Тестовое задание для программиста C#

# Постановка задачи

Разработать консольное приложение, позволяющее выполнить следующие действия:

1. Загрузить в «память» компьютера (ОЗУ) последовательность данных из указанного файла
2. Проанализировать прочитанные данные и выбрать только те, которые удовлетворяют определенному условию
3. Прошедшие проверку данные разбить на заданное количество частей и сохранить каждую из частей в файл

# Подробности реализации

Исходные данные представлены в виде списка объектов, каждый из которых описывает параметры подключения к некоторой базе данных. Список размещается в текстовом файле в кодировке Utf8. Путь к файлу задается аргументом командной строки.

Каждый объект данных представлен в виде:

[<Заголовок>]

<Имя параметра>=<Значение параметра>

…

<Имя параметра>=<Значение параметра>

<пустая строка>

, где:

<Заголовок> - имя базы данных. Тип: строка. Может содержать любые символы, не может быть пустым.

<Имя параметра> - Тип: строка. Может содержать любые символы, не может быть пустым.

<Значение параметра> - Тип: стока или число. Может отсутствовать;

<Пустая строка> - пустая строка. Является разделителем объектов данных друг от друга.

Каждый объект данных может иметь от одного до нескольких параметров. Обязательным параметром является только параметр «Connect». Другие параметры, если они будут присутствовать, нужно игнорировать при чтении данных.

Параметр «Connect» может принимать следующие значения:

* File="<путь к папке>";

,где

<путь к папке> - путь к локальной или удаленной папке файловой системы с размещением базы данных;

* Srvr="<хост>:<port>";Ref="<имя базы>";

,где

<хост> - имя компьютера в сети, на котором размещен сервер баз данных

<порт> - номер сетевого порта для подключения к серверу базы данных

<имя базы> - имя базы данных на сервере.

При чтении данных из файла требуется проверять формат представления данных. В случае нарушения структуры файла с исходными данными (отсутствует заголовок данных или обязательный параметр; отсутствует имя параметра или разделитель имени от значения – символ «=») приложение должно известить об этом путем возврата кода ошибки (-1) и выводом соответствующего сообщения в поток вывода (console). Пустой исходный файл не является ошибкой.

После чтения данных требуется проверить их по следующим условиям:

* Для файловой базы данных проверить правильность указания пути к нет – путь не должен содержать некорректных символов (см. правила формирования пути к файлу ОС Windows)
* Для серверной базы данных требуется обязательное наличие имени сервера и названия базы данных на нём. Порт для подключения к серверу может отсутствовать.

Данные, которые не соответствуют заданному условию, должны быть сохранены в файл с именем «bad\_data.txt», который должен располагаться в той же папке, в которой размещено само приложение. Если файл уже существует, он должен быть перезаписан.

Данные, прошедшие проверку, разделить на 5 частей и сохранить каждую из частей в файл «base\_<номер части>.txt». Размещение файлов - папка с приложением. Если файлы с таким именем уже существуют, они должны быть перезаписаны. Формат сохранения данных должен быть таким же, как при чтении их из файла (см. формат исходных данных).

При выполнении задания учесть следующее:

* Формат, в котором читаются и записываются данные, может быть изменен
* Могут быть добавлены дополнительные проверки данных
* Количество частей, на которые разбиваются данные, может изменяться
* Результирующие данные могут быть сохранены в другое местоположение

Структура приложения должна разрабатываться с применением ООП (объектно-ориентированного программирования).

# Примерное содержание файла с исходными данными

[KTA\_NDS]

Connect=File="\\clusterfs126\users\91776\DB\Accounting3";

ID=e74bf314-e943-4830-bd93-befc04aa714e

External=0

[МХМ]

Connect=File="s:\usersdata\643343\МХТ";

[Оптовик]

Connect=Srvr="host371:1741";Ref="432345\_base01";

ID=e74bf314-e943-4830-bd93-36aba8c5a66d

External=1

[ООО &quot;Сити&quot;]

Connect=Srvr="host166";Ref="635445\_base23";