**لیست توابع و کلاس‌های مهم + توضیح مختصر**

**🏗️ مدیریت Session و ابزارها**

* **get\_session\_local()** → ساختن یک Session محلی برای SQLAlchemy (با یا بدون Flask context).
* **normalize\_symbol\_text(txt)** → پاک‌سازی نام نمادها از کاراکترهای مخفی و فاصله اضافی.
* **check\_memory\_usage\_mb()** → برگرداندن مصرف حافظه فعلی پروسه (MB).
* **safe\_sleep(seconds)** → sleep ایمن با امکان قطع سریع (Ctrl+C).
* **retry\_on\_exception(...)** → دکوریتور برای اجرای دوباره توابع در صورت بروز خطا (با backoff).
* **create\_requests\_session()** → ساخت session برای requests با retry و headerهای پیش‌فرض.

**📦 کلاس‌ها (Wrapperها)**

* **PytseClientWrapper** → یک لایه ایمن روی pytse\_client برای:
  + all\_symbols() → دریافت همه نمادها
  + get\_all\_symbols() → گرفتن لیست نمادها
  + ticker(symbol\_id) و get\_ticker(symbol\_name) → ساخت آبجکت تیکر
  + fetch\_historical() → دریافت داده‌های تاریخی
* **InternalTseWrapper** → fallback برای گرفتن داده مستقیم از سایت tsetmc در صورت عدم کارکرد pytse\_client.
  + search\_symbol\_by\_name(query)
  + fetch\_export\_txt(instrument\_id)
  + fetch\_instrument\_info\_api(instrument\_id)
  + fetch\_loader\_page(instrument\_id) و parse\_loader\_page(html)
  + get\_instrument\_details(instrument\_id)

**📊 توابع اصلی دیتا**

* **filter\_symbols(symbols, exclude\_list)** → فیلتر کردن نمادها بر اساس نوع بازار و پسوند نام.
* **fetch\_symbols\_from\_pytse\_client(db\_session, limit)** → گرفتن لیست نمادها از pytse-client و ذخیره در DB.
* **fetch\_realtime\_data\_for\_all\_symbols(db\_session)** → دریافت داده‌های لحظه‌ای برای همه نمادها (هنوز پیاده‌سازی نشده).
* **fetch\_and\_process\_historical\_data(db\_session, ...)** → گرفتن و پردازش داده‌های تاریخی (OHLCV + حقیقی/حقوقی) و ذخیره در DB.
* **update\_symbol\_fundamental\_data(db\_session, ...)** → بروزرسانی داده‌های بنیادی (EPS, P/E, MarketCap و …).
* **fetch\_realtime\_snapshot(db\_session, symbol\_name, symbol\_id)** → ذخیره یک snapshot لحظه‌ای (قیمت + حقیقی/حقوقی + عمق بازار) در DB.

**🚀 توابع اجرای کلان**

* **run\_full\_data\_update(db\_session, ...)** → اجرای آپدیت کامل برای همه نمادها (تاریخی + بنیادی + تکنیکال + لحظه‌ای + الگوهای شمعی).
* **run\_daily\_update(db\_session, ...)** → اجرای آپدیت روزانه به صورت batch (پردازش همه نمادها مرحله به مرحله).
* **initial\_populate\_all\_symbols\_and\_data(db\_session, limit)** → پر کردن اولیه دیتابیس با نمادها، داده‌های تاریخی، بنیادی و لحظه‌ای.

**📉 تحلیل تکنیکال**

* **run\_technical\_analysis(db\_session, limit, symbols\_list)** → اجرای اندیکاتورها و محاسبات تکنیکال روی داده‌های تاریخی.
* **run\_candlestick\_detection(db\_session, ...)** *(در انتهای فایل استفاده می‌شود)* → تشخیص الگوهای شمعی.

**🛠️ توابع کمکی**

* **get\_last\_dates(db\_session)** → گرفتن آخرین تاریخ داده برای هر نماد از جدول HistoricalData.
* **get\_symbol\_id(symbol\_identifier)** → تبدیل شناسه نماد (اسم یا tse\_index) به ID داخلی دیتابیس.