



# MOVING AVERAGES

Application I – Version 2.1

**Application I:**

**Platform**

## I – Lesson 1: Moving Average Platform Application - Overview

- The web-based platform performs SMA and EMA calculation, charts the price vs MA, builds Strategy I in detail, and implements strategy back-testing using Strategy I.
- It is fully automated and can be accessed through the website.
- It does not require any prerequisite software installation; however, all your work will be done online rather than downloaded to your own device like the spreadsheet.
- In this tutorial, we will explore the functionalities currently offered by the platform.
  - MA implementation: this section explains how the platform calculates SMA and EMA using Excel, Javascript (Js) and Python.
  - Charting: this section explains basic charting in the platform.
  - Step-by-Step Strategy Building: this section illustrates the platform application of building a strategy. The platform allows you to manipulate various parameters and analyze their impact on the trading strategy outcome.
  - Strategy Back-testing Implementation: this section illustrates the platform application of optimizing returns and P&L.
    - For more on the last two points, please refer to the “Strategy BackTesting” tutorial for more details.

## II – Lesson 2: Moving Average Platform Application - Map

- We can hypothetically divide the platform into three sections.
  - “Calculations” Section: This section allows you to create a new “calculation” or to access a saved one. A calculation is an instance, or an application of the model given a particular data set. To access saved calculations, click “All my calculations” and to create a new one, click “Calculation preset.”
  - Holidays and Holiday Dates: you have to set-up the holiday calendar for the relevant market. You can either do it by clicking “Holidays” on the LHS of the Dashboard or by entering each date separately under “Holiday Dates” in the “INPUT” TAB.
  - TABS: this is where all the action takes place. In the next few slides, we will discuss these tabs in detail.
    - ✓ “INPUT”: input data and implementation/testing buttons
    - ✓ “HISTORICAL DATA”: historical data set (Date, Open, High, Low, Close, Adj Close, Volume) as determined by historical data dates on the “INPUT” tab.
    - ✓ “MA”: model implementation, e.g., “MA” (Moving Average).
    - ✓ “STRATEGY”: step-by-step implementation of strategy, e.g., Strategy I in “MA”.
    - ✓ “CHART”: price vs model line chart.
    - ✓ “BACK TESTING”: strategy back-testing; return or P&L optimization outcome.

### III – Lesson 3: INPUT TAB

#### III.1 – Common Elements

##### III.1.1 - Generic Parameters I

- This tab is the equivalent of a control panel for the platform; all user input data is fed into the system through this tab, and all remaining tabs are generated through it.
- User input data – Generic Parameters I: These parameters are required for all functionality.

Parameter	Description
Title	Choose a name under which this case will be saved
rounding	Number of figures after the decimals for results
Current Date	Equivalent of “Today’s” date
Ticker	Stock Exchange ticker
Historical data Start date	Start date for the complete data set
Historical data End date	End date for the complete data set
Analysis Data Analysis start date	Start date of analysis period
Analysis Data Analysis end date	End date of analysis period

- 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.

### III – Lesson 3: INPUT TAB

#### III.1 – Common Elements

##### III.1.2 - Model-Specific Parameters

- Unlike the spreadsheet implementation, the platform implements these parameters such that they feed into other tools including “BUILD STRATEGY” and “RUN BACKTESTING.” As such, they need to be set-up at the beginning. The “Moving Average” model takes the following model-specific parameters as input:

Parameter	Description
Lookback n	Moving average lookback period
Price	Open/High/Low/Close/Volume/Adj Close

# III – Lesson 3: INPUT TAB

## III.1 – Common Elements

### III.1.3 - TAB Screenshot: Generic and Model-Specific Parameters

The screenshot displays the FinanceApp interface with the following components:

- Header:** FinanceApp logo, search bar with "Explore Materialize", and user profile with 99 notifications.
- Left Sidebar:** Navigation menu with "Dashboard", "Calculations" (dropdown), "All my calculations", "Calculation preset", and "Holidays".
- Top Tabs:** INPUT (active), HISTORICAL DATA, MA, STRATEGY, CHA..., TESTING, BACK TESTING.
- Main Content Area:**
  - Holiday Dates:** A list of dates with a plus icon to add and a trash icon to delete. Dates include 2019-01-02, 2019-02-28, 2019-03-01, 2019-03-02, 2019-03-04, and 2019-03-06.
  - Title:** Input field containing "Joe Test 1".
  - rounding:** Input field containing "10".
  - Current date:** Input field containing "2021-05-06".
  - Ticker:** Input field containing "AMD".
  - Historical data:**
    - Start date:** Input field containing "2017-07-03".
    - End date:** Input field containing "2020-07-01".
  - Analysis Data:**
    - Analysis start date:** Input field containing "2018-01-02".
    - Analysis end date:** Input field containing "2019-05-10".
  - MA-Specific:**
    - Lookback n:** Input field containing "18".
    - Price:** Dropdown menu with "Close" selected.

### III – Lesson 3: INPUT TAB

#### III.1 – Common Elements

##### III.1.4 - Generic Parameters IIA

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

Check Box Text	Description
SMA via Formula	Calculate SMA using formulae that are visible
EMA via Formula	Calculate EMA using formulae that are visible
SMA Js Function	Calculate SMA using Js code
EMA Js Function	Calculate EMA using Js code
SMA Python Function	Calculate SMA using Python code
EMA Python Function	Calculate EMA using Python code



### III – Lesson 3: INPUT TAB

#### III.1 – Common Elements

##### III.1.5 - Generic Parameters IIB

- 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.
- Choice of Functions - 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 Function (C order)	Calculate EMA using Python code + C compiler
EMA Python Function (F order)	Calculate EMA using Python code + Fortran compiler
EMA Python Function (Cython)	Calculate EMA using Python/Cython code

# III – Lesson 3: INPUT TAB

## III.1 – Common Elements

### III.1.6 - Choice of Functions

The screenshot shows the FinanceApp interface. The top navigation bar includes the FinanceApp logo, a search bar with the text "Explore Materialize", and user profile icons. The left sidebar contains navigation items: Dashboard, Calculations (with a dropdown arrow), All my calculations (selected), Calculation preset, and Holidays. The main content area displays a table titled "Choice of Functions" with two columns: "MA Calculation" and "MA Speed Improvement". The table lists various function implementations for SMA and EMA, each with a checkmark indicating its availability.

Date	MA Calculation	MA Speed Improvement
2019-03-09		
2019-03-25		
2019-03-26		
2019-03-28		
2019-03-29		
2019-03-31		
2019-04-01		
2019-04-02		
	✓ SMA via Formula	✓ SMA Python Function (C order)
	✓ EMA via Formula	✓ SMA Python Function (F order)
	✓ SMA Js Function	✓ SMA Python Function (Cython)
	✓ EMA Js Function	✓ EMA Python Function (C order)
	✓ SMA Python Function	✓ EMA Python Function (F order)
	✓ EMA Python Function	✓ EMA Python Function (Cython)

### III – Lesson 3: INPUT TAB

#### III.2 – Strategy Building Specific Parameters

##### III.2.1 - Strategy Input Parameters

- These parameters are specific to Strategy I implementation, and you need to set them before clicking <BUILD STRATEGY>. Note that the latter reads the value for the lookback period “n” from the MA-Specific Parameters, and the threshold date from “Strategy BackTesting Input Parameters.”

Parameter	Description
iStrategy	Strategy I implementation
Sample Choice	In-, out-of- and complete sample
Model Choice (modelName)	Type of model: MA, “MovingAverage”
Specific Model Choice (iMethod)	Specific model: SMA, EMA
Python Speed Improvement (s)	None, Cython, C or Fortran compiler
Optimization Switch (optSwitch)	Fixed at No
Output to Excel? (outputExcel)	Fixed at Yes
Use Model Sheet for Manual Calc? (useModelSheet)	Fixed at No: calculates MA values instead of those in MA tab
Optimize Returns or P&L? (iOpt)	Redundant (no optimisation)

### III – Lesson 3: INPUT TAB

#### III.2 – Strategy Building Specific Parameters

##### III.2.2 - TAB Screenshot: Strategy Input Parameters

### Strategy Input Parameters

iStrategy	Sample Choice	
1	All	
Model Choice (modelName)	Specific Model Choice (iMethod)	
MovingAverage	SMA	
Python Speed Improvement (s)	Optimization Switch (optSwitch)	Output to Excel? (outputExcel)
none	No	Yes
Use Model Sheet for Manual Calc?	Optimize Returns or P&L? (iOpt)	
No	PnL	

## III – Lesson 3: INPUT TAB

### III.3 – Strategy BackTesting Parameters

#### III.3.1 - Strategy BackTesting Input Parameters

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

Parameter	Description
n	Lookback period range + increment
Threshold Date	Date dividing the data sample
Optimization Switch (optSwitch)	Fixed at Yes
Output to Excel? (outputExcel)	Output intermediate results to Excel. Fixed at Yes
Optimize Returns or PnL? (iOpt)	Optimize using Returns or PnL

### III – Lesson 3: INPUT TAB

#### III.3 – Strategy BackTesting Parameters

##### III.3.2 - TAB Screenshot: Strategy BackTesting Input Parameters

Strategy BackTesting Input Parameters		Testing Range		
Parameter	Min	Max	increment	
n	<input type="text" value="7"/>	<input type="text" value="9"/>	<input type="text" value="1"/>	
Threshold Date	<input type="text" value="2018-01-18"/>			
Optimization Switch (optSwitch)	<input type="text" value="Yes"/>			▼
Output to Excel? (outputExcel)	<input type="text" value="Yes"/>			▼
Optimize Returns or PnL? (iOpt)	<input type="text" value="PnL"/>			▼

### III – Lesson 3: INPUT TAB

#### III.4 – More Common Elements

##### III.4.1 - Buttons

###### Generating Processes

- This tab is also the “door” to calling implementation and testing procedures. You can do so by clicking the corresponding buttons on the tab.
- As we have mentioned earlier, the platform is fully automated and so to launch each functionality, you need to click the corresponding button at the bottom of the page.

Button Name	Action
<LOAD PRICES>	Loads data from Yahoo!
<UPDATE “MA” TAB>	Creates MA TAB
<GENERATE CHARTS>	Creates Price and MA Chart
<RUN BACKTESTING>	Back-tests a Trading Strategy
<BUILD STRATEGY>	Step-by-step Strategy I implementation
<SAVE CALCULATION>	Saves this case

## IV – Lesson 4: HISTORICAL DATA TAB

### IV.1 - Overview

- Historical data is essential to running any functionality on the platform, which retrieves it automatically from Yahoo! Finance according to user specification. The data is output to the HISTORICAL DATA TAB, where it is visible to the user.
- This is generated via the “INPUT” TAB.
  - Change the Ticker (stock symbol), Historical data Start date and Historical data End date.
  - Click the <LOAD PRICES> button at the bottom of the tab.
- The data can be overwritten but make sure you follow the existing column order/names in the template which can be downloaded by clicking “Download template” at the top of the “HISTORICAL DATA” TAB..



# IV – Lesson 4: HISTORICAL DATA TAB

## IV.2 - Screenshot

The screenshot shows the FinanceApp interface with the 'HISTORICAL D...' tab selected. The sidebar on the left contains navigation items: Dashboard, Calculations (with a dropdown arrow), All my calculations, Calculation preset (highlighted), and Holidays. The main content area has a top navigation bar with tabs: INPUT, HISTORICAL D..., MA, STRATEGY, CH..., TESTING, and BACK TESTING. Below the tabs, there are buttons for 'Download template', 'EXPORT XLSX', 'IMPORT XLSX OR CSV FILE', and 'CLEAR DATA'. The main data table has the following structure:

Date	Open	High	Low	Close	Adj Close	Volume
2021-01-04	92.1100006104	96.0599975586	90.9199981689	92.3000030518	92.3000030518	51802600
2021-01-05	92.0999984741	93.2099990845	91.4100036621	92.7699966431	92.7699966431	34208000
2021-01-06	91.6200027466	92.2799987793	89.4599990845	90.3300018311	90.3300018311	51911700
2021-01-07	91.3300018311	95.5100021362	91.1999969483	95.1600036621	95.1600036621	42897200

## V – Lesson 5: MA TAB

### V.1 - Overview

- The MA TAB displays the results for the MA calculations done by the code given the input parameters you enter in the “INPUT” tab.
- These calculations are generated through the “INPUT” tab as follows:
  - Update generic input data.
  - Update MA-specific data.
  - Select the function(s) to be used in computing MA (Choice of Functions – MA Calculation).
  - Select the function(s) with speed improvement in computing MA (Choice of Functions – MA Speed Improvement).
  - Click <Update “MA” TAB>.
  - The “MA” worksheet will be automatically generated including formatting.
  - 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.
  - You can download the results by clicking <EXCEL> or <CSV> , or by choosing <COPY> (and paste in Excel ©), at the top left hand side corner.

# V – Lesson 5: MA TAB

## V.2 - Screenshot

FinanceApp
Explore Materialize
🔍
🔖
🔔 99
👤

Dashboard
Calculations
All my calculations
Calculation preset
Holidays

INPUT			HISTORICAL DATA	MA	STRATEGY	CHARTS	TESTING	BACK TESTING	
# ^	Date	Price	SMA via Formula	EMA via Formula	SMA Js Function	EMA Js Function	SMA Python Function	EMA Python Function	SMA Pyth Function (order)
20	03/01/2019	17.0499992371	18.6238886515	18.9329842639	18.6238886515	18.9329842638	18.6238886515	18.9329842638	18.6238886515
21	04/01/2019	19.0000000000	18.5983331468	18.9400385519	18.5983331468	18.9400385518	18.5983331468	18.9400385518	18.5983331468
22	07/01/2019	20.5699996948	18.6305553648	19.1116134090	18.6305553648	19.1116134090	18.6305553648	19.1116134090	18.6305553648
23	08/01/2019	20.7500000000	18.6733331680	19.2840751554	18.6733331680	19.2840751554	18.6733331680	19.2840751554	18.6733331680
24	09/01/2019	20.1900005341	18.6572221120	19.3794357216	18.6572221120	19.3794357216	18.6572221120	19.3794357216	18.6572221120
25	10/01/2019	19.7399997711	18.6505553987	19.4173898321	18.6505553987	19.4173898321	18.6505553987	19.4173898321	18.6505553987
26	11/01/2019	20.2700004578	18.6711110009	19.5071383190	18.6711110009	19.5071383190	18.6711110009	19.5071383190	18.6711110009
27	14/01/2019	20.2299995422	18.7488887575	19.5832289741	18.7488887575	19.5832289741	18.7488887575	19.5832289741	18.7488887575
28	15/01/2019	20.3799991608	18.7977775998	19.6670995201	18.7977775998	19.6670995201	18.7977775998	19.6670995201	18.7977775998
29	16/01/2019	19.7299995422	18.8849998050	19.6737205751	18.8849998050	19.6737205751	18.8849998050	19.6737205751	18.8849998050
30	17/01/2019	20.2500000000	19.0133331087	19.7343815672	19.0133331087	19.7343815672	19.0133331087	19.7343815672	19.0133331087
31	18/01/2019	20.7700004578	19.2266664505	19.8433940820	19.2266664505	19.8433940820	19.2266664505	19.8433940820	19.2266664505
32	22/01/2019	19.7600002289	19.3994442622	19.8346157817	19.3994442622	19.8346157817	19.3994442622	19.8346157817	19.3994442622

## VI – Lesson 6: STRATEGY TAB

### VI.1 - Overview

- This tab illustrates in detail how Strategy I is built in the code, using all generic parameters (which have been discussed in previous slides), MA-Specific parameters, threshold date in the “Strategy Backtesting Input Parameters” section, and all parameters in the “Strategy Input Parameters” section on the “INPUT” tab so you need to set-up these parameters before you update the tab.
- The tab is fully automated, and it is generated through the “INPUT” tab by clicking the <Build Strategy> button.
- There are three sections in this tab:
  - Excel Strategy: shows the results of implementing this strategy using formulae.
  - JS Strategy: shows the results of implementing this strategy using Javascript code.
  - Python: shows the results of implementing this strategy using Python code.
- The three different methodologies implement the same strategy, albeit with different tools. So, the outcome of all three approaches should be exactly the same.
- In the next two slides, we will explain the outcome and show an example of the “STRATEGY” tab.

# VI – Lesson 6: STRATEGY TAB

## VI.2 - “Excel Strategy” Screenshot

INPUT
HISTORICAL D...
MA
STRATEGY
CH...
TESTING
BACK TESTING

EXPORT ALL

### Excel strategy

Copy
Excel
CSV

Date ^	Price	MA	MA Position	MA Return	MA Strategy	MA PnL
2019-01-03	17.0499992371	18.6238886515	-1.0000000000	-0.0993011796	0.0993011796	1.78000061
2019-01-04	19.0000000000	18.5983331468	1.0000000000	0.1082888202	-0.1082888202	-1.9500007
2019-01-07	20.5699996948	18.6305553648	1.0000000000	0.0793947097	0.0793947097	1.56999969
2019-01-08	20.7500000000	18.6733331680	1.0000000000	0.0087125578	0.0087125578	0.18000031
2019-01-09	20.1900005341	18.6572221120	1.0000000000	-0.0273587879	-0.0273587879	-0.5599994
2019-01-31	24.4099998474	20.6749999788	1.0000000000	0.0555932482	0.0555932482	1.31999969

Showing 21 to 40 of 109 entries

Previous
1
2
3
4
5
6
Next

---

<b>N</b>	18
<b>MA_StrategyCumSum</b>	-0.3652435561
<b>MA_StrategyCumRet</b>	0.6940275956
<b>MA_R</b>	-0.3059724044
<b>MA_CumPnL</b>	-8.1200027466

# VI – Lesson 6: STRATEGY TAB

## VI.3 - “Js Strategy” Screenshot

JS strategy

Copy Excel CSV

Date ^	Price	MA	MA Position	MA Return	MA Strategy	MA PnL
2018-12-03	23.70999990845	0	0		0	0
2018-12-04	21.1200008392	0	0	-0.1156763629	0	0
2018-12-06	21.2999992371	0	0	0.0084865383	0	0
2018-12-07	19.45999990845	0	0	-0.0903460072	0	0
2018-12-10	19.9899997711	0	0	0.0268711074	0	0
2018-12-11	19.9799995422	0	0	-0.0005003868	0	0
2019-01-02	18.8299999237	18.8599997626	-1.0000000000	0.0198451591	-0.0198451591	-0.3700008392

Showing 1 to 20 of 109 entries

Previous 1 2 3 4 5 6 Next

---

### Strategy

N	18
MA_StrategyCumSum	-0.3652435561
MA_StrategyCumRet	0.6940275956
MA_R	-0.3059724044
MA_CumPnL	-8.1200027466

# VI – Lesson 6: STRATEGY TAB

## VI.4 - “Python Strategy” Screenshot

Python strategy

Copy Excel CSV

Date ^	Price	MA	MA Position	MA Return	MA Strategy	MA PnL
2018-12-03	23.70999990845	0	0	0	0	0
2018-12-04	21.1200008392	0	0	-0.1156763629	0	0
2018-12-06	21.2999992371	0	0	0.0084865383	0	0
2018-12-07	19.4599990845	0	0	-0.0903460072	0	0
2018-12-10	19.9899997711	0	0	0.0268711074	0	0
2018-12-11	19.9799995422	0	0	-0.0005003868	0	0
2019-01-02	18.8299999237	18.8599997626	-1.0000000000	0.0198451591	-0.0198451591	-0.3700008392

Showing 1 to 20 of 109 entries

Previous 1 2 3 4 5 6 Next

---

Strategy

<b>N</b>	18
<b>MA_StrategyCumSum</b>	-0.3652435561
<b>MA_StrategyCumRet</b>	0.6940275956
<b>MA_R</b>	-0.3059724044
<b>MA_CumPnL</b>	-8.1200027467

## VI – Lesson 6: STRATEGY TAB

### VI.5 - Analysis

- Each implementation presents the results in two consecutive tables.
- The top tables illustrate step-by-step implementation at each observation date.
  - 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 depending on whether the price crosses MA from above or below.
  - MA Return: daily log returns.
  - MA Strategy: return of the strategy (generated by the position) on each observation date.
  - MA PnL: profit or loss on the position on each observation date.
- The bottom (smaller) tables display results summaries.
  - 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. “MA” corresponds to the model name.
  - MA\_R and MA\_CumPnL: show the final return and PnL of this strategy.
- You can download final results by clicking <DOWNLOAD ALL> on the top right-hand side of the page, or individual results by clicking <EXCEL> or <CSV>, or <COPY> (and then paste into Excel ©), on the top left-hand side of each implementation.



## VII – Lesson 7: BACKTESTING TAB

### VII.1 - Overview

- Strategy back-testing uses model functions, so it is dependent on the type of model you choose, and of course the strategy. At this moment in time, only Strategy I of the Moving Average model is implemented and back-tested in the platform.
- In a nutshell, the actual back-testing uses MA functions that are developed in Python. You can test the validity of these functions' results by running model testing through the spreadsheet. The platform acts as a user interface whereby Python code takes input from, and writes output to, the various tabs on the platform.
- The tab is fully automated, and it can be generated through the INPUT TAB.
  - Update Generic Parameters I & II. In the Model-Specific parameters, you only need to update the type of price to use during optimization. Finally, update strategy backtesting parameters.
  - In the background, Python code performs optimization procedures over the in-sample dataset and implements them in the out-of-sample as well as the complete sample data set. It returns the optimized parameter (n) and the corresponding returns/PnL for each sample given this optimized parameter.
  - The back-testing results are displayed in the BACKTESTING TAB.

# VII – Lesson 7: BACKTESTING TAB

## VII.2 - Screenshot 1

FinanceApp
Explore Materialize
🔍
🔖
🔔 99
👤

Dashboard
Calculations ▾
○ All my calculations
○ Calculation preset
Holidays

	INPUT	HISTORICAL DATA	MA	STRATEGY	CHARTS	TESTING	BACK TESTING	
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 Py Function (C order)
<b>iMethod</b>	SMA ▾	EMA ▾	EMA ▾	EMA ▾	EMA ▾	SMA ▾	SMA ▾	SMA ▾
<b>iStrategy</b>	Strategy I ▾	Strategy I ▾	Strategy I ▾	Strategy I ▾	Strategy I ▾	Strategy I ▾	Strategy I ▾	Strategy I ▾
<b>n</b>	9	9	9	9	9	9	9	9
<b>In-Sample Optimized pnl</b>	10.0999965668	14.1000003815	14.1000003815	14.1000003815	14.1000003815	10.0999965668	10.0999965668	10.0999965668
<b>Whole Sample Returns based on</b>	-4.7800083156	1.9999980932	1.9999980932	1.9999980932	1.9999980932	-4.7800083156	-4.7800083156	-4.7800083156

# VII – Lesson 7: BACKTESTING TAB

## VII.3 - Screenshot 2

FinanceApp

🔍 Explore Materialize
🗨️ 99
👤

- 🏠 Dashboard
- 📊 Calculations ▼
- All my calculations
- Calculation preset
- 🏠 Holidays

**Out-Of-Sample Returns based on Optimized pnl**

	-12.6000041962	-9.4600048068	-9.4600048068	-9.4600048068	-9.4600048068	-12.6000041962	-12.6000041962	-12.6000041962
MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...	MAOutput StrategyBackTes...
Example All	Example All	Example All	Example All	Example All	Example All	Example All	Example All	Example All
MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In	MAOutput StrategyBackTest In
Example In	Example In	Example In	Example In	Example In	Example In	Example In	Example In	Example In
MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out	MAOutput StrategyBackTest Out
Example Out	Example Out	Example Out	Example Out	Example Out	Example Out	Example Out	Example Out	Example Out

**Files**

[DOWNLOAD](#)

## VII – Lesson 7: BACKTESTING TAB

### VII.4 - Analysis

- The previous slides show 2 main sections in the tab.
  1. Input Parameters and output:
    - Input parameters: The first three rows show values for iMethod, iStrategy and n as defined on the “INPUT” tab. You should not change the parameters on this tab; they only display information. Any changes must be done on the “INPUT” tab.
    - Output: Optimized Returns and Parameters: this is the output returned by the Python code. The data set is divided into in-sample, where parameters are optimized to maximize returns/PnLs, out-of-sample and the complete data set to test the robustness of the optimized parameters.
      - ✓ Optimized Parameter: lookback period across all moving average methods.
      - ✓ Optimized Returns (or PnL): for all datasets across all moving average methods.
  2. Excel files, showing the intermediate step-by-step calculation, that can be downloaded. You can download each individual file or just click the <DOWNLOAD> button on the bottom left-hand side of the page to download final results.

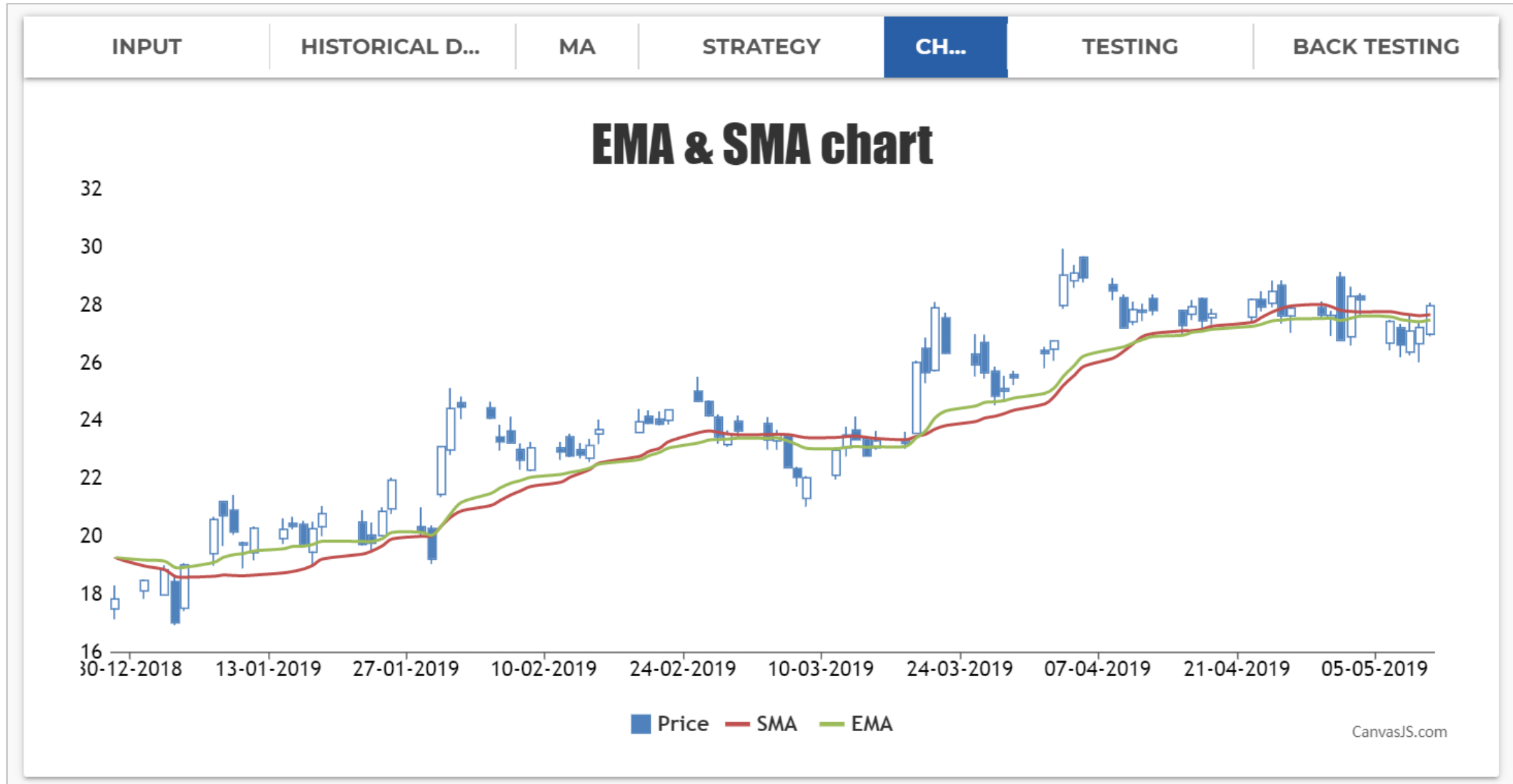
## VIII – Lesson 8: CHARTS TAB

### VIII.1 - Overview

- This tab displays a chart for the price vs moving average (SMA, EMA).
- It is fully automated and initiated through the “INPUT” tab by clicking the <GENERATE CHARTS> button.
- It relies on the “HISTORICAL DATA” tab so that you have properly set-up this tab before generating the chart.
- The following slide displays a chart example.

# VIII – Lesson 8: CHARTS TAB

## VIII.2 - Screenshot



- Strategy terminology in the software:

$\langle modelName \rangle \_Return: return_i$

$lsPosition_{i-1}: \langle modelName \rangle \_Position$

$Strategy Return_i: \langle modelName \rangle \_Strategy$

$\langle modelName \rangle \_StrategyCumRet: e^{(\sum_i \langle modelName \rangle \_StrategyCumRet)}$

# THANK YOU

---

## Traders Island

©TRADERSISLAND 2014-2022. ALL RIGHTS RESERVED.