Module 2 Challenge – All Stock Analysis VBA Analysis

Date: 10/02/2020

Student Name: Soona Cheon

1. Overview of the project:

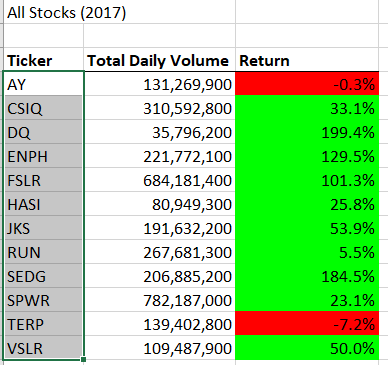
The purpose of the project is using the VBA coding to analyze the stock performance (=return) of 2017, 2018. The 2017 data consists 3004 stock transaction for 12 tickers, and the 2018 works, and 2018 Stock transaction consists 3013 records for the same 12 tickers for comparison.

In this report, I will analyze:

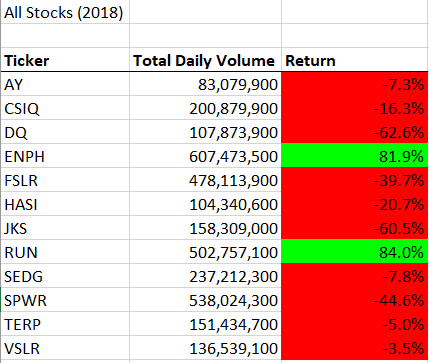
1. Data Analysis
   1. total volume of each ticket in 2017 and 2018
   2. return of each ticker analysis of 2017, and 2018
2. VBA Code Analysis
   1. Performance comparison of AllStockAnalysis Code vs. AllStocksAnalysisRefactored Code
3. Analysis and Challenges
4. The Benefit of VBA
   1. Compare to Excel Pivot Table and Chart, VBA code is more dynamic
   2. By using the VBA, the data analysis was much easier and fast.
5. Useful VBA Code
   1. It is useful to create the button, which can clear the worksheets and run the modules on the specific worksheet
   2. It is also useful to add the function to select “year” to analyze on the worksheets
   3. It is also very useful to add the timer function to analyze the VBA code performance.
6. Challenges
   1. There are many different coding styles in VBA, and the refactored VBA challenge showed that the wrong design causes the slowness of the code performance to process the same data.
7. Results
8. **Data Analysis Outcome**
   * There are 12 tickers:

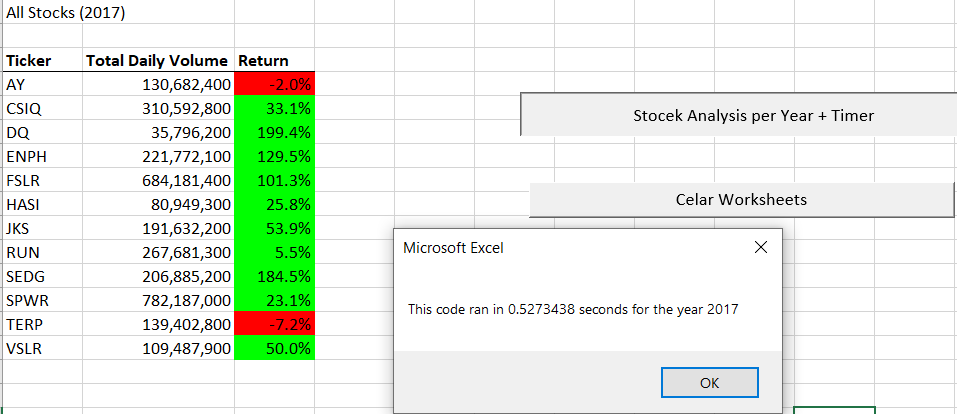
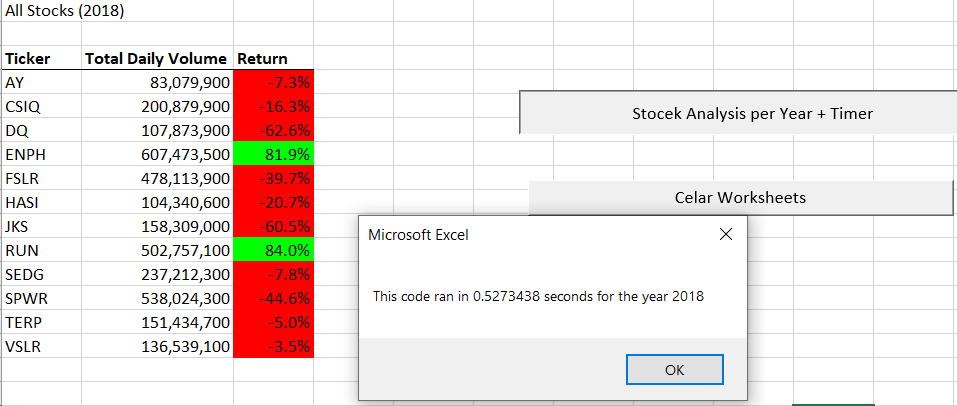
|  |
| --- |
| AY |
| CSIQ |
| DQ |
| ENPH |
| FSLR |
| HASI |
| JKS |
| RUN |
| SEDG |
| SPWR |
| TERP |
| VSLR |

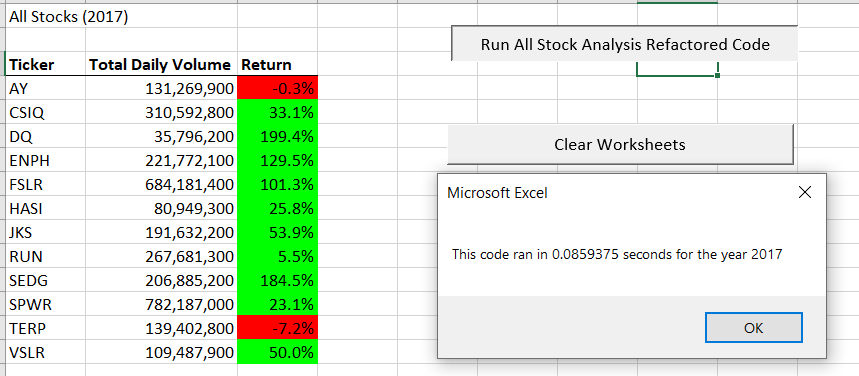
* + The performance of 2017 is:

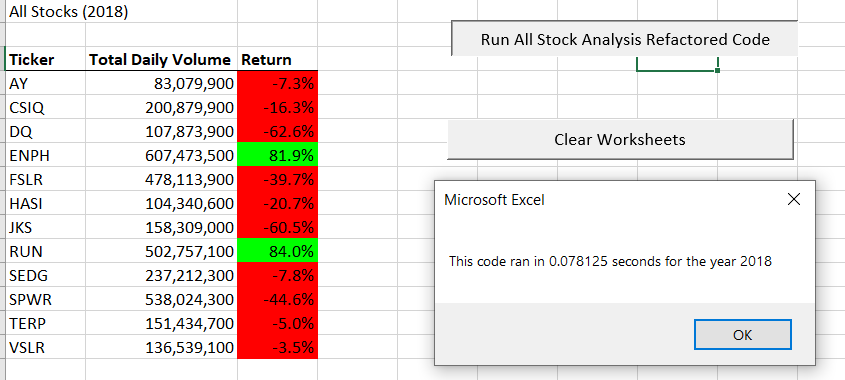


* + The performance of 2018 is:



1. VBA Code Analysis – before refactored
   * The performance of the AllStockAnalysis VBA code before refactored is as below, which was exactly same:
     + 2017 Data Processing Time: 0.5273438 sec
     + 2018 Data Processing Time: 0.5273438 sec
   * 
   * 
2. VBA refactored Code:
   * Changes: In the code, we added the tickerIndex variable, and 3 arrays (tickerVolumes, tickerStatingPrices, ticketEndingPrices), and rewrote the programming logic.
   * **Benefits**: Performance is much faster than before as below
     + 2017 Data Processing Time: 0.0859375 sec
     + 2018 Data Processing Time: 0.078125 sec





* Cons: we need to rewrite the code, by adding arrays and changing the structure of the code, and it is time consuming.

1. Conclusion:
   1. By rewriting the code, we can enhance the performance.

|  |  |  |
| --- | --- | --- |
| Data (Year) | Original Code Speed | Refactored Code Speed ☺ |
| 2017 | 0.5273438 sec | 0.0859375 sec |
| 2018 | 0.5273438 sec | 0.078125 sec |

* 1. If we need to create a complicated code with big data, we should refactor the code to improve code processing time (performance time).
  2. If we add more calculations, arrays or/and variables, it will slow down the VBA code. DRY and light code structure should be the best practice in coding.

Thank you!