

Документација за користење на Синглетон шема

Што е Синглетон шема?

Синглетон (Singleton) е дизајн-шема кој гарантира дека одредена класа ќе има само една инстанца за време на извршувањето на апликацијата. Покрај тоа, овој патерн овозможува глобална достапност на истата инстанца на сите делови од апликацијата.

Главни карактеристики на Синглетон шемата:

- Гарантира единствена инстанца на класа.
- Обезбедува контролиран пристап до инстанцата преку глобална точка за пристап.
- Го намалува користењето на ресурси кога е потребна само една инстанца.

Зошто користиме Синглетон шема во кодовите?

Код: `DataFetcherSingleton``

Во овој код, Синглетон се користи за да се гарантира дека:

1. Оптимизација на ресурсите:

- Прелистувачките опции (`webdriver.ChromeOptions``) и основната URL адреса се иницијализираат само еднаш, без повторно создавање на повеќе инстанци за секој обид за преземање на податоци.
- Наместо создавање нов објект при секој повик на методите, се користи една иста инстанца за преземање симбол кодови, тековни податоци и историски податоци.

2. Синхронизација на пристап:

- Со користење на `'Lock``, обезбедуваме безбедна синхронизација при креирањето на инстанцата, што е особено важно во сценарија со повеќе нишки.

3. Ефикасност при паралелно извршување:

- Инстанцата на Синглетон се користи во комбинација со `'ThreadPoolExecutor``, што овозможува паралелно преземање на податоци, без да се создаваат повеќе непотребни објекти.

Код: `NewsSentimentAnalyzerSingleton`

Во овој код, Синглетон се користи за да:

1.Гарантира уникатна инстанца на `SentimentIntensityAnalyzer`:

- Библиотеката `nltk` за анализа на сентимент се иницијализира само еднаш и се користи глобално за сите анализи на вести.

2.Управување со веб-пристап:

- Основната URL адреса за вести е дефинирана на едно место, со што се елиминира потребата за дуплирање на податоците и грешки.

3.Конзистентност:

- Единствената инстанца овозможува сите анализи на вести да бидат вршени со истиот објект, осигурувајќи дека секој процес на анализа се базира на идентични параметри.

Код: `IndicatorAnalyzerSingleton`

Во овој код, Синглетон обезбедува:

1.Глобален пристап до индикаторите:

- Методите за пресметување на индикатори, како што се `WMA`, `HMA` и `KAMA`, се статички и достапни преку единствена инстанца, овозможувајќи повторно користење на истите алгоритми.

2.Ефикасност:

- Единствената инстанца на Синглетон ја елиминира потребата за создавање на повеќе објекти при работа со различни временски рамки (дневни, неделни, месечни податоци).

3.Подобро управување со ресурси:

- Поставките и методите за пресметка на индикатори се делат меѓу сите временски рамки, намалувајќи ја потребата за дуплирање на логиката.

Предности од користење на Синглетон во овие кодови:

1.Подобрена мемориска ефикасност:

- Намалување на создавање на повеќе инстанци и повторно иницијализирање на истите податоци.

2.Конзистентност:

- Истите методи и параметри се користат низ сите делови на апликацијата, овозможувајќи конзистентно однесување.

3.Полесно тестирање:

- Единствената инстанца може лесно да се контролира и тестира, без потреба за подесување на повеќе објекти.

4.Подобро управување со нишки:

- Сите кодови користат `Lock` за да се избегнат потенцијални проблеми со конкурентен пристап.

Заклучок:

Користењето на Синглетон шемата во овие кодови овозможува подобра оптимизација на ресурсите, контрола на пристапот до важните објекти и поедноставено тестирање. Овој дизајн-шема е идеален за апликации кои бараат глобален пристап до специфични функционалности или податоци, како што се извлекување податоци, анализа на вести и примена на финансиски индикатори.