

«ОБРАБОТЧИК МЕТЕОРОЛОГИЧЕСКИХ ДАННЫХ»

Инструкция по использованию программы

Выполнил:

Шошин Д.Д.

Содержание

| | |
|--------------------------------------------|----|
| Описание программы..... | 3 |
| Порядок работы с программой | 4 |
| Предварительная настройка приложения | 11 |
| Логирование процесса работы программы..... | 14 |

Описание программы

Программа «Обработчик метеорологических данных» (рисунок 1) предназначена для создания записей о параметрах погоды с последующим отображением отчёта и сохранением данных в файл или базу данных.

Обработчик метеорологических данных

Добавление записи

Температура, °C: 0

Условия: Ясно

Заметка:

Дата: 07.10.2023

Время: 14:01:58

+ Добавить

| Дата | Время | Температура, °C | Условия | Заметка |
|------|-------|-----------------|---------|---------|
|------|-------|-----------------|---------|---------|

Сохранение в файл

Сохранить

Путь к файлу:

Записи ещё не сохранены в файл.

Открыть в проводнике

Отчёт

Сформировать отчёт

Отчёт ещё не сформирован.

Сохранение в базу данных

Сохранить

Путь к файлу:

Записи ещё не сохранены в базу данных.

Открыть в проводнике

Рисунок 1 – Окно программы «Обработчик метеорологических данных»

При разработке данных использовались следующие технологии:

C#, NET 6.0, DI, WPF, MVVM, SQLite, Serilog.

Порядок работы с программой

Работа с программой осуществляется в следующем порядке:

1. Запустить программу «Обработчик метеорологических данных» (файл WeatherDataSaver.exe);
2. Задать параметры погоды для новой записи:

2.1. Температура в градусах Цельсия – целое либо дробное число в диапазоне от -100 до 100;

Примечание: в случае ввода некорректных символов при валидации данных будет показана ошибка, а значение температуры примет последнее корректное значение (рисунок 2).

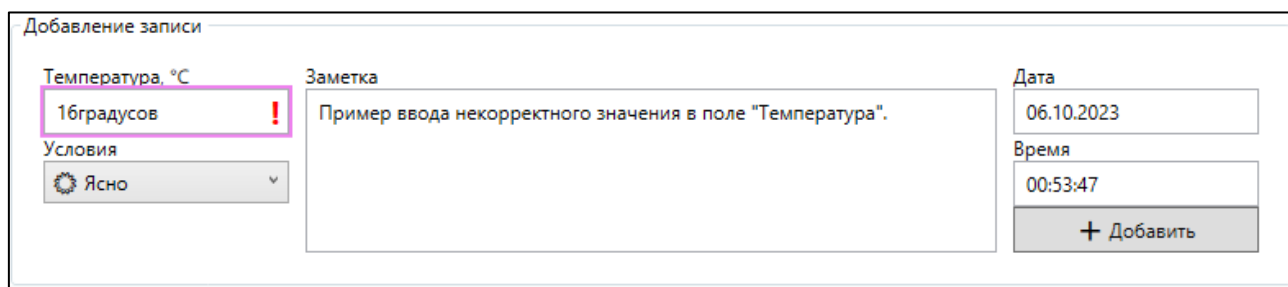


Рисунок 2 – Ошибка при валидации значения температуры

- 2.2. Условия – одно значение из вариантов: «Ясно», «Облачно», «Туман», «Дождь», «Снег» или «Град»;
 - 2.3. Заметка – необязательное поле, позволяющее записать любое дополнительное однострочное или многострочное текстовое сообщение;
 - 2.4. Дата – автоматически прописывается текущая дата;
 - 2.5. Время – автоматически прописывается текущее время;
 3. Нажать «Добавить» для внесения записи в таблицу.
 4. При необходимости занесения нескольких записей повторить шаги 2 и 3.
- Пример заполненной таблицы данных показан на рисунке 3.

| Дата | Время | Температура, °C | Условия | Заметка |
|------------|----------|-----------------|---------|-----------------------------------------|
| 06.10.2023 | 00:57:49 | 10 | Облачно | На улице очень пасмурно, сильный ветер. |
| 06.10.2023 | 00:58:37 | 14.5 | Ясно | Небо прояснилось, потеплело. |

Рисунок 3 – Пример заполнения таблицы данных

5. При необходимости удаления одной или нескольких записей можно воспользоваться контекстным меню либо комбинациями клавиш «Delete» и «Shift + Delete» для удаления одной или всех записей соответственно (рисунок 4).

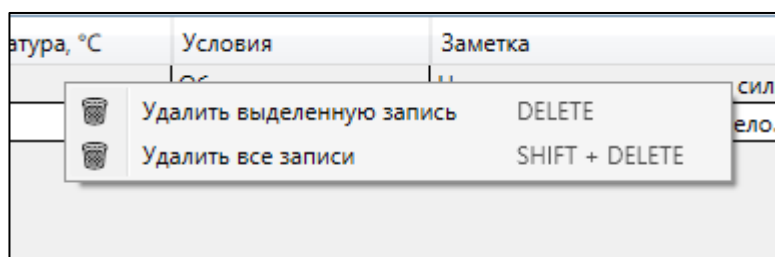


Рисунок 4 – Контекстное меню удаления записей

6. После добавления всех необходимых записей данных можно выполнить одно из следующих действий:

6.1. Сохранить данные в файл (рисунок 5):

- 6.1.1. Нажать кнопку «Сохранить» в левой области «Сохранение в файл» либо использовать комбинацию «Ctrl + S»;

- 6.1.2. Путь к файлу отобразится в поле «Путь к файлу»;

Примечание: имя файла задаётся уникально и автоматически генерируется программой, другие сохранённые файлы при этом не удаляются. Формат сохранения файлов с данными указывается в конфигурации программы и может быть JSON, XML или CSV (по умолчанию – JSON). Примеры каждого из форматов файлов приведены на рисунках 6 а-в.

6.1.3. Для просмотра папки со всеми файлами данных необходимо нажать кнопку «Открыть в проводнике»;

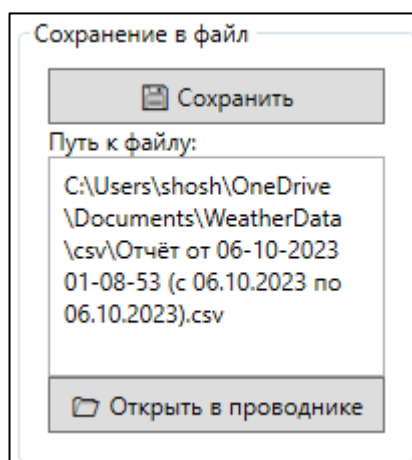


Рисунок 5 – Область сохранения данных в файл

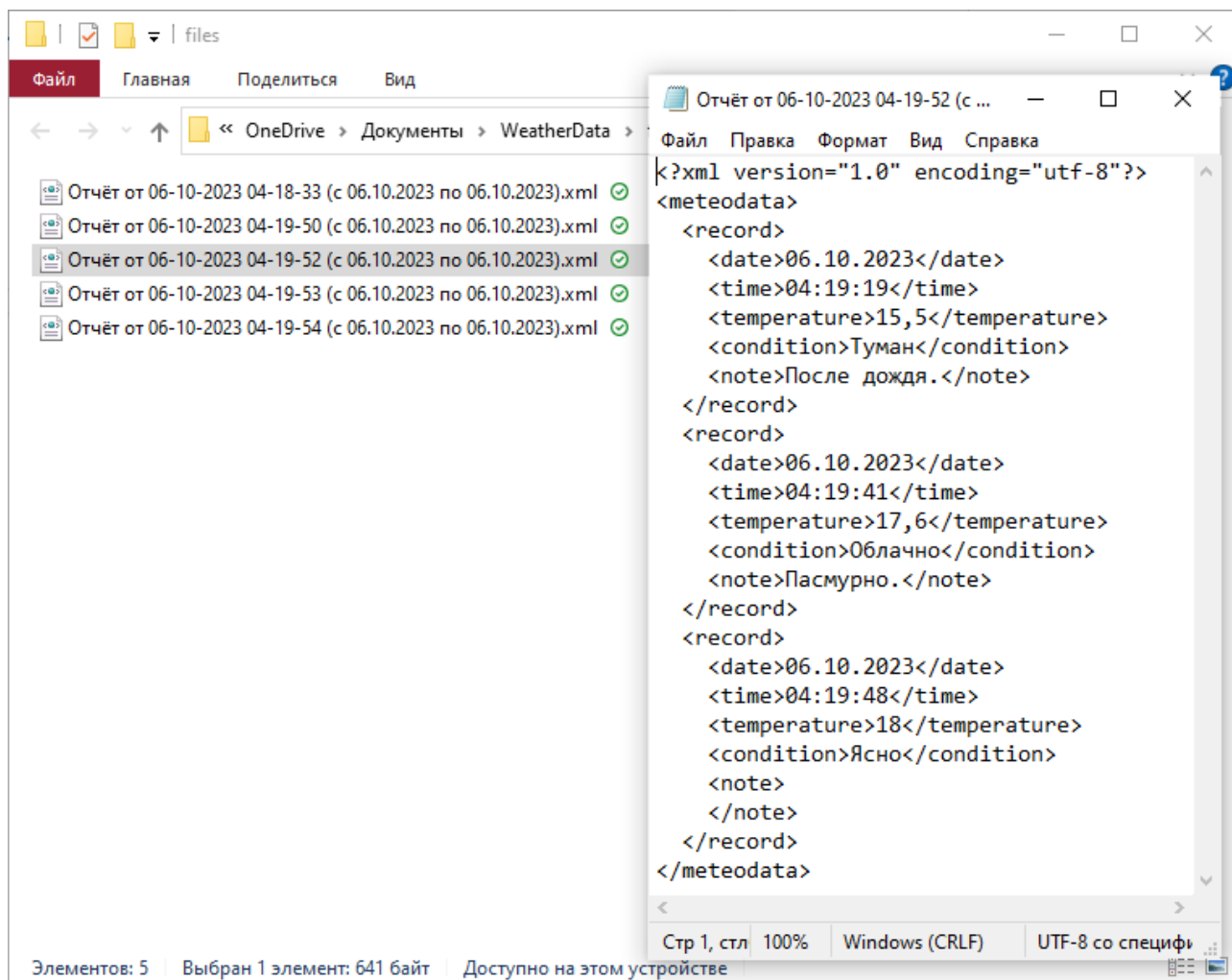


Рисунок 6а – Пример содержания XML-файла

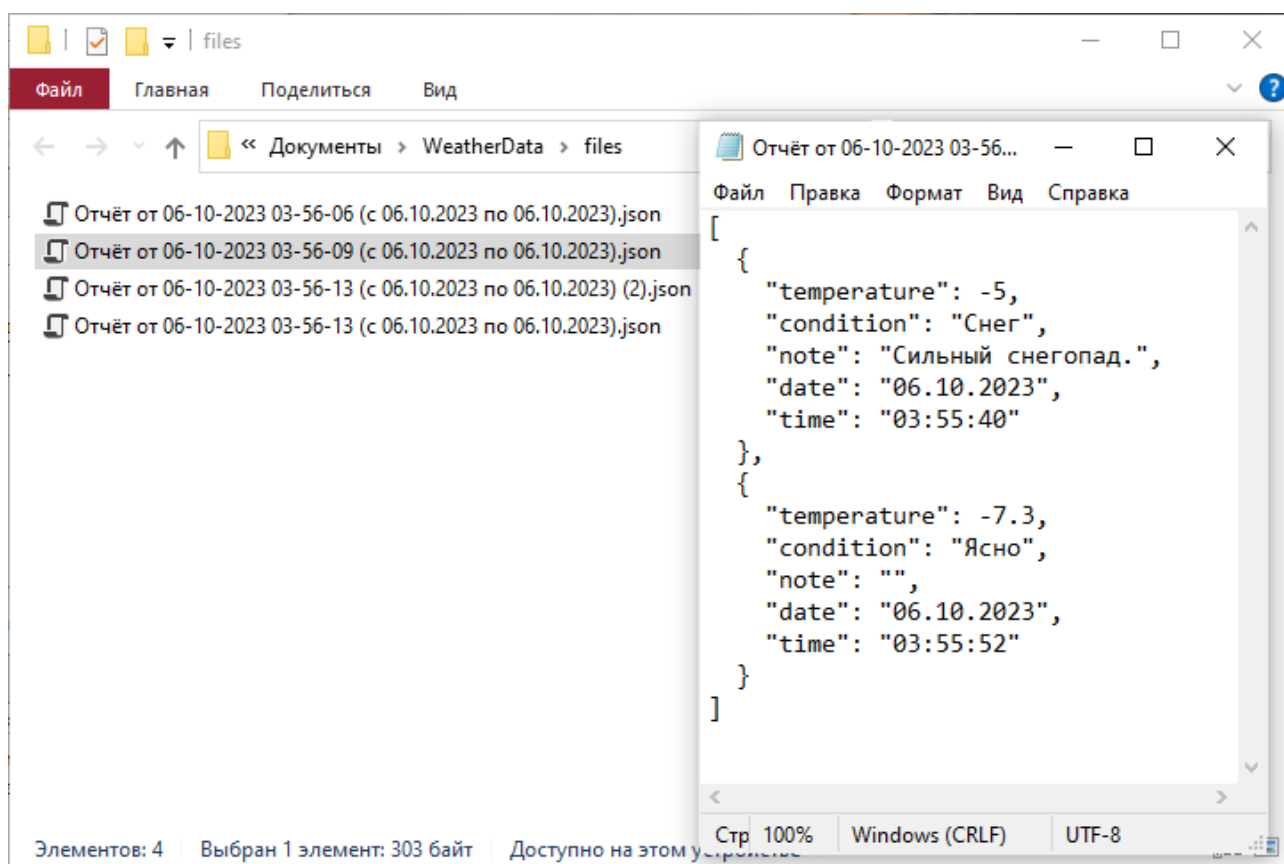


Рисунок 6б – Пример содержания JSON-файла

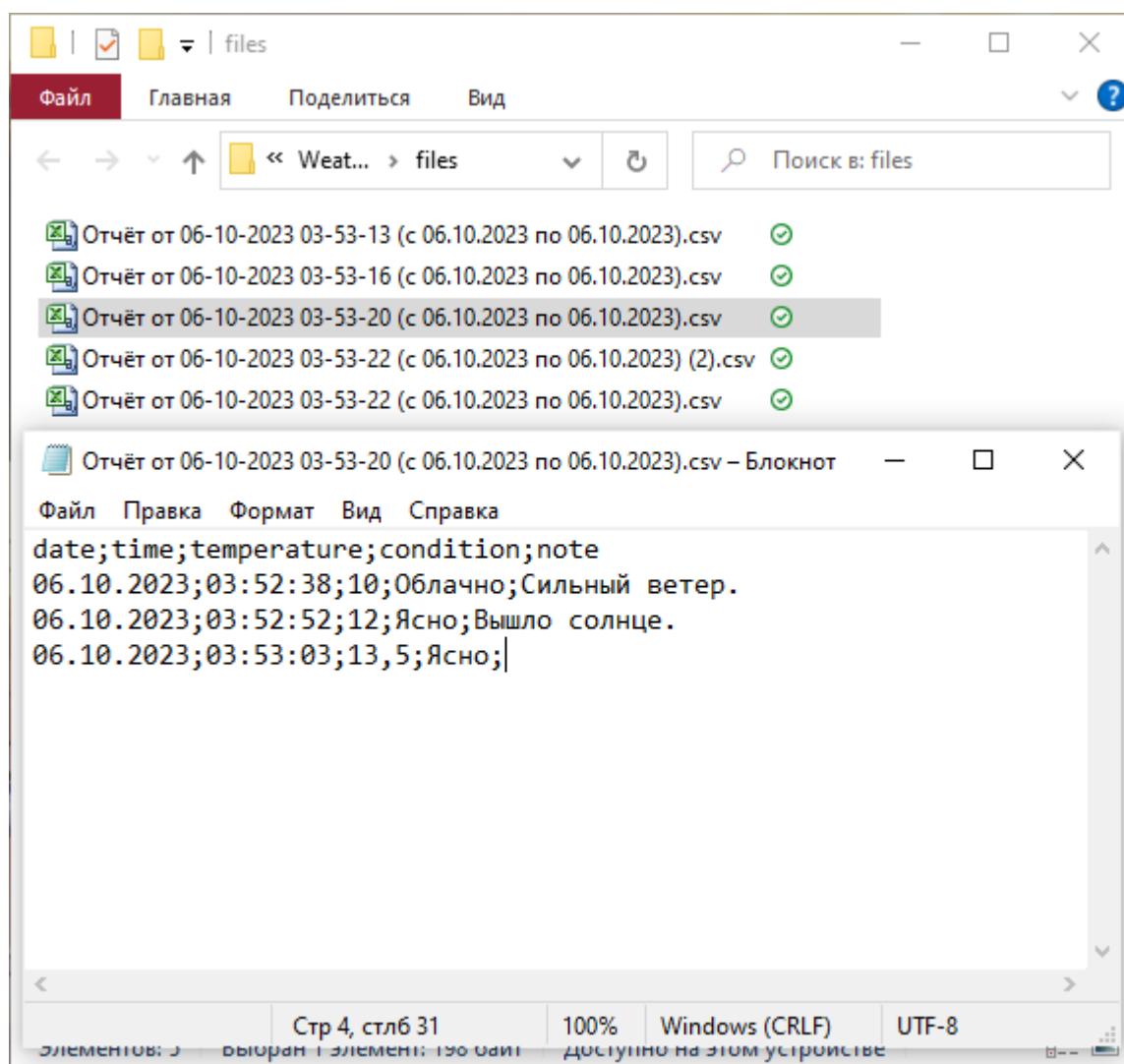


Рисунок 6в – Пример содержания CSV-файла

6.2.Сформировать текстовый отчёт (рисунок 7):

6.2.1. Нажать кнопку «Сформировать отчёт» в области «Отчёт»;

6.2.2. В текстовом поле в области «Отчёт» будет выведен структурированный текст отчёта со всеми данными из таблицы;

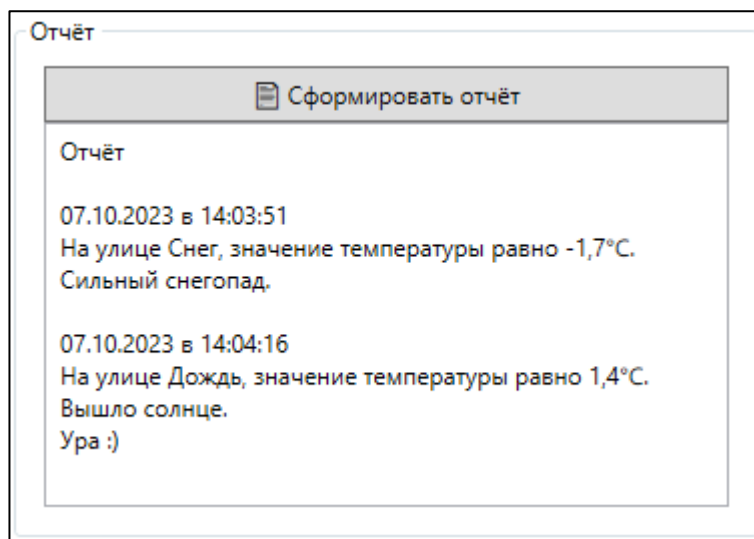


Рисунок 7 – Область формирования отчёта

6.3. Сохранить данные в базу данных (рисунок 8):

6.3.1. Нажать кнопку «Сохранить» в правой области «Сохранение в базу данных» либо использовать комбинацию «Ctrl + D»;

6.3.2. Путь к файлу базы данных отобразится в поле «Путь к файлу»;

Примечание: данные добавляются к уже существующим в базе данных. Название файла базы данных прописано в файле конфигурации программы (по умолчанию – «database.db»). Если файл базы данных отсутствует, он будет создан автоматически. Для работы с базой данных используется SQLite.

6.3.3. Для просмотра папки с базой данных необходимо нажать кнопку «Открыть в проводнике» (рисунок 9);

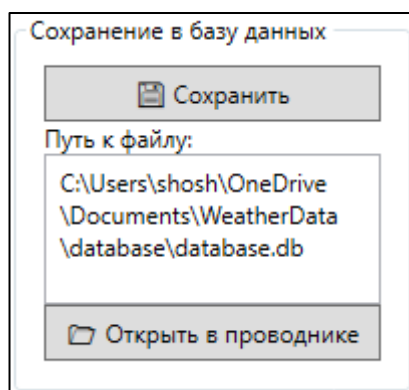


Рисунок 8 – Область сохранения данных в базу данных

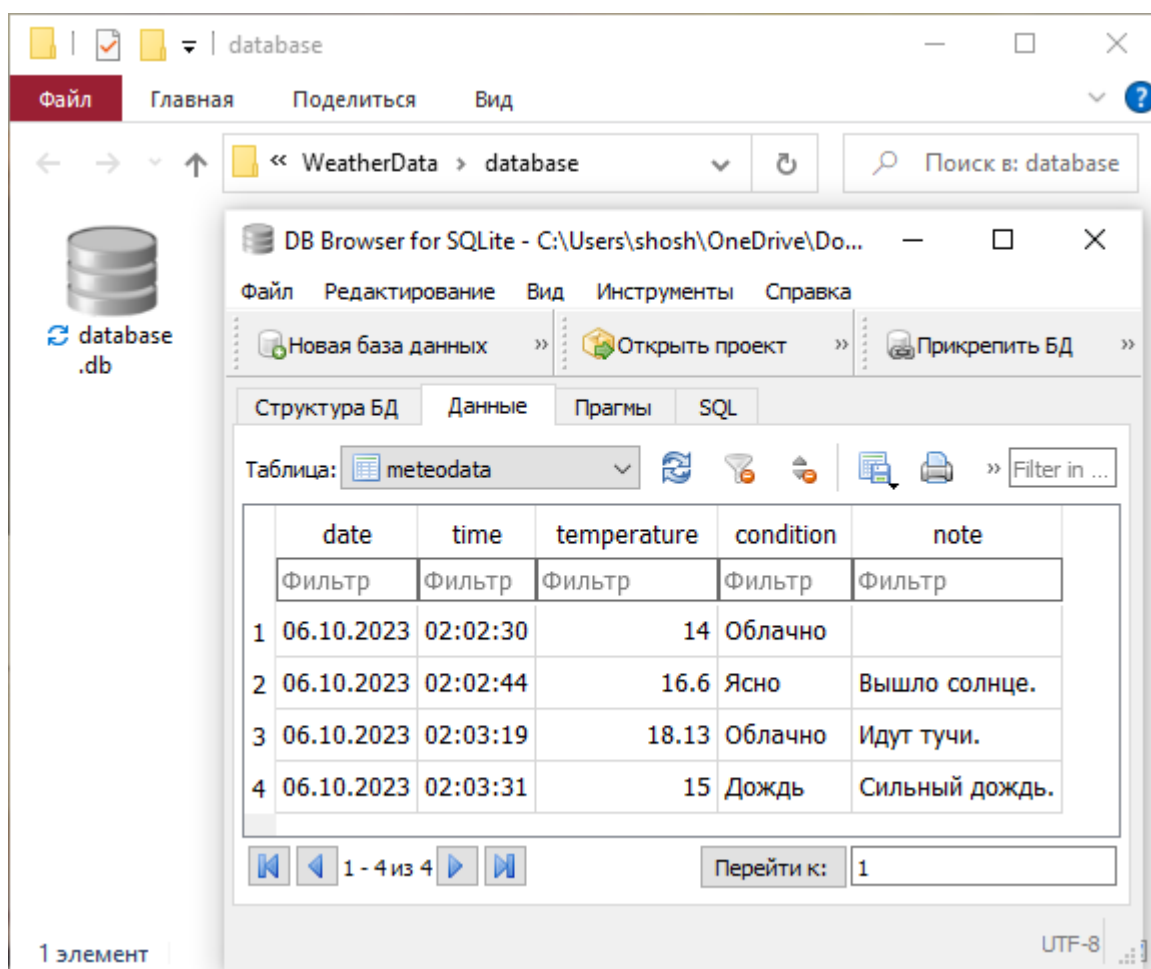


Рисунок 9 – Пример просмотра содержания базы данных при помощи программы «BD Browser for SQLite»

7. После обработки данных работа с программой закончена и её можно закрывать. Однако, если необходимо продолжить работу, то можно удалить все записи (см. шаг 5) и снова работать с программой, начиная с шага 2.

Предварительная настройка приложения

Для того, чтобы настроить параметры работы программы перед её запуском, откройте в любом текстовом редакторе файл «App.config», который расположен в одной директории с программой (рисунок 10). Возможные значения «value» для каждого параметра «key» представлены в таблице 1.

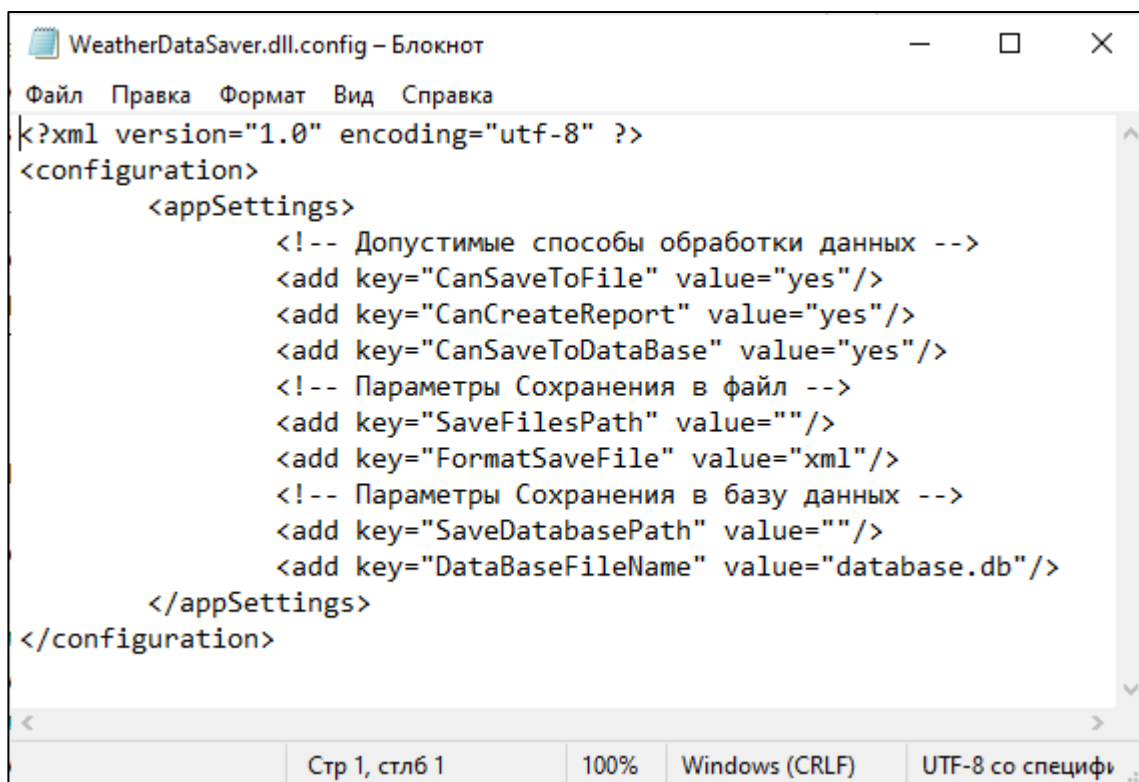


Рисунок 10 – Содержание файла конфигурации программы

Таблица 1. Параметры конфигурации программы.

| Параметр (key) | Значения (value) | Описание | Рисунки |
|------------------------------------|------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|---------|
| Способы обработки данных | | | |
| CanSaveToFile | yes/no | Возможность сохранять данные в файл | 11-14 |
| CanCreateReport | yes/no | Возможность формировать текстовый отчёт | |
| CanSaveToDataBase | yes/no | Возможность сохранять данные в базу данных | |
| Настройки сохранения в файл | | | |
| SaveFilePath | Без значения/ абсолютный путь/ относительный путь | Путь к папке для сохранения файлов. Если оставить пустым, то путь формируется в папке Документы текущего пользователя | 15 |
| FormatSaveFile | json/xml/csv | Формат сохранения файлов | ба-в |
| Настройки сохранения в базу данных | | | |
| SaveDatabasePath | Без значения/ абсолютный путь/ относительный путь | Путь к папке для сохранения базы данных. Если оставить пустым, то путь формируется в папке Документы текущего пользователя | 15 |
| DataBaseFileName | Имя файла | Название файла базы данных с расширением .db . По умолчанию «database.db». | 15 |

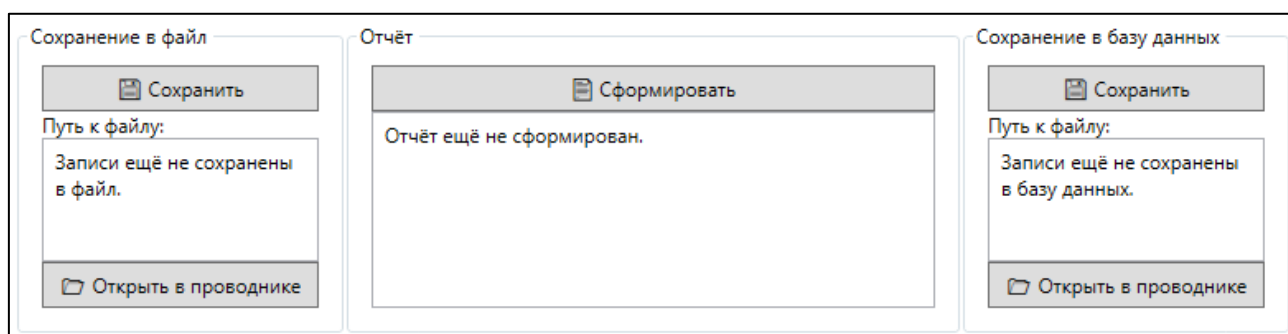


Рисунок 11 – Программа со всеми способами обработки данных

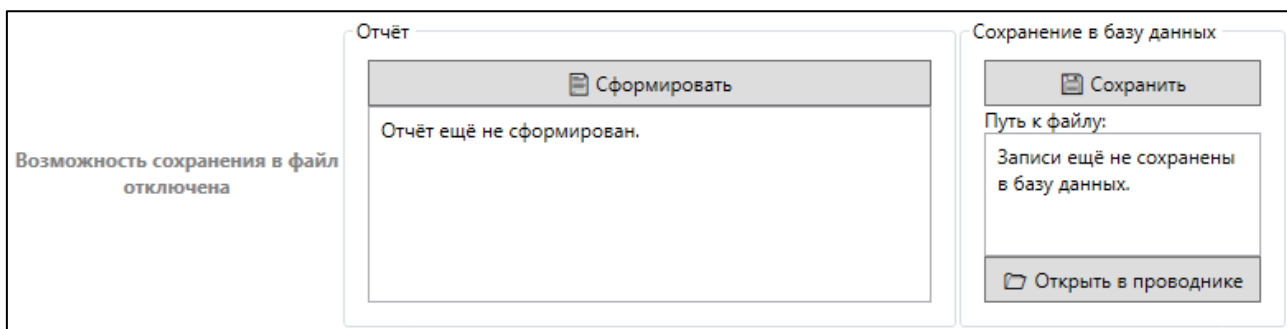


Рисунок 12 – Программа без возможности сохранять данные в файл

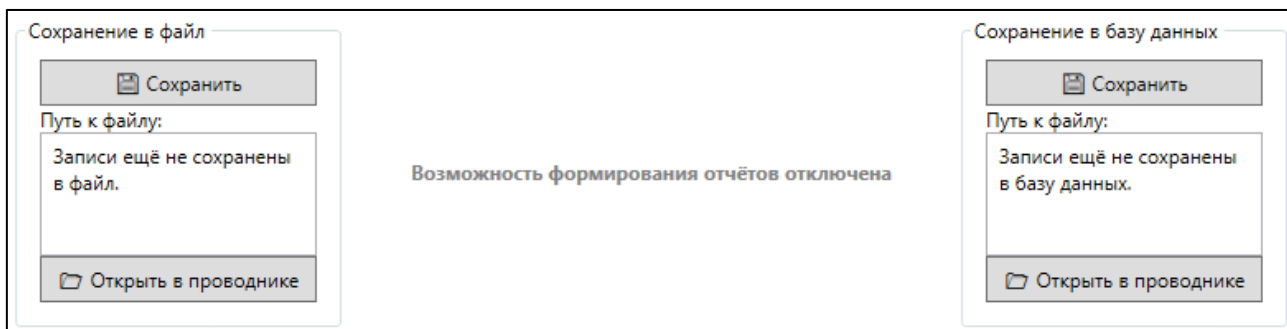


Рисунок 13 – Программа без возможности формировать текстовый отчёт

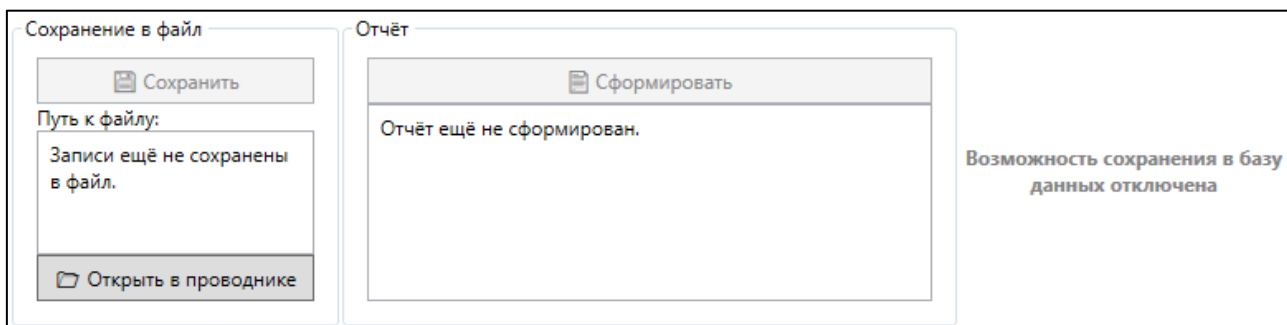


Рисунок 14 – Программа без возможности сохранять данные в базу данных

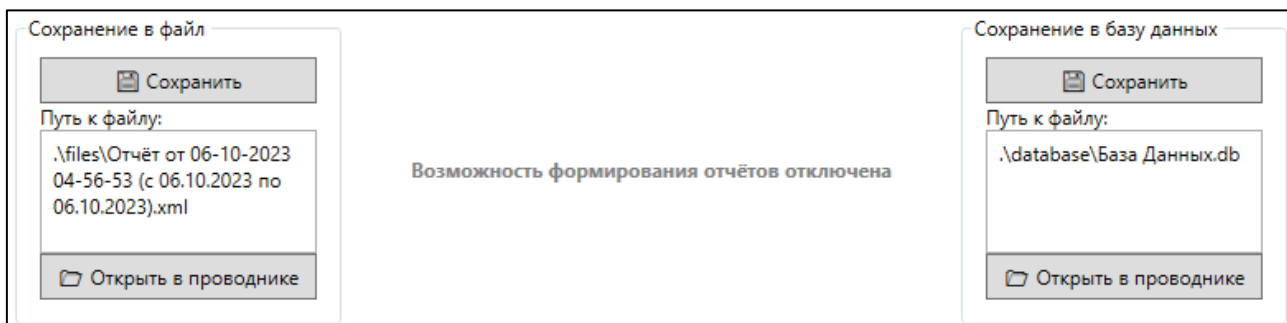


Рисунок 15 – Изменённые пути сохранения и название базы данных

Логирование процесса работы программы

Данные о работе программы автоматически сохраняются в папку «logs», расположенную в каталоге с программой. В названии файла логирования прописана дата и время запуска программы. В случае возникновения каких-либо ошибок в работе программы можно открыть файл лога и увидеть все действия, которые предшествовали или вызвали сбой (рисунок 16).

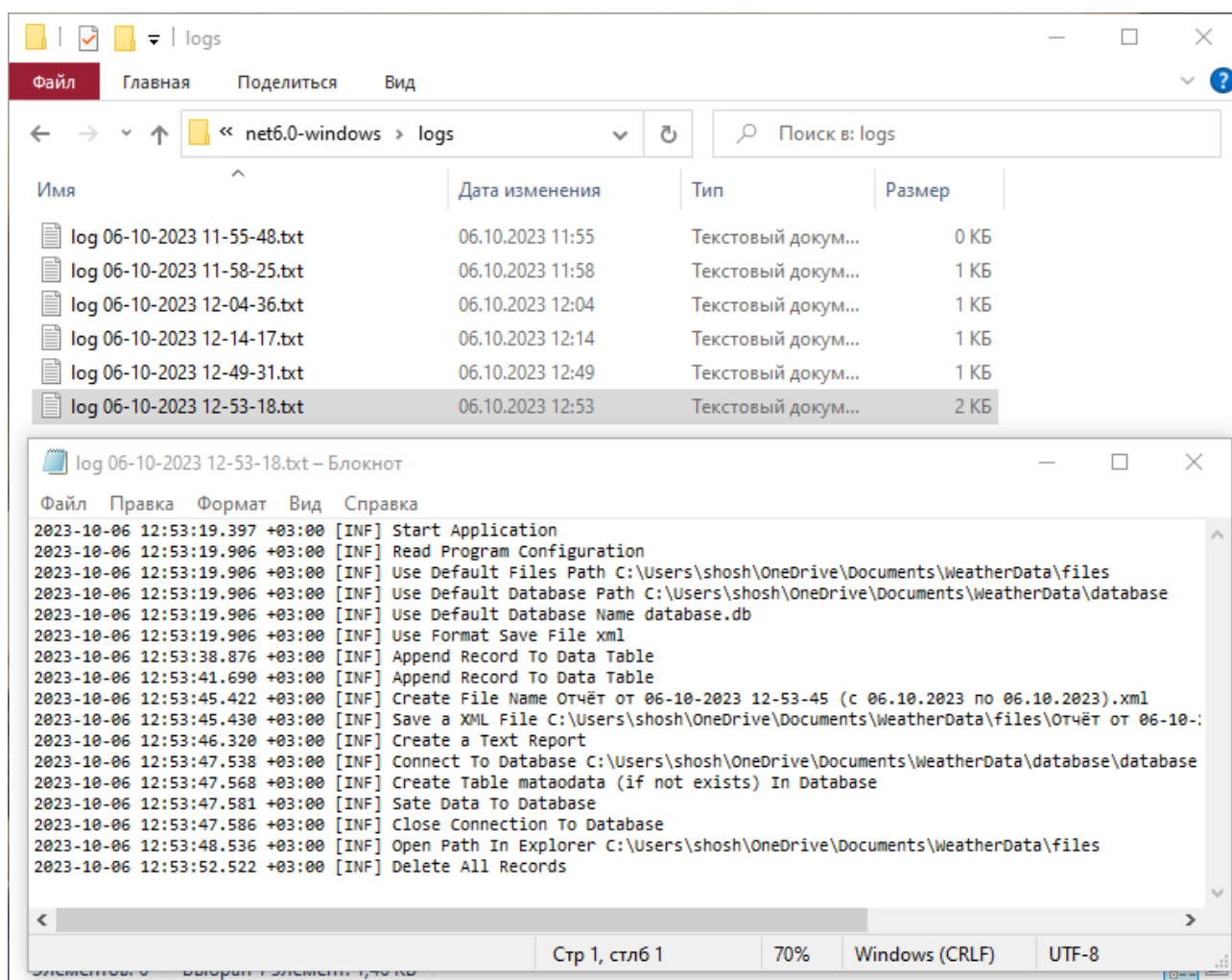


Рисунок 16 – Пример открытого файла логирования программы