**Фреймворк для анализа биржевых котировок.**

**Описание реализованного функционала (метод Run() класса Main)**

1. Чтение данных из файла data\daily\_snapshots\input\test.txt и запись в память в виде таблицы класса DataSet согласно файла конфигурации settings\daily\_snapshots\tick\_ds.ini.
2. Упаковка данных в три DataSet’а согласно файла конфигурации settings\daily\_snapshots\zz\_pack\_ds.ini
3. Запись в файлы data\daily\_snapshots\output\zz\_base.txt, zz\_first.txt, zz\_second.txt согласно файла конфигурации settings\daily\_snapshots\zz\_pack\_ds.ini

**Описание реализованных компонентов**

1. daily\_snapshots.xlsm – файл пректа
2. components\daily\_snapshots\ProjectManager\v01\ProjectManager.bas – управление проектом
   1. Init() – подключение компонентов
   2. RemoveComponents() – удалить компоненты из проекта
   3. ImportComponents() – импортировать компоненты в проект
   4. ExportComponents() – экспортировать компоненты в текстовые файлы
   5. CreateFolder(ByVal folderPath As String) – создание папки для файла компонента
   6. ClearFolder(ByVal folderPath As String) – удалить файлы из папки
3. components\daily\_snapshots\Main\v01\Main.bas – главный класс проекта
   1. Init() – подключение файлов конфигурации проекта
   2. Run() – запуск проекта
4. components\common\CommonMethods\v01\CommonMethods.cls – хелпер, состоящий из методов, не выделившихся в отдельный класс
5. components\common\Errors\v01\Errors.cls – хранитель описания ошибки
   1. Init() – объявление переменных
   2. RiseError(msg As String) – установить ошибку
6. components\common\RWFile\v01\RWFile.cls – чтение/запись файла
   1. OpenFile(file\_Path As String, mode As String) – открыть файл
   2. ReadLine() – чтение строки из файла
   3. WriteLine(line As String) – запись строки в файл
   4. CloseFile() – закрыть файл
7. components\common\RWini\v01\RWini.cls – чтение/запись ini-файла
   1. ParseLine(ByVal line As String) – распознавание секции или параметра со значением
   2. ReadSettings(filePath As String, settings As Scripting.Dictionary) – чтение ini файла в ассоциативный массив
   3. ComposeSettings(settings As Scripting.Dictionary, settingsLnk As Scripting.Dictionary) – сборка ассоциативных массивов в один для составления единой конфигурации из нескольких ini-файлов и достижения уровня вложенности больше 2-ух, если это необходимо
8. components\common\SettingsReader\v01\SettingsReader.cls – чтение значений конфигурации
   1. GetValue(dict As Scripting.Dictionary, key As String) – выдает значение конфигурации по ключу. В случае отсутствия ключа, выдает значение с ключом “default” или пустую строку
9. components\common\DataSet\v01\DataSet.cls – класс для хранения и работы с двумерной таблицей данных
   1. Create(ByVal dsSettings As Scripting.Dictionary) – создает таблицу данных согласно настройкам dsSettings
   2. ReadFromFile(ByVal filePath As String, ByVal dsSettings As Scripting.Dictionary) – читает таблицу данных из csv-файла
   3. WriteToFile(ByVal filePath As String, ByVal dsSettings As Scripting.Dictionary) - записывает таблицу данных в csv-файл
   4. GetCell(ByVal colName As String, ByVal rowCount As Long) – получить значение ячейки таблицы
   5. SetCell(ByVal colName As String, ByVal rowCount As Long, val As Variant) – установить значение ячейки таблицы
10. components\common\DataSetIterator\v01\DataSetIterator.cls – хранение указателя на строку таблицы
    1. Init(Data\_Set As DataSet, col\_Names As String) – инициализация итератора для таблицы
    2. InitByIndex(Data\_Set As DataSet, col\_Names As String, idx() As Long) - инициализация итератора для таблицы по отдельному массиву индексов таблицы, например каким-либо образом отсортированному
    3. InitRange(Data\_Set As DataSet, col\_Names As String, start\_Index As Long, stop\_Index As Long) – инициализация итератора для части таблицы
    4. NextRow() – передвинуть указатель на следующую строку
    5. NextRowByIndex(idx() As Long) – передвинуть указатель на следующую строку, определяемую по отдельному массиву индексов таблицы
    6. NextRangeRow() – передвинуть указатель на следующую строку части таблицы
11. components\common\TypeConvertor\v01\TypeConvertor.cls – конвертация типа данных при чтении из файла и форматирование данных при записи в файл
    1. Cast(val As Variant, vType As String, dec As String) – конвертация типа данных
    2. FormatV(val As Variant, vFormat As String, dec As String) – форматирование данных
12. components\exchange\Buffer\v01\Buffer.cls – буфер FIFO
13. components\exchange\ZigZagAbsolute\v01\ZigZagAbsolute.cls
14. components\exchange\ExchangeMethods\v01\ExchangeMethods.cls