**Раздел 1. Практическая значимость разрабатываемого программного средства**

Разрабатываемое программное средство (ПС) — CSV-Viewer — предназначено для удобного преобразования данных из CSV-файлов в HTML-таблицу, обеспечивая визуально понятное представление информации.

CSV-файлы широко используются для хранения и обмена табличными данными, однако их чтение в исходном текстовом виде затруднительно. Разработка данного ПС позволяет автоматически формировать структурированное представление данных в виде HTML-таблицы, обеспечивая удобное форматирование, проверку ошибок и выбор визуального оформления.

Основными пользователями могут быть разработчики, аналитики, офисные работники и все, кто работает с CSV-данными и нуждается в их удобном отображении. Программа будет полезна как для личного использования, так и для применения в организациях, где необходимо преобразовывать CSV-данные в читаемый формат.

Данное ПС может применяться в различных сценариях:

* автоматизация отчетов и анализа данных: например, в компании регулярно формируются CSV-отчеты о продажах. Без CSV-Viewer сотрудникам приходится вручную загружать файлы в Excel, настраивать стили и преобразовывать данные в HTML для отчётности. С помощью разработанного ПС можно быстро создать таблицу в нужном формате и визуальном стиле без лишних действий;
* использование в качестве программного модуля на веб-ресурсах: CSV-Viewer может быть использован на сайтах, сервисах и блогах для автоматического представления табличных данных. Например, финансовые ресурсы могут динамически отображать биржевые котировки, интернет-магазины — обновляемые списки товаров, а блоги — сравнительные таблицы характеристик товаров или услуг;
* публикация результатов опросов: допустим, организация проводит опрос среди сотрудников или клиентов, собирая данные в CSV-файл. С помощью программы можно легко преобразовать эти данные в наглядную HTML-таблицу и разместить на сайте или внутреннем портале, чтобы результаты были удобны для просмотра.

Если бы не было данного программного средства, то пользователям пришлось бы либо вручную редактировать CSV-файлы в текстовых редакторах, что неудобно и может привести к ошибкам, либо использовать сложные табличные процессоры, которые не всегда подходят для простого и быстрого преобразования данных в HTML-формат. Также без данного ПС отсутствовала бы возможность автоматизированной проверки корректности CSV-файлов перед их конвертацией.

**Раздел 2. Функциональные требования к ПС**

Программное средство CSV-Viewer должно обеспечивать выполнение следующих функций:

1. Проверка исходного файла: пользователь вводит путь к CSV-файлу через консоль; программа проверяет корректность пути и формата файла, анализирует структуру данных. Реакция программы на возможные ошибки:

* некорректный путь к CSV-файлу: программа завершает работу, выводя соответствующее сообщение об ошибке;
* CSV-файл пустой: программа завершает работу, выводя соответствующее сообщение;
* пустые значения в строке заголовка: программа завершает работу, выводя соответствующее сообщение об ошибке;
* пустая строка в CSV-файле или пустые значения в строке данных: программа информирует об этом пользователя, выводя соответствующее предупреждение, указав номер некорректной строки; программа пропускает некорректную строку, продолжая свою работу;
* неверное количество столбцов в строке данных: программа завершает работу, информируя пользователя о номере некорректной строки и количествах найденных и ожидаемых столбцов;
* отсутствие данных в CSV-файле: программа завершает работу, выводя соответствующее сообщение об ошибке.

Пример корректного исходного файла:

Id,Имя,Возраст

1,Артём,25

2,Валерия,30

Пример некорректного исходного файла:

Id,Имя,,

1;Артём

2,Валерия,30,438

1. Генерация HTML-таблицы: преобразование содержимого CSV-файла в HTML-таблицу; корректное отображение данных, включая специальные символы; возможность выбора темы оформления.

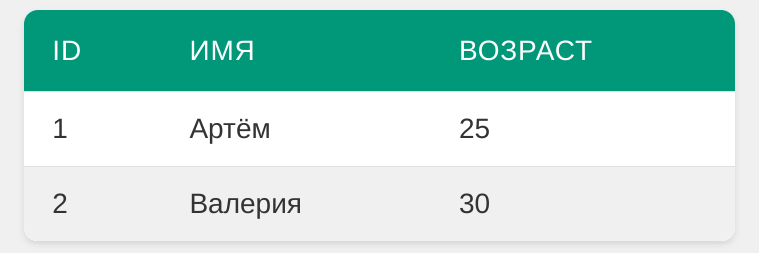
Пример исходного файла:

Id,Имя,Возраст

1,Артём,25

2,Валерия,30

Пример сгенерированной HTML-таблицы:



1. Выбор темы оформления и шрифта: пользователь через консоль выбирает одну из доступных тем и один из доступных шрифтов; программа применяет выбранное оформление к итоговому HTML-файлу.

Доступные темы оформления HTML-таблицы: светлая (light), темная (dark), зеленая (green).

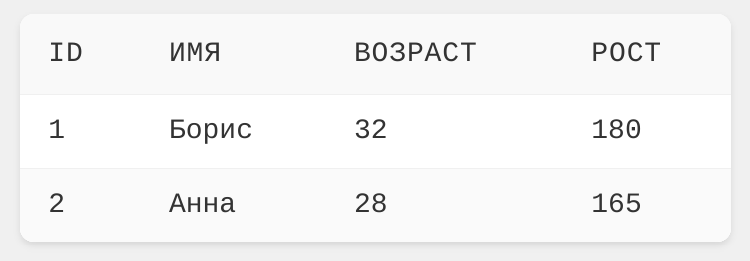
Доступные варианты шрифтов: Arial, Courier New, Times New Roman.

Способ указания темы, шрифта: программа просит пользователя ввести одну из тем и шрифтов, предоставляя доступные варианты и их номера; в квадратных скобках программа указывает вариант по умолчанию, который будет использоваться, если пользователь не станет вводить значение; пользователь по желанию вводит номер темы и шрифта через консоль.

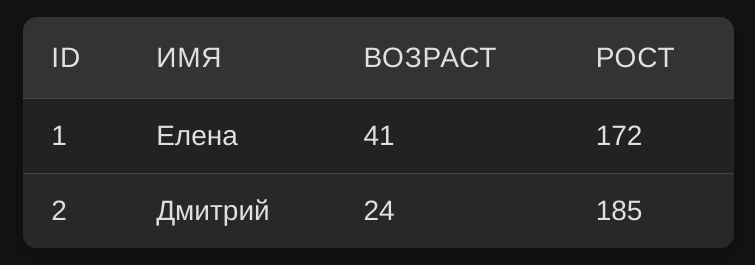
При вводе некорректного значения темы или шрифта программа выводит соответствующее сообщение об ошибке, завершая свою работу.

Примеры различных стилей оформления HTML-таблицы:

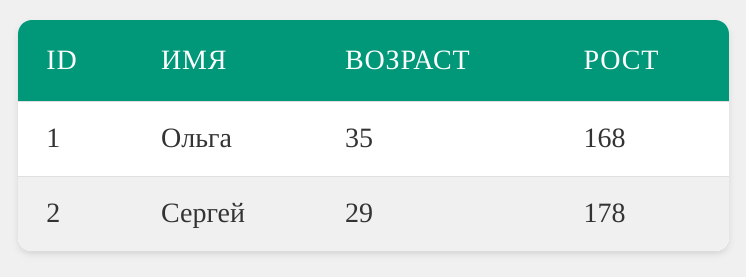
1. Тема - светлая (light), шрифт - Courier New:



1. Тема - темная (dark), шрифт - Arial:



1. Тема - зеленая (green), шрифт - Times New Roman:



1. Сохранение и просмотр результата: итоговый HTML-файл сохраняется в указанной пользователем директории; возможность посмотреть сгенерированный файл, открыв его в браузере.

Пользователь указывает абсолютный или относительный путь до директории, в которую желает сохранить итоговый файл (например, C:\Users\Alex\Desktop\).

Программа проверяет введенный путь к директории:

* если директория существует, программа сохраняет в неё итоговый файл, выводя соответствующее сообщение и указанием полного пути до сгенерированного файла;
* в случае отсутствия введенной директории программа выводит соответствующее сообщение, завершая свою работу.