6) = "JKS"
31  tickers(7) = "RUN"
32  tickers(8) = "SEDG"
33  tickers(9) = "SPWR"
34  tickers(10) = "TERP"
35  tickers(11) = "VSLR"
36
37  'Activate data worksheet
38  Sheets(yearValue).Activate
39
40  'Get number of rows to count
41  RowCount = Cells(Rows.Count, "A").End(xlUp).Row
42
43  '1a) Create a ticker Index and set it equal to zero
44  For i = 0 To 11
45    tickerIndex = tickers(i)
46
47
48  '1b)Create Output arrays
49  Dim tickervolume As Long
50  Dim tickerStartingPrices As Double
51  Dim tickerEndingPrices As Double
52
53  '2a)Initialize tickervolume equal zero
54  tickervolume = 0

```

```

1 Sub AllStocksAnalysisRefactored2()
2   'Initialize variables for starting price and ending
  price
3   Dim startTime As Single
4   Dim endTime As Single
5
6   'Input box
7   yearValue = InputBox("What year would you like to r
  un the analysis on?")
8   'Start clock
9   startTime = Timer
10
11  'Format the output sheet on All Stocks Analysis wor
  ksheet
12  Worksheets("All Stocks Analysis").Activate
13  Range("A1").Value = "All Stocks (" + yearValue +
  ")"
14
15  'Create a header row
16  Cells(3, 1).Value = "Ticker"
17  Cells(3, 2).Value = "Total Daily Volume"
18  Cells(3, 3).Value = "Return"
19
20
21
22  'Initialize array of all tickers
23  Dim tickers(11) As String
24  tickers(0) = "AY"
25  tickers(1) = "CSIQ"
26  tickers(2) = "DQ"
27  tickers(3) = "ENPH"
28  tickers(4) = "FSLR"
29  tickers(5) = "HASI"
30  tickers(6) = "JKS"
31  tickers(7) = "RUN"
32  tickers(8) = "SEDG"
33  tickers(9) = "SPWR"
34  tickers(10) = "TERP"
35  tickers(11) = "VSLR"
36
37  'Activate data worksheet
38  Sheets(yearValue).Activate
39
40  'Get number of rows to count
41  RowCount = Cells(Rows.Count, "A").End(xlUp).Row
42
43  '1a) Initialize and set tickerindex = 0
44  tickerIndex = 0
45
46
47  '1b)Create Output arrays
48  Dim tickerVolumes(12) As Long
49  Dim tickerStartingPrices(12) As Single
50  Dim tickerEndingPrices(12) As Single
51
52  '2a)Initialize tickervolume equal zero
53  For i = 0 To 11

```

```

55
56 'Activate sheet
57 Sheets(yearValue).Activate
58
59 '2b) Loop through all rows to find if ticker matches
60 For j = 2 To RowCount
61
62     If Cells(j, 1).Value = tickerIndex Then
63
64         '3a) Increase ticker volume and add ticker volume to
        current ticker
65         tickervolume = tickervolume + Cells(j,
66         8).Value
67
68         'End if
69     End If
70
71 '3b) If this row is first row with current ticker
72     If Cells(j, 1).Value = tickerIndex And Cells
73     (j - 1, 1).Value <> tickerIndex Then
74
75         'Assign ticker starting price to corresponding
        variable
76         tickerStartingPrices = Cells(j, 6).Value
77
78 '3c) If current row is last row to include ticker
79     ElseIf Cells(j, 1).Value = tickerIndex And Ce
80     lls(j + 1, 1).Value <> tickerIndex Then
81
82         'Assign ticker ending price to variable
83         tickerEndingPrices = Cells(j, 6).Value
84
85 'Increase ticker index if next row doesn't matc
86 h current row
87 Else
88
89     'End if
90 End If
91 Next j
92
93 '4) Loop through your arrays and output the Ticker,
94 Total Daily Volume, and Return.
95
96 Worksheets("All Stocks Analysis").Activate
97
98 Cells(4 + i, 1).Value = tickerIndex
99 Cells(4 + i, 2).Value = tickervolume
100 Cells(4 + i, 3).Value = tickerEndingPrices / ti
101 ckerStartingPrices - 1
102
103 Next i
104
105 'Formatting
106 Worksheets("All Stocks Analysis").Activate
107 Range("A3:C3").Font.FontStyle = "Bold Italic"

```

```

53     tickerVolumes(i) = 0
54 Next i
55
56 '2b) Loop through all rows to find if ticker matches
57 For j = 2 To RowCount
58
59     '3a) Increase current ticker volume
60     tickerVolumes(tickerIndex) = tickerVolumes(tick
61     erIndex) + Cells(j, 8).Value
62
63 '3b) If this row is first row with current ticker
64     If Cells(j, 1).Value = tickers(tickerIndex) And
65     Cells(j - 1, 1).Value <> tickers(tickerIndex) Then
66
67         'Assign ticker starting price to corresponding
        variable
68         tickerStartingPrices(tickerIndex) = Cells
69         (j, 6).Value
70
71 '3c) If current row is last row to include ticker
72     If Cells(j, 1).Value = tickers(tickerIndex) And
73     Cells(j + 1, 1).Value <> tickers(tickerIndex) Then
74
75         'Assign ticker ending price to variable
76         tickerEndingPrices(tickerIndex) = Cells(j,
77         6).Value
78         tickerIndex = tickerIndex + 1
79
80 'Increase ticker index if next row doesn't matc
81 h current row
82 Else
83
84     'End if
85 End If
86 Next j
87
88 '4) Loop through your arrays and output the Ticker,
89 Total Daily Volume, and Return.
90
91 For k = 0 To 11
92     Worksheets("All Stocks Analysis").Activate
93
94     Cells(4 + k, 1).Value = tickers(k)
95     Cells(4 + k, 2).Value = tickerVolumes(k)
96     Cells(4 + k, 3).Value = tickerEndingPrices(k) /
97     tickerStartingPrices(k) - 1
98
99 Next k
100
101 'Formatting
102 Worksheets("All Stocks Analysis").Activate
103 Range("A3:C3").Font.FontStyle = "Bold Italic"

```

```

103      Range("A3:C3").Borders(xlEdgeBottom).LineStyle
      = xlContinuous
104      Range("A3:C3").Font.Size = 14
105      Range("B4:B15").NumberFormat = "#,##0"
106      Range("C4:C15").NumberFormat = "0.0%"
107      Columns("B").AutoFit
108
109      dataRowStart = 4
110      dataRowEnd = 15
111
112      For k = dataRowStart To dataRowEnd
113
114          If Cells(k, 3) > 0 Then
115              Cells(k, 3).Interior.Color = vbGreen
116          ElseIf Cells(k, 3) < 0 Then
117              Cells(k, 3).Interior.Color = vbRed
118          Else
119              Cells(i, 3).Interior.Color = xlNone
120      'End if
121      End If
122
123      Next k
124
125      'Stop clock
126      endTime = Timer
127      MsgBox "This code ran in" & (endTime - startTime) &
      "seconds for the year" & (yearValue)
128
129  End Sub
130

```

```

97      Range("A3:C3").Borders(xlEdgeBottom).LineStyle
      = xlContinuous
98      Range("A3:C3").Font.Size = 14
99      Range("B4:B15").NumberFormat = "#,##0"
100     Range("C4:C15").NumberFormat = "0.0%"
101     Columns("B").AutoFit
102
103     dataRowStart = 4
104     dataRowEnd = 15
105
106     For l = dataRowStart To dataRowEnd
107
108         If Cells(l, 3) > 0 Then
109             Cells(l, 3).Interior.Color = vbGreen
110         ElseIf Cells(l, 3) < 0 Then
111             Cells(l, 3).Interior.Color = vbRed
112         Else
113             Cells(l, 3).Interior.Color = xlNone
114     'End if
115     End If
116
117     Next l
118
119     'Stop clock
120     endTime = Timer
121     MsgBox "This code ran in" & (endTime - startTime) &
    "seconds for the year" & (yearValue)
122
123 End Sub

```