**Программа управления персоналом**

1. **Цель и функциональные требования**:
   * Программа должна была обеспечивать управление персоналом: добавление, удаление, сортировка, фильтрация и сохранение данных сотрудников, таких как имя, возраст и отдел.
   * Также был реализован функционал для работы с текстовыми файлами, которые содержат информацию о сотрудниках в формате CSV.
2. **Структура данных**:
   * Для хранения данных использовался класс **Employee**, который содержал поля: имя сотрудника, возраст и отдел.
   * Данные хранились в **List**, что обеспечивало быстрый доступ и удобное управление. Однако, как и в предыдущей программе, при большом объеме данных такой подход может быть неэффективным.
3. **Графический интерфейс**:
   * В программе использовался интерфейс на **Swing**, который содержал таблицу для отображения сотрудников, кнопки для добавления и удаления данных, а также поля для ввода информации.
   * Цветовая схема была сине-белой, что придавало интерфейсу современный вид и улучшало восприятие.
4. **Работа с файлами**:
   * Программа поддерживала чтение данных из файлов и их запись. При загрузке данных из файла строки разбивались по запятой, что позволило извлечь необходимые поля.
   * Были проблемы с загрузкой файлов, если данные в них не соответствовали формату (например, пропущены или пустые поля).
5. **Обработка ошибок**:
   * В процессе работы с файлами программа корректно обрабатывает ошибки ввода-вывода. Однако, если файл имеет некорректный формат, программа не всегда давала пользователю точную информацию о том, что именно не так с файлом.
6. **Сортировка и фильтрация**:
   * Реализована сортировка сотрудников по возрасту и отделу. Также была добавлена возможность фильтровать сотрудников по отделам.
   * Фильтрация и сортировка работают корректно, но для большого количества данных программа может работать медленно.
7. **Прочие проблемы**:
   * Программа не оптимизирована для работы с большими объемами данных, что приводит к замедлению при большом числе сотрудников.
   * Также интерфейс не адаптирован для работы с небольшими экранами, что может создавать неудобства для пользователей с низким разрешением экрана.

**Программа управления персоналом**

1. **Проверка формата файлов**:
   * Добавить проверку корректности данных при загрузке файла, чтобы программа могла уведомить пользователя о некорректных строках или полях в файле.
   * Можно использовать CSV-парсер с поддержкой обработки ошибок или предусмотреть механизмы для исправления ошибок в данных.
2. **Оптимизация производительности**:
   * Для улучшения работы с большими объемами данных можно использовать более эффективные алгоритмы сортировки или внедрить пагинацию для отображения данных.
   * Также стоит рассмотреть возможность использования **SQLite** или другой базы данных для хранения данных, что значительно ускорит работу с большими объемами данных.
3. **Поддержка мобильных устройств и разных разрешений**:
   * Интерфейс должен быть более адаптирован для работы на устройствах с разным разрешением экрана, включая смартфоны и планшеты.
   * Использовать более гибкие компоненты, такие как **JTable** с возможностью масштабирования и адаптации к экрану.