Traders Island

MOVING AVERAGES

Application II

Version 2.1

Traders Island

Application II:

Spreadsheet

Version 2.1

I – Lesson 1: Moving Average Spreadsheet Application
I.1 - Overview

- The spreadsheet “Moving Average.xlsm” has been fully automated to perform SMA and EMA calculation, chart the price vs MA, build Strategy I in detail, and implement strategy back-testing using Strategy I.
- It performs the same functions that the platform does (and a bit more), albeit it is built in Excel © and can be downloaded on your device.
- It requires Excel © and Python 2.7 (https://www.python.org/downloads/release/python-2710/) installations. For development purposes*, you must have VBA developer up and running on Excel © Workbooks.
- The spreadsheet currently implements the following functionalities.
  - MA implementation using Excel, VBA* and Python
  - Model Testing
  - Charting
  - Step-by-Step Strategy I Implementation
  - Strategy Backtesting

* For more on the VBA code, please refer to Technical Analysis Spreadsheet Guide_MA.docx

Worksheets

The worksheets are fully automated. The “Input” worksheet, described below, serves as a control panel for all other worksheets (except “README”).

- “README”: spreadsheet overview including colour coding.
- “Input”: input data and implementation/testing buttons
- “Historical Data”: historical data set according to dates on “Input” sheet.
- “MA”: MA implementation using different methodologies/approaches.
- “Charts”: price vs indicator/oscillator (moving average here) chart.
- “Testing”: model testing as crucial parameters change.
- “Strategy Backtesting”: optimised parameters and maximised returns.
- “Strategy”: step-by-step implementation of Strategy I.

I – Lesson 1: Moving Average Spreadsheet Application
I.3 - “README” and “Historical Data” Worksheets: Overview

“README”

- This worksheet briefly explains the functionality of each of the remaining worksheets.
- It also explains the sheet’s colour coding.

“Historical Data”

- This is generated via the “Input” sheet (Yahoo! Finance feed) following the below steps.
  - Change the stock symbol in C1.
  - Update Historical Data Start and End Dates in cells C8 and C9 respectively.
  - Click <Load Historical Prices into the "Historical Data" Sheet>.
- The data on the “Historical Data” worksheet can be overwritten but make sure you follow the existing column order: Date/Open/High/Low/Close/Volume/Adj Close.

II – Lesson 2: "Input" Worksheet
II.1 – Overview and Generic Parameters I

- User input is fed into the system, excluding the "Testing" worksheet, through this worksheet.
- User input data – Generic Parameters I: These parameters are required for all functionality.

Key Cell	Parameter	Description
C1	Stock Symbol	Stock Exchange ticker
C5	Current Date	Equivalent of "Today's" date
C8	Historical Data Start Date	Start date for the complete data set
C9	Historical Data End Date	End date for the complete data set
C12	Analysis Data Start Date	Start date of analysis period
C13	Analysis Data End Date	End date of analysis period
Col E, 5 on	Holiday Dates	Market holidays for the complete data set

- A quick note on the difference between Historical and Analysis Data dates: you may prefer to experiment with parts of the data set and the whole data set (bounded by Historical Data Dates), so the Analysis Data date set allows you to extract a smaller data set from the existing one to work with.

- User input data – Generic Parameters IIA: select the checkbox(es) with the MA methodology that you would like to work with.
- Table 1: MA Calculation (Check Boxes 1)

Check Box Text	Description
SMA via Excel (Manual)	Calculate SMA using Excel formulae
EMA via Excel (Manual)	Calculate EMA using Excel Formulae
SMA VBA Function	Calculate SMA using VBA code
EMA VBA Function	Calculate EMA using VBA code
SMA Python Function	Calculate SMA using Python code
EMA Python Function	Calculate EMA using Python code

- User input data – Generic Parameters IIB: select the checkbox(es) with the MA speed improvement that you would like to work with. These functions use tools that speed up the calculation.
- Table 2: MA Speed Improvement (Check Boxes 2)

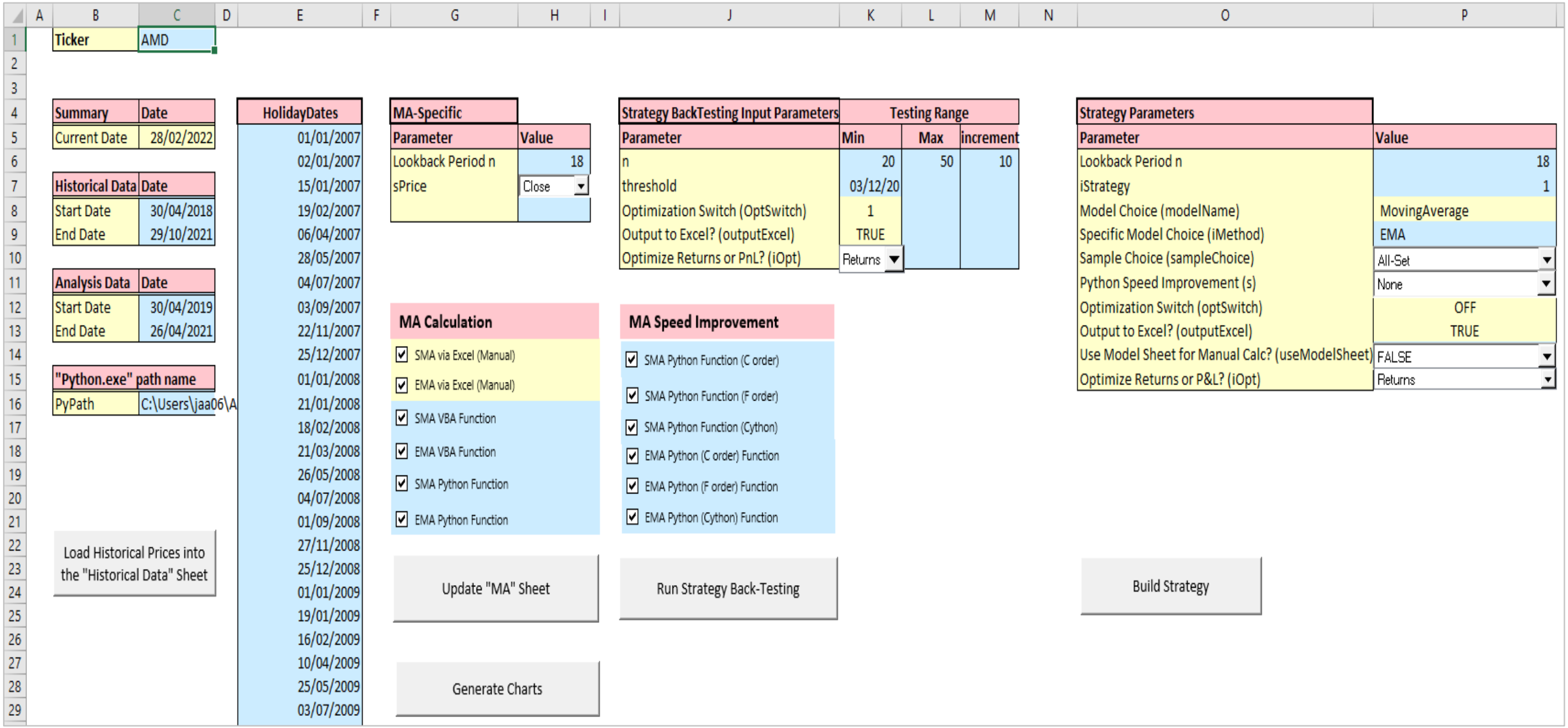
Check Box Text	Description
SMA Python Function (C order)	Calculate SMA using Python code + C compiler
SMA Python Function (F order)	Calculate SMA using Python code + Fortran compiler
SMA Python Function (Cython)	Calculate SMA using Python/Cython code
EMA Python (C order) Function	Calculate EMA using Python code + C compiler
EMA Python (F order) Function	Calculate EMA using Python code + Fortran compiler
EMA Python (Cython) Function	Calculate EMA using Python/Cython code

Generating Processes

- This worksheet is also the “door” to calling implementation and testing procedures. You can do so by clicking the corresponding buttons on the worksheet.
- For the purpose of this tutorial, only the following buttons are relevant.

Button Name	Action
<Load Historical Prices into the "Historical Data" Sheet>	Loads data from Yahoo!
<Update “MA" Sheet>	Creates MA sheet
<Generate Charts>	Creates Price and MA Chart
<Strategy BackTesting>	Back-tests a Trading Strategy
<Build Strategy>	Step-by-step Strategy I

II – Lesson 2: "Input" Worksheet
II.5 – Screenshot



Ticker	AMD

Summary	Date
Current Date	28/02/2022

Historical Data	Date
Start Date	30/04/2018
End Date	29/10/2021

Analysis Data	Date
Start Date	30/04/2019
End Date	26/04/2021

"Python.exe" path name	
PyPath	C:\Users\jaa06\A

HolidayDates
01/01/2007
02/01/2007
15/01/2007
19/02/2007
06/04/2007
28/05/2007
04/07/2007
03/09/2007
22/11/2007
25/12/2007
01/01/2008
21/01/2008
18/02/2008
21/03/2008
26/05/2008
04/07/2008
01/09/2008
27/11/2008
25/12/2008
01/01/2009
19/01/2009
16/02/2009
10/04/2009
25/05/2009
03/07/2009

MA-Specific	
Parameter	Value
Lookback Period n	18
sPrice	Close

Strategy BackTesting Input Parameters	Testing Range		
Parameter	Min	Max	increment
n	20	50	10
threshold	03/12/20		
Optimization Switch (OptSwitch)	1		
Output to Excel? (outputExcel)	TRUE		
Optimize Returns or PnL? (iOpt)	Returns		

Strategy Parameters	
Parameter	Value
Lookback Period n	18
iStrategy	1
Model Choice (modelName)	MovingAverage
Specific Model Choice (iMethod)	EMA
Sample Choice (sampleChoice)	All-Set
Python Speed Improvement (s)	None
Optimization Switch (optSwitch)	OFF
Output to Excel? (outputExcel)	TRUE
Use Model Sheet for Manual Calc? (useModelSheet)	FALSE
Optimize Returns or P&L? (iOpt)	Returns

MA Calculation

- ☑ SMA via Excel (Manual)
- ☑ EMA via Excel (Manual)
- ☑ SMA VBA Function
- ☑ EMA VBA Function
- ☑ SMA Python Function
- ☑ EMA Python Function

MA Speed Improvement

- ☑ SMA Python Function (C order)
- ☑ SMA Python Function (F order)
- ☑ SMA Python Function (Cython)
- ☑ EMA Python (C order) Function
- ☑ EMA Python (F order) Function
- ☑ EMA Python (Cython) Function

Load Historical Prices into the "Historical Data" Sheet

Update "MA" Sheet

Run Strategy Back-Testing

Build Strategy

Generate Charts

II – Lesson 2: “Input” Worksheet
II.6 – Map

- Generic Input: you need to do the following before initiating any action
  - Enter Stock Symbol: C1
  - Enter Historical Data Start Date (Whole data set): C8
  - Enter Historical Data End Date (Whole data set): C9
  - Enter Analysis Data Start Date (Subset data set): C12
  - Enter Analysis Data End Date (Subset data set): C13
  - Enter Holiday Dates: starting E4
  - Click <Load Historical Prices into the "Historical Data" Sheet>
- Specific Input:
  - MA specific: the data in this table (H6:H7) is relevant for any procedures related to MA computation. You only need to update H7, if desired.
  - Strategy BackTesting specific: the data here (K6:M9) is relevant for strategy back-testing processes.
  - Strategy parameters: the data in this block (P6:P15) is used to build Strategy I and illustrate its workings.
  - MA Calculation and MA Speed Improvement function selection: the checkboxes coinciding in Columns G:H and K allow the user to determine which functions to use.
- The following two slides explain model and testing parameters in the “Input” worksheet. The slides to follow will delve into the details of how to run implementation and testing procedures.

MA-Specific Parameters

- These parameters are specific to “MA” calculation, but you need to set the sPrice to be able to run strategy back-testing (and build the strategy sheet too).

Key Cell	Parameter	Description
H6	Lookback Period n	Moving average period
H7	sPrice	Open/High/Low/Close/Volume/Adj Close

Strategy BackTesting Input Parameters

- These parameters are specific to the strategy back-test calculation, and you need to set them before clicking <Run Strategy BackTesting>.

Key Cell	Parameter	Description
K6:M6	n	Lookback period range + increment
K7	threshold	Date dividing the data sample
K8	Optimization Switch (OptSwitch)	Fixed at 1 (ON)
K9	Output to Excel? (outputExcel)	Output intermediate results to Excel
K10	Optimise Returns or PnL? (iOpt)	Optimise using Returns or PnL

II – Lesson 2: “Input” Worksheet
II.8 - “Strategy” Worksheet Parameters

- These parameters are specific to Strategy I implementation, and you need to set them before clicking <Build Strategy>.

Key Cell	Parameter	Description
P6	Lookback Period n	MA period
P7	iStrategy	Only Strategy I is implemented
P8	Model Choice (modelName)	Type of model: MA, RSI…
P9	Specific Model Choice (iMethod)	Specific model: SMA, EMA…
P10	Sample Choice (sampleChoice)	In-, out-of- and complete sample
P11	Python Speed Improvement (s)	None, Cython, C or Fortran compiler
P12	Optimization Switch (optSwitch)	Fixed at 0 (OFF)
P13	Output to Excel? (outputExcel)	Fixed at TRUE
P14	Use Model Sheet for Manual Calc? (useModelSheet)	Uses MA values in the “MA” sheet instead of calculating from scratch
P15	Optimise Returns or P&L? (iOpt)	Redundant (no optimisation)

III – Lesson 3: “MA” Worksheet
III.1 – Guide

- This worksheet displays SMA and EMA values based on the input parameters in the MA-Specific block (H6:H7), and the function choice blocks (MA Calculation, MA Speed Improvement) on the “Input” worksheet.
- It is fully automated, and it is generated through the “Input” worksheet by following the below steps:
  1. Update generic input data.
  2. Update MA-specific data.
  3. Select the function(s) to be used in computing MA. These can be found in the checkboxes in columns G and H.
  4. Select the function(s) with speed improvement in computing MA. They can be found in the checkboxes in column J.
  5. Click <Update “MA” Sheet>.
  6. The “MA” worksheet will be automatically generated including formatting.
  7. Note that there are several functions that implement the same technique. The aim behind that is to give you more than one function to test the implementation. It also widens the spectrum for speed improvement.

III – Lesson 3: “MA” Worksheet
III.2 – Screenshot

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	18		MA_SMA via Excel (Manual)	MA_EMA via Excel (Manual)	MA_SMA VBA Function	MA_EMA VBA Function	MA_SMA Python Function	MA_EMA Python Function	MA_EMA Python (C order) Function	MA_EMA Python (F order) Function	MA_EMA Python (Cython) Function	MA_SMA Python Function (Cython)	MA_SMA Python Function (F order)	MA_SMA Python Function (C order)
2	Date	Close	MA	MA	MA	MA	MA	MA	MA	MA	MA	MA	MA	MA
3	02/01/2018	10.98												
4	03/01/2018	11.55												
5	04/01/2018	12.12												
6	05/01/2018	11.88												
7	08/01/2018	12.28												
8	09/01/2018	11.82												
9	10/01/2018	11.96												
10	11/01/2018	12.14												
11	12/01/2018	12.02												
12	16/01/2018	11.91												
13	17/01/2018	12.18												
14	18/01/2018	12.47												
15	19/01/2018	12.59												
16	22/01/2018	12.65												
17	23/01/2018	12.94												
18	24/01/2018	12.71												
19	25/01/2018	12.41												
20	26/01/2018	12.95	12.20	12.20	12.20	12.20	12.20	12.20	12.20	12.20	12.20	12.20	12.20	12.20
21	29/01/2018	13.32	12.33	12.32	12.33	12.32	12.33	12.32	12.32	12.32	12.32	12.33	12.33	12.33
22	30/01/2018	12.87	12.40	12.37	12.40	12.37	12.40	12.37	12.37	12.37	12.37	12.40	12.40	12.40
23	31/01/2018	13.74	12.49	12.52	12.49	12.52	12.49	12.52	12.52	12.52	12.52	12.49	12.49	12.49
24	01/02/2018	13.25	12.57	12.60	12.57	12.60	12.57	12.60	12.60	12.60	12.60	12.57	12.57	12.57

... | Input | Historical Data | **MA** | Charts | Strategy | Strategy BackTesting | ⊕

- The following two slides show a worked example on how to compute the SMA.
- SMA computation is performed given the following data.

Parameter	Value
Stock Symbol	AMD
Historical Data Start Date*	09/06/2000
Historical Data End Date*	16/05/2019
Analysis Data Start Date	02/01/2018
Analysis Data End Date	10/05/2019
Lookback Period n	10

* Any historical data dates would work as long as they enclose analysis data dates.

- To replicate the worked example, you should follow the below steps:

1. Open “MovingAverage.xlsm” and go to the “Input” worksheet.
2. Type “AMD” in cell C1.
3. Fill in the dates as per the previous slide in cells C8, C9, C12 and C13 respectively.
4. Click the *<1. Load Historical Prices into the "Historical Data" Sheet>* button.
5. Input “10” into cell H6.
6. Under “MA Calculation”, you may tick all “SMA” boxes or just the “SMA via Excel (Manual)” box for quicker calculation.
7. Click the *<Update “MA" Sheet>* button.
8. The code now creates a new “MA” worksheet showing the corresponding results for the chosen functions.

III – Lesson 3: "MA" Worksheet
III.3 – Worked Examples
III.3.3 – SMA Example Output

10 Date	Close	MA_SMA via Excel (Manual) MA	MA_SMA VBA Function MA	MA_SMA Python Function MA	MA_SMA Python Function (Cython) MA	MA_SMA Python Function (F order) MA	MA_SMA Python Function (C order) MA
02/01/2018	10.98						
03/01/2018	11.55						
04/01/2018	12.12						
05/01/2018	11.88						
08/01/2018	12.28						
09/01/2018	11.82						
10/01/2018	11.96						
11/01/2018	12.14						
12/01/2018	12.02						
16/01/2018	11.91	11.87	11.87	11.87	11.87	11.87	11.87
17/01/2018	12.18	11.99	11.99	11.99	11.99	11.99	11.99
18/01/2018	12.47	12.08	12.08	12.08	12.08	12.08	12.08
19/01/2018	12.59	12.13	12.13	12.13	12.13	12.13	12.13
22/01/2018	12.65	12.20	12.20	12.20	12.20	12.20	12.20
23/01/2018	12.94	12.27	12.27	12.27	12.27	12.27	12.27
24/01/2018	12.71	12.36	12.36	12.36	12.36	12.36	12.36

- The following two slides show a worked example on how to compute EMA.
- EMA computation is being done given the following data

Parameter	Value
Stock Symbol	AMD
Historical Data Start Date*	09/06/2000
Historical Data End Date*	16/05/2019
Analysis Data Start Date	02/01/2018
Analysis Data End Date	10/05/2019
Number of Periods	10

* Any historical data dates would work as long as they enclose analysis data dates.

- Note: The input data for this example is the same as that for the previous SMA example.

- To replicate the worked example, you should follow the below steps

1. Open “MovingAverage.xlsm” and go to the “Input” worksheet.
2. Type “AMD” in cell C1.
3. Fill in the dates as per the previous slide in cells C8, C9, C12 and C13 respectively.
4. Click the *<1. Load Historical Prices into the "Historical Data" Sheet>* button.
5. Input “10” into cell H6.
6. Under “MA Calculation”, you may tick all “EMA” boxes or just the “EMA via Excel (Manual)” box for quicker calculation.
7. Click the *<Update “MA" Sheet>* button.
8. The code now creates a new “MA” worksheet showing the corresponding results for the chosen functions.

III – Lesson 3: "MA" Worksheet
III.3 – Worked Examples
III.3.6 – EMA Example Output

10 Date	Close	MA_EMA via Excel (Manual) MA	MA_EMA VBA Function MA	MA_EMA Python Function MA	MA_EMA Python (C order) Function MA	MA_EMA Python (F order) Function MA	MA_EMA Python (Cython) Function MA
02/01/2018	10.98						
03/01/2018	11.55						
04/01/2018	12.12						
05/01/2018	11.88						
08/01/2018	12.28						
09/01/2018	11.82						
10/01/2018	11.96						
11/01/2018	12.14						
12/01/2018	12.02						
16/01/2018	11.91	11.87	11.87	11.87	11.87	11.87	11.87
17/01/2018	12.18	11.92	11.92	11.92	11.92	11.92	11.92
18/01/2018	12.47	12.02	12.02	12.02	12.02	12.02	12.02
19/01/2018	12.59	12.13	12.13	12.13	12.13	12.13	12.13
22/01/2018	12.65	12.22	12.22	12.22	12.22	12.22	12.22
23/01/2018	12.94	12.35	12.35	12.35	12.35	12.35	12.35
24/01/2018	12.71	12.42	12.42	12.42	12.42	12.42	12.42

IV – Lesson 4: “Strategy BackTesting” Worksheet
IV.1 – Parameters

- As we have mentioned earlier, strategy back-testing is dependent on the type of trading strategy used. Currently, only one type of Moving Average strategy, Strategy I, is implemented and back-tested in this spreadsheet.
- In summary, the actual back-testing uses MA functions that are developed in Python, and you can test the validity of these functions’ results by running model testing through the spreadsheet. The spreadsheet acts as a user interface whereby Python code takes input from the user through, and writes back output to, the spreadsheet.
- The following slide displays a screenshot of the “Strategy BackTesting” worksheet.
  - Ranges of input parameters are displayed here according to how they are defined on the “Input” sheet.
  - The worksheet is fully automated, and it is generated through the “Input” worksheet by clicking the <Strategy BackTesting> button.
  - In the background, Python code performs optimization procedures over the three-sample data sets and gives back optimised parameters as well as the corresponding returns for each sample (table starting B11).

IV – Lesson 4: "Strategy BackTesting" Worksheet
IV.2 – Screenshot

	A	B	C	D	E	F	G	H	I	J
1	**Strategy BackTesting Input Parameters**	**Testing Range**								
2	**Parameter**	**Min**	**Max**	**increment**						
3	n	7	9	1						
4	threshold	01/01/19								
5	Optimization Switch (OptSwitch)	1								
6	Output to Excel? (outputExcel)	TRUE								
7	Optimize Returns or PnL? (iOpt)	2								
8	sPrice	Close								
9										
10										

11	MA Optimized Parameters and Returns/PnL	MA_SMA Python Function	MA_EMA Python Function	MA_EMA Python (C order) Function	MA_EMA Python (F order) Function	MA_EMA Python (Cython) Function	MA_SMA Python Function (Cython)	MA_SMA Python Function (F order)	MA_SMA Python Function (C order)
12	**iMethod**	**1**	**2**	**2**	**2**	**2**	**1**	**1**	**1**
13	**Python Speed Improvement Approach (s)**	**0**	**0**	**1**	**2**	**3**	**3**	**2**	**1**
14	**iStrategy**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
15	**n**	**7**	**7**	**7**	**7**	**7**	**7**	**7**	**7**
16	**Optimization Switch (optSwitch)**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
17	**Output to Excel also? (outputExcel)**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**
18	n	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00
19	In-Sample Optimized PnL	13.99	16.01	16.01	16.01	16.01	13.99	13.99	13.99
20	Whole Sample PnL based on Optimized Parameters	-1.50	-6.02	-6.02	-6.02	-6.02	-1.50	-1.50	-1.50
21	Out-Of-Sample PnL based on Optimized Parameters	-13.40	-19.94	-19.94	-19.94	-19.94	-13.40	-13.40	-13.40

22		
23	**Description**	
24	***ONLY Python functions (excl. Manual) are used for Strategy Back-Testing***	
25	**n: Lookback Period**	e.g. 13, 50, 150, 200 days
26	**iMethod: MA Computation Technique**	<> 2 for SMA technique
27		= 2 for EMA technique
28	**s: Type of Speed Improvement**	= 1 for C order
29		= 2 for F order
30		= 3 for Cython
31		else no speed improvement
32	**iStrategy: Trading Strategy**	= 1 for Strategy I (only this strategy is being back-tested now)
33		
34		

README | Input | Historical Data | Testing | Charts | MA | **Strategy BackTesting** | Strategy

- We have seen a screenshot of the strategy back-testing worksheet in the previous slide. There are 3 main sections in this worksheet:
  1. Input Parameters: ranges of input parameters are directly copied , by VBA code, from the “Input” worksheet to this one. You must not change the parameters on this worksheet. Any changes must be done on the “Input” worksheet (cells K6:M8 and K9).
  2. Optimised Returns and Parameters: this is the main output as given by the Python code. The data set is divided into in-sample, where parameters are optimised to produce maximised returns, all data set and out-of-sample data set to test the robustness of the optimised parameters.
     - Optimised Parameter: lookback period.
     - Optimised Returns (or PnL): for all datasets.
  3. Description: the last section on the worksheet describes other parameters on this worksheet and their permitted values.

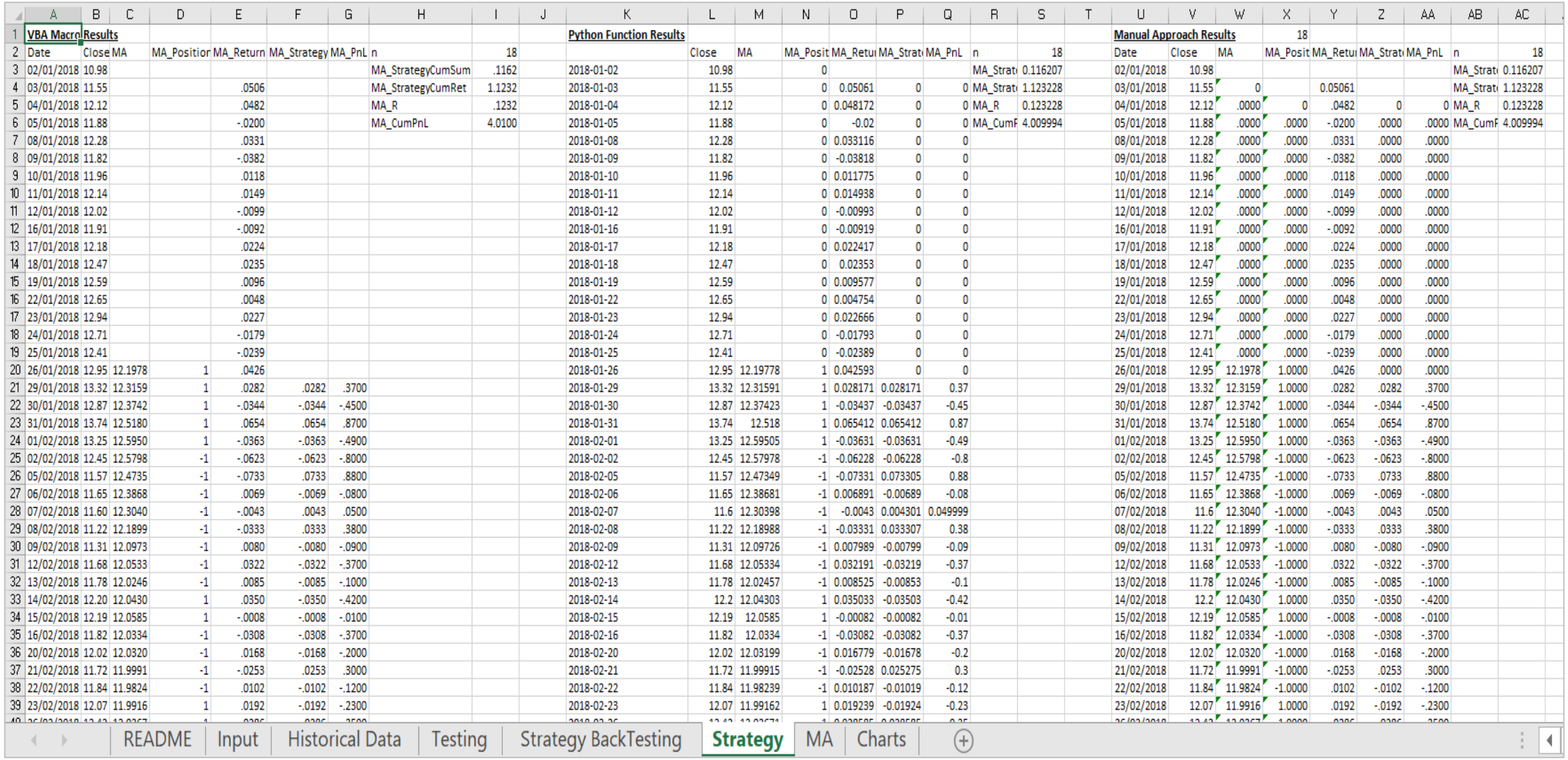
V – Lesson 5: “Strategy” Worksheet
V.1 – Overview

- This worksheet illustrates in detail how Strategy I is built in the code, using the input parameters in the Strategy Parameters block (P6:P15) on the “Input” worksheet.
- The worksheet is fully automated, and it is generated through the “Input” worksheet by clicking the <Build Strategy> button.
- There are three sections in this worksheet:
  - VBA Macro Results: shows the results of implementing this strategy using VBA code.
  - Python Function Results: shows the results of implementing this strategy using Python code.
  - Manual Approach Results: shows a step-by-step implementation using Excel © formulae.
- The three different methodologies are implementing the same strategy, albeit with different tools. So, the outcome of all three approaches should be exactly the same.
- In the next few slides, we will explain inputs and outputs, and show a screenshot of the “Strategy” worksheet.

V – Lesson 5: “Strategy” Worksheet
V.2 – Description

- Each methodology presents the results in two tables.
- The first table illustrates the step-by-step implementation for each observation.
  - Date: observation date, the first of which is defined by the sample choice.
  - Price: e.g. “Close” on that date.
  - MA: moving average as defined by the Input parameters.
  - MA_Position: buy or sell (price crossesing MA from below/above respectively).
  - MA_Return: daily log returns.
  - MA_Strategy: return of the strategy on each observation date.
  - MA_PnL: profit or loss on the position on each observation date.
- The second (smaller) table on the right is a summary table of the final results.
  - n: the lookback period (it is there for clarity purposes).
  - MA_StrategyCumSum and MA_StrategyCumRet: show the cumulative sum and cumulative return of the strategy (refer to documentation on Strategy BackTesting in the corresponding section on the website).
  - MA_R and MA_CumPnL: show the final return and PnL of this strategy.

V – Lesson 5: "Strategy" Worksheet
V.3 – Screenshot



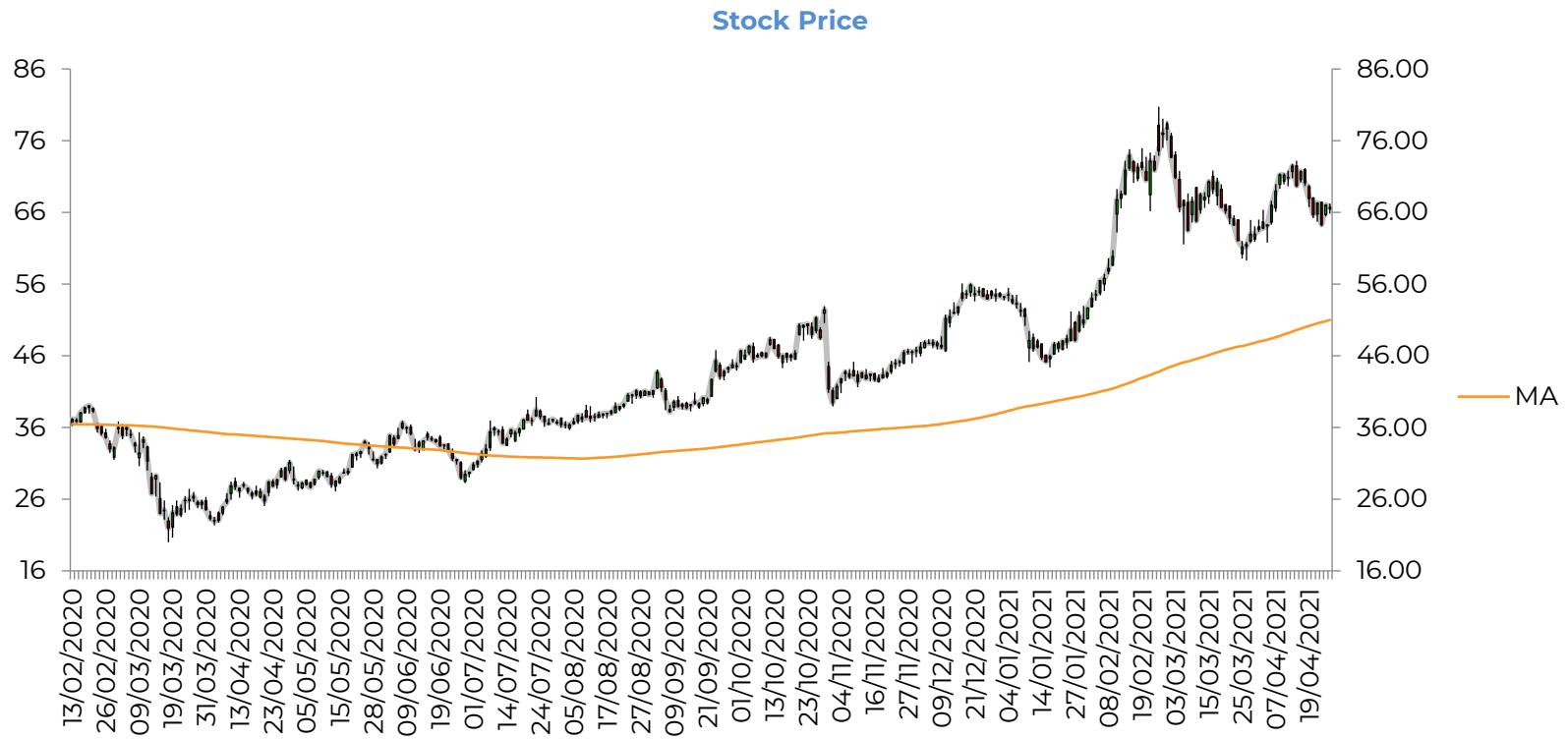
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1	VBA Macro Results										Python Function Results										Manual Approach Results			18					
2	Date	Close	MA	MA_Position	MA_Return	MA_Strategy	MA_PnL	n	18			Close	MA	MA_Posit	MA_Retu	MA_Strat	MA_PnL	n	18		Date	Close	MA	MA_Posit	MA_Retu	MA_Strat	MA_PnL	n	18
3	02/01/2018	10.98						MA_StrategyCumSum	.1162		2018-01-02	10.98		0				MA_Strat	0.116207		02/01/2018	10.98						MA_Strat	0.116207
4	03/01/2018	11.55			.0506			MA_StrategyCumRet	1.1232		2018-01-03	11.55		0	0.05061	0	0	MA_Strat	1.123228		03/01/2018	11.55	0		0.05061			MA_Strat	1.123228
5	04/01/2018	12.12			.0482			MA_R	.1232		2018-01-04	12.12		0	0.048172	0	0	MA_R	0.123228		04/01/2018	12.12	.0000	0	.0482	0	0	MA_R	0.123228
6	05/01/2018	11.88			-.0200			MA_CumPnL	4.0100		2018-01-05	11.88		0	-0.02	0	0	MA_CumF	4.009994		05/01/2018	11.88	.0000	.0000	-.0200	.0000	.0000	MA_CumF	4.009994
7	08/01/2018	12.28			.0331						2018-01-08	12.28		0	0.033116	0	0				08/01/2018	12.28	.0000	.0000	.0331	.0000	.0000		
8	09/01/2018	11.82			-.0382						2018-01-09	11.82		0	-0.03818	0	0				09/01/2018	11.82	.0000	.0000	-.0382	.0000	.0000		
9	10/01/2018	11.96			.0118						2018-01-10	11.96		0	0.011775	0	0				10/01/2018	11.96	.0000	.0000	.0118	.0000	.0000		
10	11/01/2018	12.14			.0149						2018-01-11	12.14		0	0.014938	0	0				11/01/2018	12.14	.0000	.0000	.0149	.0000	.0000		
11	12/01/2018	12.02			-.0099						2018-01-12	12.02		0	-0.00993	0	0				12/01/2018	12.02	.0000	.0000	-.0099	.0000	.0000		
12	16/01/2018	11.91			-.0092						2018-01-16	11.91		0	-0.00919	0	0				16/01/2018	11.91	.0000	.0000	-.0092	.0000	.0000		
13	17/01/2018	12.18			.0224						2018-01-17	12.18		0	0.022417	0	0				17/01/2018	12.18	.0000	.0000	.0224	.0000	.0000		
14	18/01/2018	12.47			.0235						2018-01-18	12.47		0	0.02353	0	0				18/01/2018	12.47	.0000	.0000	.0235	.0000	.0000		
15	19/01/2018	12.59			.0096						2018-01-19	12.59		0	0.009577	0	0				19/01/2018	12.59	.0000	.0000	.0096	.0000	.0000		
16	22/01/2018	12.65			.0048						2018-01-22	12.65		0	0.004754	0	0				22/01/2018	12.65	.0000	.0000	.0048	.0000	.0000		
17	23/01/2018	12.94			.0227						2018-01-23	12.94		0	0.022666	0	0				23/01/2018	12.94	.0000	.0000	.0227	.0000	.0000		
18	24/01/2018	12.71			-.0179						2018-01-24	12.71		0	-0.01793	0	0				24/01/2018	12.71	.0000	.0000	-.0179	.0000	.0000		
19	25/01/2018	12.41			-.0239						2018-01-25	12.41		0	-0.02389	0	0				25/01/2018	12.41	.0000	.0000	-.0239	.0000	.0000		
20	26/01/2018	12.95	12.1978	1	.0426						2018-01-26	12.95	12.19778	1	0.042593	0	0				26/01/2018	12.95	12.1978	1.0000	.0426	.0000	.0000		
21	29/01/2018	13.32	12.3159	1	.0282	.0282	.3700				2018-01-29	13.32	12.31591	1	0.028171	0.028171	0.37				29/01/2018	13.32	12.3159	1.0000	.0282	.0282	.3700		
22	30/01/2018	12.87	12.3742	1	-.0344	-.0344	-.4500				2018-01-30	12.87	12.37423	1	-0.03437	-0.03437	-0.45				30/01/2018	12.87	12.3742	1.0000	-.0344	-.0344	-.4500		
23	31/01/2018	13.74	12.5180	1	.0654	.0654	.8700				2018-01-31	13.74	12.518	1	0.065412	0.065412	0.87				31/01/2018	13.74	12.5180	1.0000	.0654	.0654	.8700		
24	01/02/2018	13.25	12.5950	1	-.0363	-.0363	-.4900				2018-02-01	13.25	12.59505	1	-0.03631	-0.03631	-0.49				01/02/2018	13.25	12.5950	1.0000	-.0363	-.0363	-.4900		
25	02/02/2018	12.45	12.5798	-1	-.0623	-.0623	-.8000				2018-02-02	12.45	12.57978	-1	-0.06228	-0.06228	-0.8				02/02/2018	12.45	12.5798	-1.0000	-.0623	-.0623	-.8000		
26	05/02/2018	11.57	12.4735	-1	-.0733	.0733	.8800				2018-02-05	11.57	12.47349	-1	-0.07331	0.073305	0.88				05/02/2018	11.57	12.4735	-1.0000	-.0733	.0733	.8800		
27	06/02/2018	11.65	12.3868	-1	.0069	-.0069	-.0800				2018-02-06	11.65	12.38681	-1	0.006891	-0.00689	-0.08				06/02/2018	11.65	12.3868	-1.0000	.0069	-.0069	-.0800		
28	07/02/2018	11.60	12.3040	-1	-.0043	.0043	.0500				2018-02-07	11.6	12.30398	-1	-0.0043	0.004301	0.049999				07/02/2018	11.6	12.3040	-1.0000	-.0043	.0043	.0500		
29	08/02/2018	11.22	12.1899	-1	-.0333	.0333	.3800				2018-02-08	11.22	12.18988	-1	-0.03331	0.033307	0.38				08/02/2018	11.22	12.1899	-1.0000	-.0333	.0333	.3800		
30	09/02/2018	11.31	12.0973	-1	.0080	-.0080	-.0900				2018-02-09	11.31	12.09726	-1	0.007989	-0.00799	-0.09				09/02/2018	11.31	12.0973	-1.0000	.0080	-.0080	-.0900		
31	12/02/2018	11.68	12.0533	-1	.0322	-.0322	-.3700				2018-02-12	11.68	12.05334	-1	0.032191	-0.03219	-0.37				12/02/2018	11.68	12.0533	-1.0000	.0322	-.0322	-.3700		
32	13/02/2018	11.78	12.0246	-1	.0085	-.0085	-.1000				2018-02-13	11.78	12.02457	-1	0.008525	-0.00853	-0.1				13/02/2018	11.78	12.0246	-1.0000	.0085	-.0085	-.1000		
33	14/02/2018	12.20	12.0430	1	.0350	-.0350	-.4200				2018-02-14	12.2	12.04303	1	0.035033	-0.03503	-0.42				14/02/2018	12.2	12.0430	1.0000	.0350	-.0350	-.4200		
34	15/02/2018	12.19	12.0585	1	-.0008	-.0008	-.0100				2018-02-15	12.19	12.0585	1	-0.00082	-0.00082	-0.01				15/02/2018	12.19	12.0585	1.0000	-.0008	-.0008	-.0100		
35	16/02/2018	11.82	12.0334	-1	-.0308	-.0308	-.3700				2018-02-16	11.82	12.0334	-1	-0.03082	-0.03082	-0.37				16/02/2018	11.82	12.0334	-1.0000	-.0308	-.0308	-.3700		
36	20/02/2018	12.02	12.0320	-1	.0168	-.0168	-.2000				2018-02-20	12.02	12.03199	-1	0.016779	-0.01678	-0.2				20/02/2018	12.02	12.0320	-1.0000	.0168	-.0168	-.2000		
37	21/02/2018	11.72	11.9991	-1	-.0253	.0253	.3000				2018-02-21	11.72	11.99915	-1	-0.02528	0.025275	0.3				21/02/2018	11.72	11.9991	-1.0000	-.0253	.0253	.3000		
38	22/02/2018	11.84	11.9824	-1	.0102	-.0102	-.1200				2018-02-22	11.84	11.98239	-1	0.010187	-0.01019	-0.12				22/02/2018	11.84	11.9824	-1.0000	.0102	-.0102	-.1200		
39	23/02/2018	12.07	11.9916	1	.0192	-.0192	-.2300				2018-02-23	12.07	11.99162	1	0.019239	-0.01924	-0.23				23/02/2018	12.07	11.9916	1.0000	.0192	-.0192	-.2300		

README | Input | Historical Data | Testing | Strategy BackTesting | **Strategy** | MA | Charts

VI – Lesson 6: “Charts” Worksheet
VI.1 – Overview

- This worksheet displays a chart for the price vs the moving average.
- It is fully automated and can be generated through the “Input” sheet by clicking the <Generate Charts> button.
- It relies on the “MA” sheet and the “Historical Data” sheet so make sure ensure that these are properly set-up before generating the chart.
- The following slide displays a chart example.

VI – Lesson 6: “Charts” Worksheet
VI.2 – Screenshot

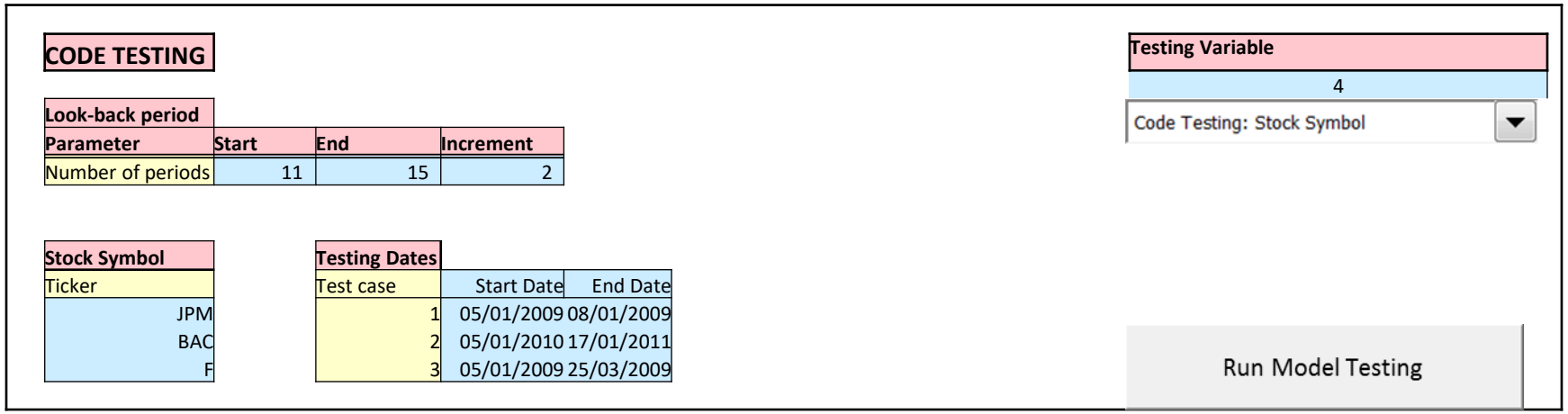


VII – Lesson 7: “Testing” Worksheet
VII.1 – Overview

- The testing of MA calculation is done in the “Testing” sheet.
  - Testing is initiated in the “Testing” rather than “Input” sheet and likewise for input data.
  - Code testing: including “manual” calculation; results are compared against each other.
- Testing Parameters
  - Lookback Period: starting period length, ending period length and the increment (how many units between subsequent periods).
  - Analysis Dates: varying the analysis data set.
  - Stock Symbol: varying stock ticker.
- Testing Approach
  - Results workbook: For each parameter, the results will be displayed in one workbook; one worksheet per parameter.
  - Comparison/Difference workbook: For each parameter, the difference between each model implementation against the first in the sheet will be computed; one worksheet per parameter.
  - Summary “.csv” file: For each workbook, one file describes the results and, if there are any errors, it reports them including the column number where they occur.

VII – Lesson 7: “Testing” Worksheet
VII.2 – Screenshot

- The following snapshot is taken from the “Testing” worksheet. The code (pertaining to the chosen functions in the “Input” sheet) is tested under three different scenarii as defined in the drop-down box in column N and then the user has to click “Run Model Testing” on the same sheet. The required input data for each scenario has to be defined on this worksheet, as opposed to defining it on “Input”, as follows.
  - Lookback Period: for each user-defined value starting at the value in cell C6 and ending at the value in cell D6 while incrementing by the number in cell E6.
  - Stock Symbol: for each stock symbol (vertical cells starting from B11), a new run will be created.
  - Analysis Dates: each test case is defined by a Start and an End Date starting in cells E11:F11.



- Strategy terminology in the software:

$$< modelName > \_Return: return_i$$

$$lsPosition_{i-1}: < modelName > \_Position$$

$$Strategy\ Return_i: < modelName > \_Strategy$$

$$< modelName > \_StrategyCumRet: e^{(\sum_i < modelName > \_StrategyCumRet)}$$

THANK YOU

Traders Island