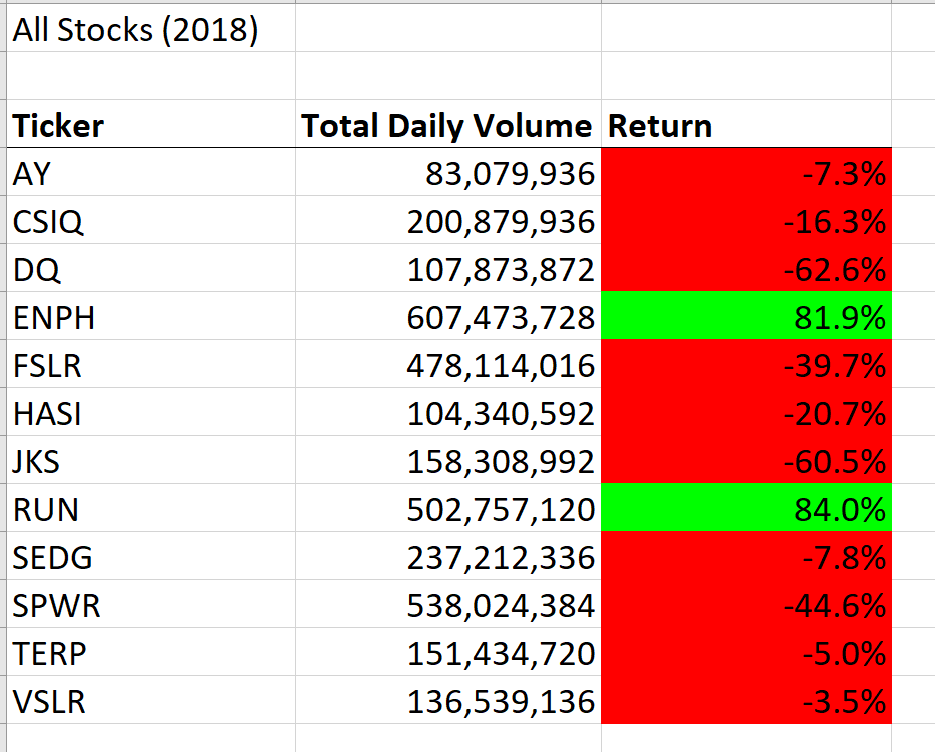
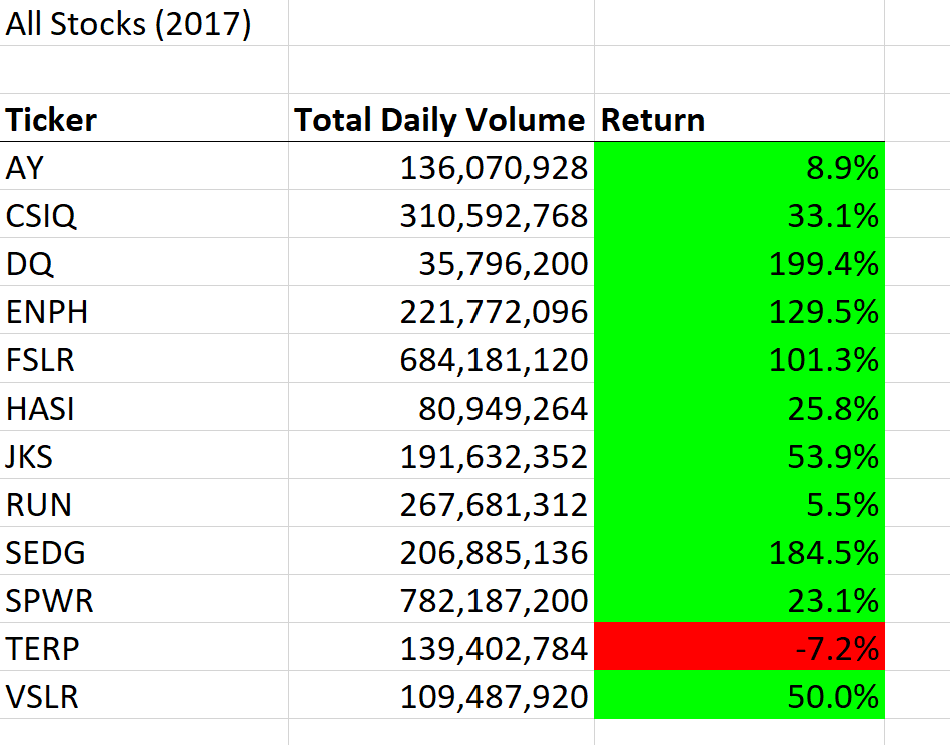
**Written Analysis of AllStocksAnalysis () Function**

The full overview of AllStocksAnalysis, is creating a summary worksheet in Excel that analyzes a series of stocks described in a large table. This is performed with VBA coming up with starting and ending prices, daily volume of stock, through its symbolic “ticker”. The further analysis involves learning about refactoring and how to apply it in a practical situation. This is performed through comparisons between two similar macros: AllStocksAnalysis() and AllStocksAnalysisRefactored().

There are two years of stocks present in the workbook: 2017, and 2018. The behaviour of the financial market between the two years is significantly different, making a strong variance for potential investors.

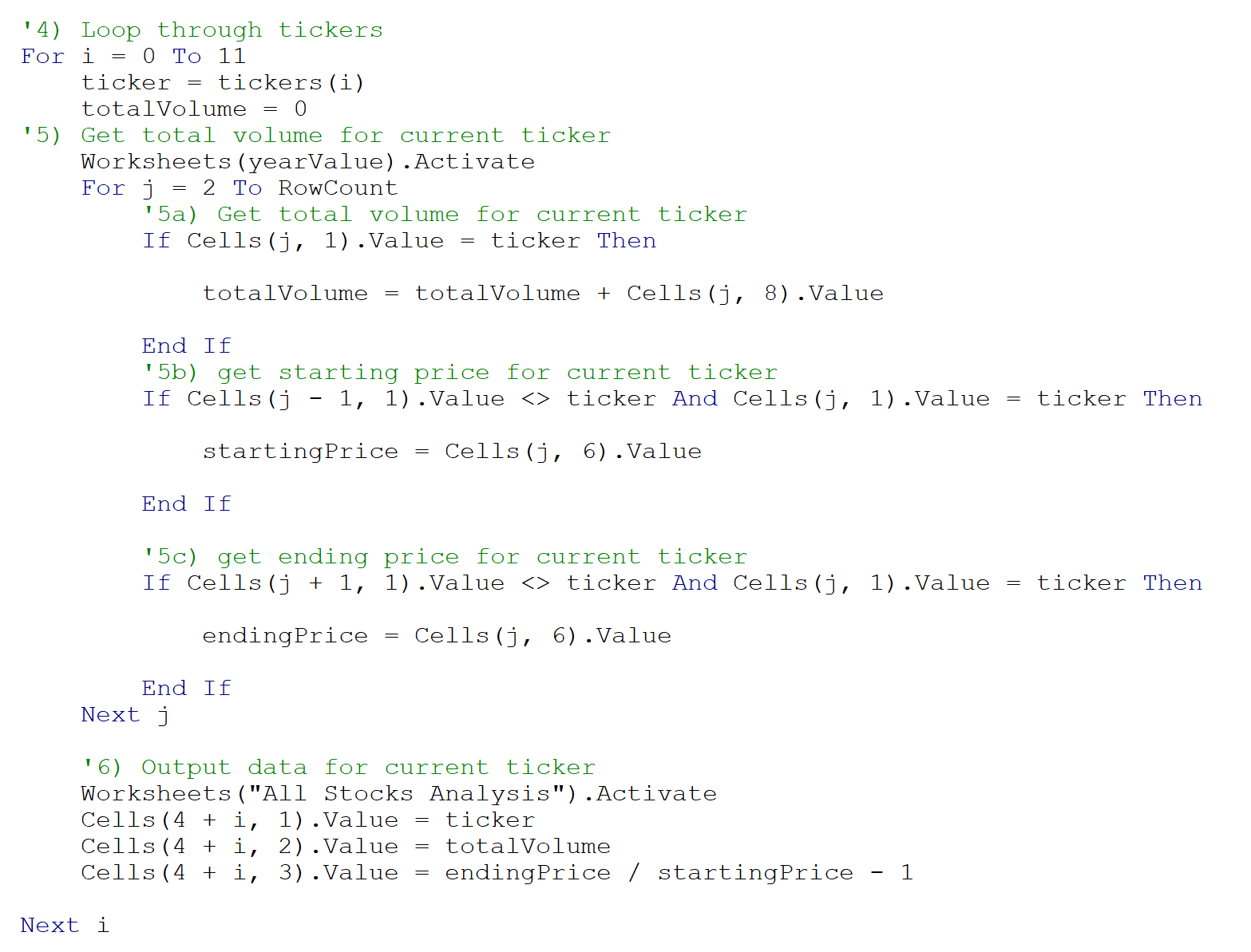


The majority of stocks were increasing in year 2017 but are mostly dropping in year 2018. The colour scheme of green and red makes this easy to visualize.

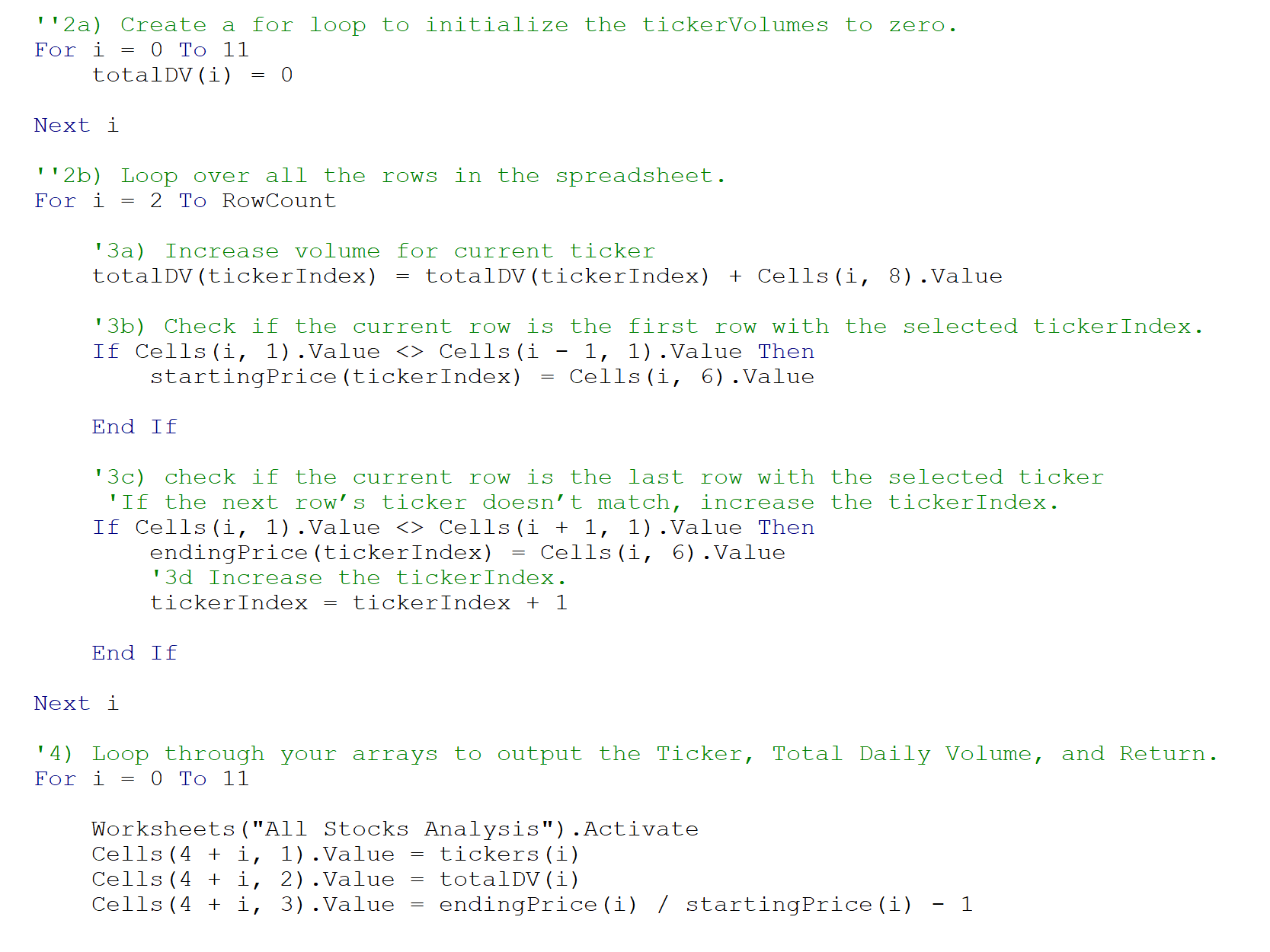
|  |  |  |
| --- | --- | --- |
| **Exchange Year** | **Initial Time (s)** | **Refactored Time (s)** |
| **2017** | 0.765 | 0.133 |
| **2018** | 0.758 | 0.148 |

The refactoring process sped up the calculation by more than 5 times in each case.

The original code, whilst being brief, contains a nested FOR statement and carrying three separate IF statements. This involves cycling unnecessarily through long lines of tables, collecting the necessary data: 11 tickers with 3013 rows gives 11x3013 = 33143 (a lot).



The refactored code has a single For statement for the majority of the calculations which should eliminate the redundancy experienced by the first draft of code.

Over

Overall this refactoring process took about 1/5th the amount of time the original one took so for a large-scale stock operation it would be well-worth the change.

In summary, refactoring code can greatly reduce calculation time if lucky. Unfortunately, it also holds the potential to introduce manual errors like “wrong variables” or the calculation would not be greatly reduced proportional to the amount of time invested by the person refactoring.

Specifically, in this example calculation time was greatly reduced though technically for the size of this Excel sheet it’s not “super-important”. All-in-all the process worked very well.