

تابع `:current_time`:

این تابع با استفاده از `copy_rates_from_pos` یک دیتافریم میگیرد و زمان آخرین سطر را برمیگرداند.

تابع `:get_price_at_minmax_time`:

این تابع بر اساس عملیات مشخص (خرید یا فروش)، سرور و اطلاعات ورودی، قیمت پرس و پذیرش جفت ارز مورد نظر را برمیگرداند.

`:param initialize`: اطلاعات اتصال به پلتفرم `MetaTrader 5`، شامل لاگین، پسورد و سرور.
`type initialize`: tuple
`:param symbol`: نماد جفت ارز مورد نظر.
`type symbol`: str
`:param sleep_time`: زمان تاخیر قبل از اجرای عملیات.
`type sleep_time`: int
`:param action`: نوع عملیات ("buy" یا "sell").
`type action`: str
`:return`: قیمت پرس (ask) در صورت "buy" و قیمت پذیرش (bid) در صورت "sell".
`rtype`: float

تابع `:get_price`:

این تابع پس از اتصال به پلتفرم `MetaTrader 5`، قیمت خرید (ask) و قیمت فروش (bid) جفت ارز مورد نظر را برمیگرداند. برای این منظور از داده‌های 5 دقیقه‌ای (بر اساس گرافیک شمعی) آخرین میانگین قیمت را استخراج می‌کند.

`:param initialize`: اطلاعات اتصال به پلتفرم `MetaTrader 5`، شامل لاگین، پسورد و سرور.
`type initialize`: tuple
`:param symbol`: نماد جفت ارز مورد نظر.
`type symbol`: str
`:return`: دیکشنری حاوی قیمت خرید (ask) و فروش (bid).
`rtype`: dict

تابع `:get_candle`:

این تابع، آخرین کندل موجود بر اساس نماد، فریم زمانی و اطلاعات اتصال به پلتفرم `MetaTrader 5` را برمیگرداند.

`:param initialize`: اطلاعات اتصال به پلتفرم `MetaTrader 5`، شامل لاگین، پسورد و سرور.
`type initialize`: List
`:param symbol`: نماد جفت ارز مورد نظر.
`type symbol`: str
`:param timeframe`: فریم زمانی جهت گرفتن داده‌ها (مثلاً '5m' برای 5 دقیقه).
`type timeframe`: str
`:return`: DataFrame حاوی اطلاعات آخرین شمع (شمع آخرین ردیف داده‌ها).
`rtype`: pd.DataFrame

تابع get_ask:

این تابع پس از اتصال به پلتفرم MetaTrader 5، قیمت پرس (ask) جفت ارز مورد نظر را برمی‌گرداند. قیمت پرس نشان‌دهنده قیمتی است که فروشنده برای خرید آماده است.

param initialize: اطلاعات اتصال به پلتفرم MetaTrader 5، شامل لاگین، پسورد و سرور.
type initialize: List:
param symbol: نماد جفت ارز مورد نظر.
type symbol: str:
return: قیمت پرس (ask) جفت ارز.
rtype: float:

تابع get_bid:

این تابع پس از اتصال به پلتفرم MetaTrader 5، قیمت پذیرش (bid) جفت ارز مورد نظر را برمی‌گرداند. قیمت پذیرش نشان‌دهنده قیمتی است که خریدار برای فروش آماده است.

param initialize: اطلاعات اتصال به پلتفرم MetaTrader 5، شامل لاگین، پسورد و سرور.
type initialize: List:
param symbol: نماد جفت ارز مورد نظر.
type symbol: str:
return: قیمت پذیرش (bid) جفت ارز.
rtype: float:

تابع get_open_positions:

این تابع پس از اتصال به پلتفرم MetaTrader 5، تمام معاملات باز (Open Positions) را برمی‌گرداند.

param initialize: اطلاعات اتصال به پلتفرم MetaTrader 5، شامل لاگین، پسورد و سرور.
type initialize: List:
return: DataFrame حاوی اطلاعات تمام مواقع باز (Open Positions).
rtype: pd.DataFrame:

تابع get_data_from_mt5:

دریافت داده‌های بازار تاریخی از MetaTrader5.

param initialize: لیست حاوی اطلاعات لاگین و سرور برای حساب MetaTrader5.
لیست باید به فرمت [login, password, server] باشد.
type initialize: List:
param Ticker: رشته‌ای که نمایانگر نماد ارز مورد نظر برای دانلود است.
type Ticker: str:

param TimeFrame: رشته‌ای که نمایانگر فریم زمانی داده برای دانلود است.
مقادیر معتبر شامل "1m"، "5m"، "15m"، "30m"، "1h"، "4h"، "1d"، "1w" هستند.
type TimeFrame: str
return: DataFrame حاوی داده‌های بازار تاریخیچه.
rtype: pd.DataFrame:

تابع clean_investing_data:

این تابع به پاکسازی داده‌های دریافتی از سایت میپردازد Investing.com.

param df: DataFrame حاوی داده‌های دریافتی از Investing.com.
type df: pd.DataFrame
return: DataFrame پاکسازی شده حاوی اطلاعات معتبر.
rtype: pd.DataFrame:

تابع get_country_index_from_investing:

این تابع از سایت Investing.com اطلاعات شاخص کشور موردنظر را دریافت می‌کند.
برای این منظور، ابتدا داده‌های مربوط به US Dollar Index خوانده و پاکسازی می‌شوند.
اگر کشور مورد نظر 'USD' باشد، داده‌های US Dollar Index برگشت داده می‌شوند.
در غیر این صورت، بر اساس نوع کشور (اگر در لیست ['CAD', 'JPY', 'SEK', 'CHF'] باشد یا خیر)، داده‌های مربوط به آن کشور از پوشه‌های مخصوص به هر کشور خوانده و پاکسازی می‌شوند.
سپس، ستون‌های مهم انتخاب شده و نسبت به US Dollar Index محاسبه می‌شود.
در نهایت، میانگین ارزش‌های High، Low و Close، و تفاوت میانگین با ردیف قبلی محاسبه می‌شود.

param country: رشته‌ای که نمایانگر کد کشور مورد نظر است (مانند 'USD', 'CAD', 'JPY' و غیره).
type country: str
return: DataFrame حاوی اطلاعات شاخص کشور مورد نظر.
rtype: pd.DataFrame:

تابع get_csv_files:

تابعی که یک لیست از فایل‌های CSV در دایرکتوری مربوط به کشور داده شده را باز می‌گرداند.

param country: نام کشور برای اینکه می‌خواهیم فایل‌های CSV را دریافت کنیم.
type country: str
return: لیستی از فایل‌های CSV در دایرکتوری مربوط به کشور داده شده بدون پسوند '.csv'.
rtype: list:

تابع clean_news:

تابعی که متن اخبار موجود در DataFrame را پاکسازی می‌کند.

param df: DataFrame: حاوی ستون 'News' که متن اخبار در آن قرار دارد.
type df: pd.DataFrame:
return: لیستی از متن‌های پاکسازی شده.
rtype: list:

تابع fix_dataframe:

تابعی که یک DataFrame را به‌روزرسانی و پاکسازی می‌کند.

param tf: DataFrame: اصلی حاوی اطلاعات اقتصادی.
type tf: pd.DataFrame:
return: DataFrame: به‌روزرسانی شده و پاکسازی شده.
rtype: pd.DataFrame:

تابع convert_to_gmt:

تابعی که زمان موردنظر را به زمان میان‌مدت گرینویچ تبدیل می‌کند.

param requested_time: زمان موردنظر برای تبدیل.
type requested_time: pd.Timestamp:
return: زمان تبدیل‌شده به فرمت GMT.
rtype: str:

تابع create_positions_file:

تابعی که یک فایل برای ثبت موقعیت‌های تجاری جدید ایجاد یا باز می‌کند.

param timezone: منطقه زمانی برای ایجاد تاریخ و زمان فایل.
type timezone: tzinfo:
return: یک Tuple شامل DataFrame حاوی موقعیت‌های تجاری و مسیر فایل ایجاد یا باز شده.
[rtype: Tuple[pd.DataFrame, str]

این تابع با دریافت منطقه زمانی به عنوان ورودی، یک فایل CSV برای ثبت موقعیت‌های تجاری جدید ایجاد یا در صورت وجود باز می‌کند. اطلاعات موقعیت‌های تجاری در یک DataFrame ذخیره شده و مسیر فایل به عنوان خروجی ارائه می‌شود.

param timezone: منطقه زمانی موردنظر برای تولید تاریخ و زمان فایل.
type timezone: tzinfo:
return: یک Tuple شامل DataFrame حاوی موقعیت‌های تجاری و مسیر فایل ایجاد یا باز شده.
[rtype: Tuple[pd.DataFrame, str]

تابع get_today_calendar:

تابعی که تقویم اقتصادی امروز را برای کشورهای مشخص شده و در منطقه زمانی موردنظر دریافت می‌کند.

`param countries`: لیستی از نام‌های کشورها برای جستجو در تقویم اقتصادی.
`type countries`: List
`param timezone`: منطقه زمانی برای دریافت تقویم اقتصادی.
`type timezone`: tzinfo
`return`: DataFrame حاوی تقویم اقتصادی امروز و مسیر فایل ایجاد یا باز شده.
`rtype`: pd.DataFrame:

تابع `merge_dataframes`:

تابعی که اطلاعات موجود در چندین فایل CSV را ادغام و یک DataFrame حاوی تمام داده‌ها ایجاد می‌کند.

`param path`: مسیر الگوی جستجو برای فایل‌های CSV.
`type path`: str
`return`: DataFrame حاوی ادغام شده‌ی تمام داده‌ها.
`rtype`: pd.DataFrame:

تابع `get_calendar_historical_data`:

تابعی که اطلاعات تقویم اقتصادی تاریخی برای کشورهای مشخص و در بازه زمانی مشخص را دریافت و ذخیره می‌کند.

`param from_year`: سال شروع بازه زمانی مورد نظر.
`type from_year`: int
`param to_year`: سال پایان بازه زمانی مورد نظر.
`type to_year`: int
`param to_date`: تاریخ پایان بازه زمانی مورد نظر.
`type to_date`: str
`param save_path`: مسیر ذخیره فایل‌های CSV.
`type save_path`: str
`param countries`: لیستی از نام‌های کشورها برای دریافت تقویم اقتصادی.
`[type countries`: List[str]