**Файловая База данных. Реализация.**

Выполнил: Айдов Виталий. 18ПИ1. (12.03.2020)

Используемый\_язык:Графический\_модуль – Java:JavaFx.

Условия разработки: JavaSDK 13.0.2, IntellijIDEA Educational Edition 2019.3 x64

Исходный код: <https://github.com/voidmirror/JavaFX_test>

Описание работы:

Файловая база данных, эмулирующая базовую запись аккаунта браузерной онлайн-стратегии Supernova (<https://supernova.ws/>).

Поля Базы данных:

* UserName - <String> Никнейм пользователя. Ключевое поле, не может повторяться.
* Password - <String> Пароль пользователя.
* E-mail - <String> Электронная почта пользователя.
* BlackMatter - <int> Количество "Тёмной материи", доступной пользователю.
* UserLevel - <int> Уровень пользователя в игре.
* UserVIP - <boolean> Наличие VIP-привилегий.

Файловое представление Базы данных: "src/database.txt"

Внутреннее представление Базы данных: Main.*accountList*.getDataBaseHashMap()

Реализованные функции:

* **Добавление новой учётной записи.** Сложность: О(1). HashMap позволяет за константное время добавлять новые поля в себя. Реализована проверка на существование по ключевому полю (UserName).
* **Редактирование учётной записи.** Сложность: О(1). Аналогично добавлению новой учётной записи. При наличии указанного ключевого поля, существующая запись редактируется, иначе создаётся новая.
* **Сохранение Базы данных в файле.** Файл: "src/database.txt". Структура: Каждой учётной записи соответствует одна строка " UserName Password E-mail BlackMatter UserLevel UserVIP"
* **Развёртывание Базы данных из файла.** Сложность: О(n). Проходит по всем строкам файла "src/database.txt", создавая новые записи в HashMap. Гарантировано, что внутри HashMap учётные записи не повторяются. Соответственно, последующая запись в файл из HashMap имеет аналогичную гарантию.
* **Поиск по заданному полю с отображением.** Сложность: О(1). Поиск по ключевому полю в HashMap осуществляется за О(1) благодаря вычислению hashcode ключевого поля и мгновенного доступа к полю HashMap по значению полученного hashcode. Сложность: О(n). Поиск по не ключевым полям осуществляется линейно, перебором ключевых значений, записью нужных полей в объект TableView <ObservableList>.
* **Удаление Базы данных.** Удаление файла Базы данных с носителя. Реализована защита от случайного удаления. Нажатие кнопки "Delete DB" спровоцирует удаление только при условии ввода слова "delete" в соответствующее соседнее текстовое поле.
* **Удаление Учётной записи по ключу (UserName).** Удаление одной записи из базы данных. По ключевому значению удаляется объект из HashMap, затем перезаписывается текстовый файл, забирая текущие значения из HashMap.
* **Создание Backup-файла**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Время (мс) | 100 | 1000 | 10 000 | 100 000 |
| Добавление | 8 | 32 | 109 | 416 |
| Удаление | 0 | 2 | 11 | 65 |