ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

*Московский институт электроники и математики им. А.Н. Тихонова*

# Руководство разработчика по работе с приложением: «Анализ публичных компаний».

Автор:

Горовой Владимир БИВ 202

Москва 2021 г.

## Требования к характеристикам компьютера и операционной системе

Наличие на компьютере интерпретатора «Python» (вне зависимости от среды разработки)

Операционная система – Windows 7, 8, 10.

## Версии интерпретатора и используемых библиотек

Интерпретатор – Python 3.7+.

Используемые библиотеки:

|  |  |
| --- | --- |
| **Библиотека** | **Версия** |
| pandas | 1.1.3+ |
| numpy | 1.19.2+ |
| selenium | 3.141.0+ |
| mysql-connector-python | 8.0.25+ |
| sqlalchemy | 1.3.20+ |
| pysimplegui | 4.43.0+ |
| plotly | 4.14.3+ |
| kaleido | 0.2.1+ |

## Описание структуры БД

Все базы данных имеют поля:

Company (Компания) – строка

Ticker (Тикер) – строка

Sector (Сектор) – строка

Sub-Sector (Подсектор) – строка

Revenue {year} (Выручка {год}) – вещественное число

Net Income {year} (Чистая прибыль {год}) – вещественное число

EBITDA {year} (EBITDA {год}) – вещественное число

Cash {year} (Свободные средства {год}) – вещественное число

Debt {year} (Долг {год}) – вещественное число

Total Assets {year} (Все активы {год}) – вещественное число

Total Liabilities {year} (Все обязательства {год}) – вещественное число

Market Cap {year} (Капитализация {год}) – вещественное число

Price {year} (Долг {год}) – вещественное число

P/E {year} (Мультипликатор P/E {год}) – вещественное число

P/S {year} (Мультипликатор P/S {год}) – вещественное число

P/B {year} (Мультипликатор P/B {год}) – вещественное число

EV/EBITDA {year} (Мультипликатор EV/EBITDA {год}) – вещественное число

NetDebt/EBITDA {year} (Мультипликатор NetDebt/EBITDA {год}) – вещественное число

D/E {year} (Мультипликатор D/E {год}) – вещественное число

ROE {year} (Мультипликатор ROE {год}) – вещественное число

ROA {year} (Мультипликатор ROA {год}) – вещественное число

EPS {year} (Мультипликатор EPS {год}) – вещественное число

## Структура каталогов

|  |  |  |
| --- | --- | --- |
| **Первый уровень** | **Второй уровень** | **Объяснение** |
| Work |  | Основной каталог с подкаталогами |
|  | Data | Содержит базы данных, из которых составляется единая база данных |
|  | Graphics | Содержит копии графических отчетов (создается и удаляется во время работы приложения) |
|  | Library | Содержит библиотеку стандартных (универсальных) функций, разработанных бригадой, которые могут использоваться для создания других приложений |
|  | Notes | Содержит документацию, в нем размещается руководства пользователя и разработчика |
|  | Scripts | Содержит специализированный модуль и файл с определением параметров настройки приложения. |

## Архитектура приложения

|  |  |  |
| --- | --- | --- |
| **Модуль** | **Местонахождение** | **Функция** |
| main.py | Work | Основной код |
| app.py | Work/Scripts | Формирование интерфейса |
| setup.py | Work/Scripts | Конфигурация настроек приложения |
| config.txt | Work/Scripts | Файл настроек приложения |
| db\_creator.py | Work/Library | Работа с базой данных на локальной сервере MySQL |
| modifier.py | Work/Library | Преобразование всех таблиц в одну с очисткой от некорректных данных |
| creator\_splist\_for\_parser.py | Work/ Library | Создание списка компаний для отправки в парсеры |
| splist\_with\_mc\_parser.py | Work/ Library | Парсинг текущих цен и капитализаций |
| first\_year\_list\_parser.py | Work/ Library | Парсинг первых годов финансовых отчетов, представленных на сайте |
| fin\_data\_parser.py | Work/ Library | Парсинг финансовых отчетов компаний |

## Листинг основного скрипта и всех модулей.

|  |  |
| --- | --- |
| **Модуль** | **Функции с докстрингами** |
| main.py  """  Модуль запуска программы  Автор: Владимир Горовой  """ |  |
| setup.py  """  Модуль настроек приложения  Автор: Владимир Горовой  """ |  |
| app.py  """  Модуль интерфейса  Авторы:  Владимир Горовой  Алексей Маркин  """ | def activate\_app  """  Функция вызова из другого скрипта  Авторы:  Владимир Горовой  Алексей Маркин  Вход: fin\_data - pd.DataFrame - таблица всей базы данных  Выход: -  """  def get\_values\_by\_name  """  Получение параметров при нажатии первой кнопки  Автор: Владимир Горовой  Вход: name – название компании  Выход: values\_data – список списков значений из базы данных  """  def get\_values\_by\_ticker  """  Получение параметров при нажатии второй кнопки  Автор: Владимир Горовой  Вход: tick – тикер компании  Выход: values\_data – список списков значений из базы данных  """  def add\_company\_by\_name  """  Описание второго окна при нажатии первой кнопки  Автор: Владимир Горовой  Вход: name – название компании  Выход: c\_lay –расположение элементов второго окна  """  def add\_company\_by\_ticker  """  Описание второго окна при нажатии второй кнопки  Автор: Владимир Горовой  Вход: tick – тикер компании  Выход: c\_lay – расположение элементов второго окна  """  def visualize  """  Визуализация параметра по названию компании в формате html  Автор: Алексей Маркин  Вход: name – название компании, par – наименование параметра  Выход: файл в формате html с графиком параметра  """  def visualize3  """  Визуализация параметра по наименованию тикера в формате html  Автор: Алексей Маркин  Вход: name – наименование тикера, par – название параметра  Выход: файл в формате html с графиком параметра  """  def visualize2  """  Визуализация параметра по названию компании в формате png  Автор: Алексей Маркин  Владимир Горовой  Вход: name – название компании, par – наименование параметра  Выход: файл в формате png с графиком параметра  """  def visualize4  """  Визуализация параметра по наименованию тикера в формате png  Автор: Алексей Маркин  Владимир Горовой  Вход: name – наименование тикера, par – название параметра  Выход: файл в формате png с графиком параметра |
| db\_creator.py  """  Модуль обработки базы данных на локальном сервере MySQL  Автор: Владимир Горовой  """ | def create\_server\_connection  """  Подключение к локальному серверу MySQL  Автор: Владимир Горовой  Вход: host\_name – имя хоста, user\_name – имя пользователя, password – пароль  Выход: connection – переменная соединения с сервером  """  def create\_database  """  Создание базы данных  Автор: Владимир Горовой  Вход: connection – переменная соединения с сервером, db\_name – имя базы данных  Выход: cursor – переменная работы с базой данных (курсор)  """  def fill\_table\_in\_database  """  Заполнение таблицы базы данных  Автор: Владимир Горовой  Вход: host\_name – имя хоста, user\_name – имя пользователя, password – пароль, database\_name – имя базы данных, data – pd.DataFrame - вставляемая таблица  Выход: -  """  def get\_data\_from\_database  """  Выгрузка данных из БД локального сервера MySQL  Автор: Владимир Горовой  Вход: имя хоста, имя пользователя, пароль, имя базы данных  Выход: data – pd.DataFrame – полученная таблица  """ |
| modifier.py  """  Модуль объединения всех таблиц в одну  Автор: Алексей Маркин  """ | def remove\_unused\_rows\_from\_table  """  Удаление строк с искаженными данными  Автор: Алексей Маркин  Вход: some\_list – исходный список pd.DataFrame  Выход: some\_list – очищенная таблица pd.DataFrame  """  def checknan  """  Проверка на None  Автор: Алексей Маркин  Вход: num – значение  Выход: True/False в зависимости от результат проверки  """  def add\_multiples  """  Добавление нескольких названий столбцов одного мультипликатора  Автор: Алексей Маркин  Вход: stn – мультипликатор(строка), multiples – список всех названий столбцов  Выход: multiples – измененный список всех названий столбцов  """  def modify  """  Функция вызова из другого скрипта  Автор: Алексей Маркин  Вход: 6 спарсенных таблиц в виде csv-фалов: fin\_data.csv, first\_year\_list.csv, splist.csv, splist\_with\_mc.csv, price\_list.csv, splist\_with\_mc\_old.csv  Выход: единая таблица в виде csv-файла: data.csv  """ |
| creator\_splist\_for\_parser.py  """  Модуль формирования списка компаний для отправки в парсеры  Автор: Алексей Маркин  """ | def activate\_creator\_splist\_for\_parser  """  Функция для вызова из другого скрипта  Автор: Алексей Маркин  Вход: splist.csv - список компаний в виде csv-файла  Выход: преобразованный под парсеры список компаний в виде csv-файла: splist\_for\_parser.csv  """ |
| splist\_with\_mc\_parser.py  """  Модуль парсинга текущих цен и капитализаций  Автор: Алексей Маркин  """ | def push\_sorting\_mc  """  Нажатие кнопки сортировки по капитализации  Автор: Алексей Маркин  Вход: browser – драйвер браузера  Выход: -  """  def push\_load\_more  """  Нажатие кнопки загрузить ещё для открытия всего контента страницы  Автор: Алексей Маркин  Вход: browser – драйвер браузера  Выход: -  """  def activate\_splist\_with\_mc\_parser  """  Функция для вызова из другого скрипта  Автор: Алексей Маркин  Вход: -  Выход: таблица с капитализациями и ценами акций компаний в виде csv-файла: splist\_with\_mc\_parser.csv  """ |
| first\_year\_list\_parser.py  """  Модуль парсинга первых годов финансовых отчетов, представленных на сайте  Автор: Владимир Горовой  """ | def get\_year  """  Получение года первой финансовой отчетности, представленной на сайте  Автор: Владимир Горовой  Вход: page – ссылка на веб-страницу, browser – драйвер браузера  Выход: year – год  """  def activate\_first\_year\_list\_parser  """  Функция для вызова из другого скрипта  Автор: Владимир Горовой  Вход: преобразованный под парсеры список компаний в виде csv-файла: splist\_for\_parser.csv  Выход: таблица с первыми годами финансовых отчетов, представленных на сайте, в виде csv-файла: first\_year\_list.csv  """ |
| fin\_data\_parser.py  """  Модуль парсинга финансовых отчетов компаний  Автор: Владимир Горовой  """ | def right\_scroll  """  Прокрутка таблицы с данными вправо  Автор: Владимир Горовой  Вход: browser – драйвер браузера  Выход: -  """  def left\_scroll  """  Прокрутка таблицы с данными влево  Автор: Владимир Горовой  Вход: browser – драйвер браузера  Выход: -  """  def get\_table  """  Получение html-кода таблицы с данными  Автор: Владимир Горовой  Вход: link – ссылка на веб-страницу, browser – драйвер браузера  Выход: table – адрес таблицы на веб-странице  """  def input\_nothing  """  Заполнение списка пустым значением  Автор: Владимир Горовой  Вход: ind – список  Выход: ind – измененный список с вставленными пустыми значениями  """  def fill\_lists1  """  Занесение значений Revenue, Net Income, EBITDA в соответсвующий список  Автор: Владимир Горовой  Вход: revenue\_row, net\_income\_row, ebitda\_row – адреса строк значений Revenue, Net Income и EBITDA на веб странице;  j – индекс компании в списке  Выход: revenues, net\_incomes, ebitdas – списки значений Revenue, Net Income и EBITDA  """  def get\_income\_statement\_info  """  Получение значений Revenue, Net Income, EBITDA из таблицы  Автор: Владимир Горовой  Вход: table – адрес таблицы на веб-странице, start\_year – год первой финотчетности, browser – драйвер браузера  Выход: revenues, net\_incomes, ebitdas – списки значений Revenue, Net Income и EBITDA  """  def fill\_lists2  """  Занесение значений Cash, Debt, Total Assets, Total Liabilities в соответсвующий список  Автор: Владимир Горовой  Вход: cash\_row, debt\_row, t\_assets\_row, t\_liabs\_row – адреса строк значений Cash, Debt, Total Assets, Total Liabilities на веб странице;  j – индекс компании в списке  Выход: cashes, debts, total\_assets, total\_liabilities – списки значений Cash, Debt, Total Assets, Total Liabilities  """  def get\_balance\_sheet\_info  """  Получение значений Cash, Debt, Total Assets, Total Liabilities из таблицы  Автор: Владимир Горовой  Вход: table – адрес таблицы на веб-странице, start\_year – год первой финотчетности, browser – драйвер браузера  Выход: cashes, debts, total\_assets, total\_liabilities – списки значений Cash, Debt, Total Assets, Total Liabilities  """  def input\_fins\_in\_table  """  Заполнение общей таблицы с финансовыми показателями компаний  Автор: Владимир Горовой  Вход: revenues, net\_incomes, ebitdas, cashes, debts, total\_assets, total\_liabilities – списки значений Revenue, Net Income, EBITDA, Cash, Debt, Total Assets, Total Liabilities;  fin\_data – pd.DataFrame - таблица, в которую заносятся данные;  num – индекс компании в списке; splist – pd.DataFrame - список компаний  Выход: fin\_data – pd.DataFrame - таблица с занесенными данными  """  def activate\_fin\_data\_parser  """  Функция для вызова из другого скрипта  Автор: Владимир Горовой  Вход: список компаний в виде csv-файла: splist\_for\_parser.csv  Выход: таблица со всеми финансовыми показателями в виде csv-файла: fin\_data.csv  """ |