
cli-trade Documentation

Release 0.4.6

Valmir França

jan. 06, 2020

Contents:

1	mteli	1
1.1	Pré-requisitos	1
1.2	Instalação	1
1.3	Uso	2
2	Instalação	5
2.1	Versao estavel	5
2.2	A partir do codigo fonte	5
3	Uso	7
4	Contribuindo	9
4.1	Formas de Contribuição	9
4.2	Começando!	10
4.3	Diretrizes de Pull Request	11
5	Creditos	13
5.1	Development Lead	13
5.2	Colaboradores	13
6	Histórico	15
6.1	0.1.0 (2019-04-08)	15
7	Indices and tables	17

Utilitário de linha de comando para leitura de gráficos do MetaTrader 5.

- Free software: MIT license
- Documentação: <https://mtcli.readthedocs.io>.

1.1 Pré-requisitos

- [MetaTrader 5](#) - plataforma de trading.
- [GeraCSV.ex5](#) - robô executado no MetaTrader 5.

1.2 Instalação

```
pip install mtcli
```

1.2.1 Procedimento no MetaTrader 5

Faça o download do [GeraCSV.ex5](#).

1. Execute o MetaTrader 5 e abra um gráfico.
2. Execute o GeraCSV.ex5.
3. Selecione a opção “anexar ao gráfico” no menu de contexto do GeraCSV.ex5.

1.2.2 Arquivo .env

Crie um arquivo .env na pasta raiz do Windows com o conteúdo abaixo:

DIGITS="2"

CSV_PATH=[caminho_dos_arquivos_do_metatrader5]

1.3 Uso

Para exibir as últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -c 20
```

Para exibir o canal das últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -v ch -c 20
```

Para exibir o preço de fechamento das últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -v c -c 20
```

Para exibir o preço máximo das últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -v h -c 20
```

Para exibir o preço mínimo das últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -v l -c 20
```

Para exibir o range das últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -v r -c 20
```

Para exibir o volume das últimas 20 barras do diário do winq19:

```
mt bars winq19 -p daily -v vol -c 20
```

Para exibir o ATR(14) do diário do winq19:

```
mt atr winq19 -p daily
```

Para exibir o ATR(20) do diário do winq19:

```
mt atr winq19 -p daily -c 20
```

Para exibir a média móvel aritmética de 20 períodos do diário do winq19:

```
mt sma winq19 -p daily -c 20
```

Para exibir a média móvel exponencial de 20 períodos do diário do winq19:

```
mt ema winq19 -p daily -c 20
```

Para exibir as retrações e extensões de Fibonacci entre 103900 e 102100 na tendência de alta:

```
mt fib 103900 102100 h
```

Para exibir as retrações e extensões de Fibonacci entre 103900 e 102100 na tendência de baixa:

```
mt fib 103900 102100 l
```


2.1 Versao estavel

Para instalar mtcli, execute o seguinte comando em seu terminal:

```
pip install mtcli
```

Este e o metodo preferido para instalar o mtcli, pois sempre instalara a versao estavel mais recente.

Se você nao possui o [pip](#) instalado, este [guia de instalacao do Python](#) pode orientar durante o processo.

2.2 A partir do codigo fonte

O codigo fonte do mtcli pode ser baixado do [repo Github](#).

Você pode clonar o repositorio publico:

```
> git clone git://github.com/vfranca/mtcli
```

Ou fazer o download do [tarball](#):

```
> curl -OL https://github.com/vfranca/mtcli/tarball/master
```

Depois de ter uma copia do codigo fonte, você pode instalá-lo com:

```
poetry install
```


CAPÍTULO 3

Uso

Para usar o mtcli em um projeto:

```
import mtcli
```


Contribuições são bem-vindas e são muito apreciadas! Cada trecho ajuda e sempre será dado crédito.

Você pode contribuir de várias maneiras:

4.1 Formas de Contribuição

4.1.1 Relatar Bugs

Relate bugs em <https://github.com/vfranca/mtcli/issues>.

Quando você estiver relatando um bug inclua:

- O nome e a versão do seu sistema operacional.
- Quaisquer detalhes sobre a configuração local que possam ser úteis na solução de problemas.
- Passos detalhados para reproduzir o bug.

4.1.2 Correção de Bugs

Procure nos issues do GitHub por bugs. Qualquer coisa marcada com “bug” e “help wanted” estará aberto a quem quiser implementá-lo.

4.1.3 Implementar Features

Procure nos issues do GitHub para obter features. Qualquer coisa marcada com “enhancement” e “help wanted” está aberta para quem quiser implementá-la.

4.1.4 Escrever Documentação

O *mtcli* sempre poderá usar mais documentação, seja como parte dos documentos oficiais de *mtcli*, em docstrings ou até na web em postagens de blog, artigos e afins.

4.1.5 Enviar Feedback

A melhor maneira de enviar feedback é registrar um issue em <https://github.com/vfranca/mtcli/issues>.

Se você estiver propondo uma feature:

- Explique em detalhes como funcionaria.
- Mantenha o escopo o mais estreito possível afim de facilitar a implementação.
- Lembre-se de que este é um projeto conduzido por voluntários e que as contribuições são bem-vindas.

4.2 Começando!

Pronto para contribuir? Veja como configurar o *mtcli* para desenvolvimento local.

1. Bifurque o repositório *mtcli* no GitHub.
2. Clone seu fork localmente:

```
git clone git@github.com:seu_usuario/mtcli.git
```

3. Instale sua cópia local em um virtualenv. Supondo que você tenha o *virtualenvwrapper* instalado, é assim que você configura seu fork para desenvolvimento local:

```
mkvirtualenv mtcli
cd mtcli /
poetry install
```

4. Crie um branch para desenvolvimento local:

```
git checkout -b nome-do-seu-bugfix-ou-feature
```

Agora você pode fazer suas alterações localmente.

5. Quando terminar de fazer as alterações, verifique se as alterações passam no *flake8* e nos testes:

```
flake8 mtcli tests
pytest
```

6. Comite suas alterações e envie seu branch para o GitHub:

```
git add .
git commit -m "Descrição detalhada de suas alterações."
git push origin nome-do-seu-bugfix-ou-feature
```

7. Envie um pull request pelo site do GitHub.

4.3 Diretrizes de Pull Request

Antes de enviar um pull request, verifique se ele atende a estas diretrizes:

1. O pull request deve incluir testes.
2. Se o pull request adicionar uma funcionalidade, os documentos deverão ser atualizados. Colocar sua nova funcionalidade em uma função com uma string e adicione-o para a lista em README.rst.

5.1 Development Lead

- Valmir França <vfranca3@gmail.com>

5.2 Colaboradores

- Maicon Dinkel Baggio <maiconbaggio@outlook.com>

6.1 0.1.0 (2019-04-08)

- Primeiro release no PyPI.

Indices and tables

- `genindex`
- `modindex`
- `search`