



Manual de usuario

TP Final Certificación Profesional Python

Alumno: Mariano Agustín Juiz

DNI: 32.152.083

Manual de usuario

Inicio del programa

Una vez iniciado, el programa solicitará al usuario el ingreso de un ticker:

```
INICIO EJECUCION DEL PROGRAMA  
Ticker ID: AMZN
```

Luego del ingreso del ticker “AMZN” correspondiente a Amazon, debe elegirse un espacio o fracción de tiempo indicando el mismo con un número del 1 al 7.

EJ:

Introducimos el número 2 en la consola:

```
Debe definir un espacio de tiempo:  
1) minuto  
2) hora  
3) día  
4) semanal  
5) mensual  
6) trimestral  
7) anual  
  
Defina un intervalo de tiempo: 2
```

Output:

```
Puede visualizar barras de entre 1 y 23 horas: 1  
Seleccione un rango de fechas formato YYYY-MM-DD para AMZN  
  
Desde: 2022-09-05  
  
Hasta: 2022-09-15
```

En dicho ejemplo, elegimos un fraccionar el tiempo en horas al seleccionar la opción nro 2) “hora” permitiéndonos seleccionar entre 1 y 23 horas, dado que si quisiéramos 24hs, pasaríamos directamente a la opción nro 3) Día.

Esto fue configurado de tal forma, considerando que si utilizamos gráficos de vela, cada una de estas últimas equivale a la fracción de tiempo elegida. Por lo tanto si eligiéramos 24hs, estaríamos automáticamente pasándonos a un rango diario.

Luego de haber seleccionado el rango de tiempo, debe ingresarse el intervalo entre fechas, para las cuales se determinó el formato YYYY-MM-DD.

Al finalizar el ingreso de inputs iniciales, el programa descarga los datos solicitados desde la API <https://polygon.io/> y los ingresa –previamente normalizados y convertidos en DF- a una database de SQLITE, notificándonos que los datos fueron correctamente ingresados y

mostrando un breve resumen para orientarnos en cuanto a los datos que debemos luego manipular.

```
Los datos fueron ingresados correctamente en la DB
  index      v      vw      o      ...      h      l      t      n
0      0  10997.0  128.4580  128.21  ...  128.7000  128.100  1662451200000  233
1      1  16554.0  128.5535  128.63  ...  128.7000  128.400  1662454800000  197
2      2  18227.0  128.7891  128.69  ...  128.9800  128.610  1662458400000  261
3      3   80204.0  128.6668  128.64  ...  128.8800  128.250  1662462000000  1180
4      4  123036.0  128.5464  128.75  ...  128.8927  127.192  1662465600000  2581

[5 rows x 9 columns]
```

Dado que los datos del mercado financieros, no solo son dinámicos, sino que aparte existen millones de variaciones que un usuario podría elegir para visualizar, se optó por establecer `if_exists="replace"` e ir borrando y descargando los nuevos datos que el usuario desee.

También es posible visualizar sin haber pasado por la descarga en SQLite, pero no tendríamos de los datos guardados.

Posteriormente se cierra la conexión y el programa toma el DF desde SQL para rearmarlo:

```
La estructura del dataframe fue rearmada exitosamente:
      Date      Open      High      Low      Close      Volume
0  2022-09-06 08:00:00  128.21  128.7000  128.100  128.69  10997.0
1  2022-09-06 09:00:00  128.63  128.7000  128.400  128.70  16554.0
2  2022-09-06 10:00:00  128.69  128.9800  128.610  128.68  18227.0
3  2022-09-06 11:00:00  128.64  128.8800  128.250  128.60   80204.0
4  2022-09-06 12:00:00  128.75  128.8927  127.192  128.50  123036.0
```

Una vez realizado este paso, se preguntará al usuario si desea graficar los datos y elegir entre cinco estilos de gráficos distintos:

```
Desea visualizar la evolución del precio?
1) SI
2) NO
--> 1
Preparando datos para visualización...
Debe definir un estilo de gráfico:
1) Classic
2) Charles
3) Mike
4) Starsandstripes
5) Yahoo

Defina un estilo de gráfico
-->2
```

Otras opciones que brinda el programa son las de visualizar el volumen, un dato relevante para cualquier analista:

```
Desea visualizar el volumen?  
1) SI  
2) NO  
:--> 1
```

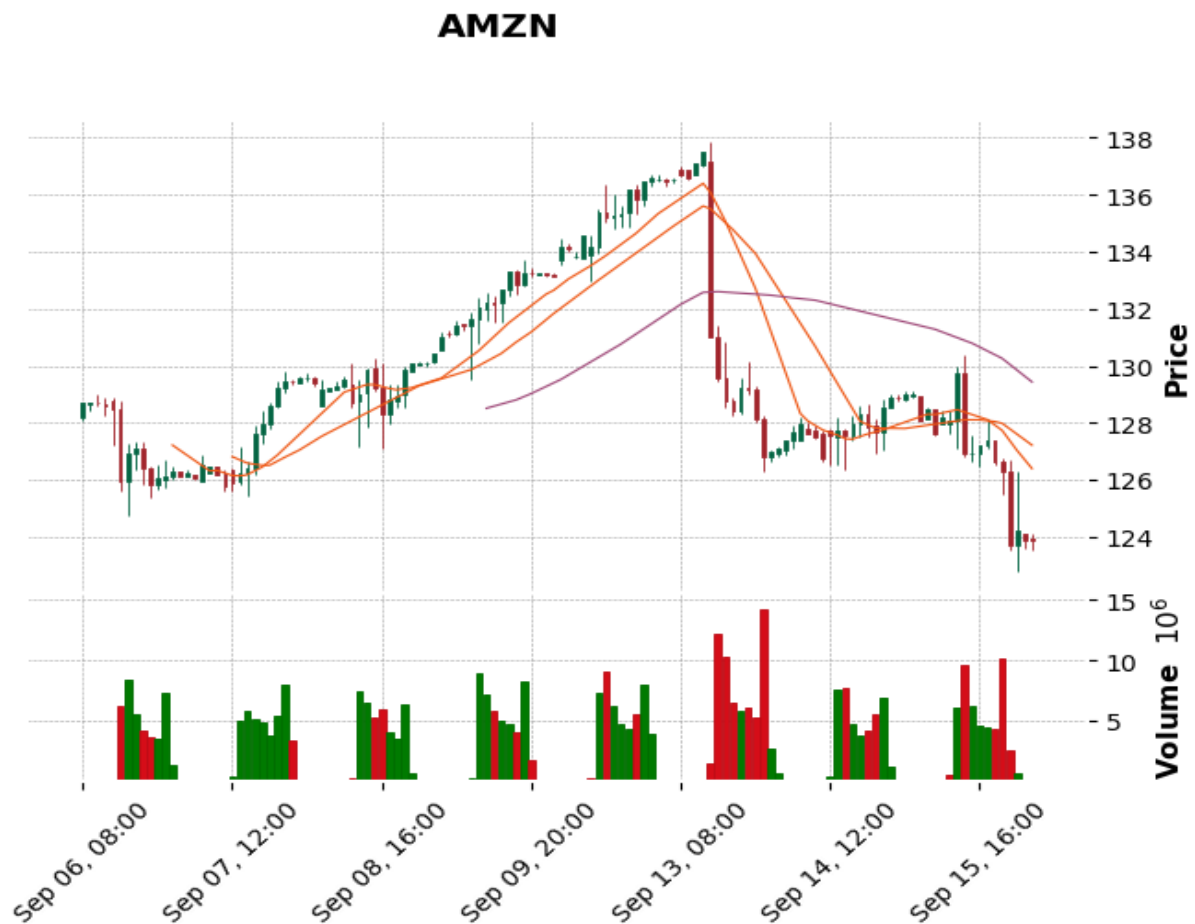
Y agregar medias móviles como indicador técnico.

Para cada media móvil debemos ingresar números que van del 1 al 200.

```
Desea agregar medias móviles?  
1) SI  
2) NO  
:--> 1  
Seleccione un periodo entre 1 y 200 para cada media móvil.  
  
Media móvil 1: 13  
  
Media móvil 2: 21  
  
Media móvil 3: 55
```

En el ejemplo se tomaron números de la secuencia de Fibonacci, dado que se utilizan habitualmente para hacer trading.

Finalizada la carga de números, el programa nos devuelve el siguiente gráfico:



y un resumen de los datos graficados.

```
Resumen de datos:
      Date      Open      High      Low      Close      Volume
2022-09-06 08:00:00 128.21 128.7000 128.100 128.69 10997.0
2022-09-06 09:00:00 128.63 128.7000 128.400 128.70 16554.0
2022-09-06 10:00:00 128.69 128.9800 128.610 128.68 18227.0
2022-09-06 11:00:00 128.64 128.8800 128.250 128.60 80204.0
2022-09-06 12:00:00 128.75 128.8927 127.192 128.50 123036.0
FIN DE EJECUCION DEL PROGRAMA

In [61]:
```

Nota: para que el programa funcione correctamente, fueron utilizadas las librerías detalladas.

```
import requests
from datetime import datetime as dt_dt
import pandas as pd
import sqlite3
import mplfinance as mpf
```