

Факультет Программной Инженерии и Компьютерной Техники

Программирование

Лабораторная работа №5

Вариант № 810

Преподаватель: Харитонова Анастасия Евегеньевна

Выполнил: Бахтаиров Илья Андреевич

Группа: R3181

Санкт-Петербург

2020

**Текст задания**

**Разработанная программа должна удовлетворять следующим требованиям:**

* Класс, коллекцией экземпляров которого управляет программа, должен реализовывать сортировку по умолчанию.
* Все требования к полям класса (указанные в виде комментариев) должны быть выполнены.
* Для хранения необходимо использовать коллекцию типа java.util.ArrayList
* При запуске приложения коллекция должна автоматически заполняться значениями из файла.
* Имя файла должно передаваться программе с помощью: аргумент командной строки.
* Данные должны храниться в файле в формате xml
* Чтение данных из файла необходимо реализовать с помощью класса java.io.BufferedReader
* Запись данных в файл необходимо реализовать с помощью класса java.io.BufferedWriter
* Все классы в программе должны быть задокументированы в формате javadoc.
* Программа должна корректно работать с неправильными данными (ошибки пользовательского ввода, отсутсвие прав доступа к файлу и т.п.).

**В интерактивном режиме программа должна поддерживать выполнение следующих команд:**

* help : вывести справку по доступным командам
* info : вывести в стандартный поток вывода информацию о коллекции (тип, дата инициализации, количество элементов и т.д.)
* show : вывести в стандартный поток вывода все элементы коллекции в строковом представлении
* add {element} : добавить новый элемент в коллекцию
* update id {element} : обновить значение элемента коллекции, id которого равен заданному
* remove\_by\_id id : удалить элемент из коллекции по его id
* clear : очистить коллекцию
* save : сохранить коллекцию в файл
* execute\_script file\_name : считать и исполнить скрипт из указанного файла. В скрипте содержатся команды в таком же виде, в котором их вводит пользователь в интерактивном режиме.
* exit : завершить программу (без сохранения в файл)
* remove\_at index : удалить элемент, находящийся в заданной позиции коллекции (index)
* add\_if\_max {element} : добавить новый элемент в коллекцию, если его значение превышает значение наибольшего элемента этой коллекции
* shuffle : перемешать элементы коллекции в случайном порядке
* filter\_greater\_than\_difficulty difficulty : вывести элементы, значение поля difficulty которых больше заданного
* print\_ascending : вывести элементы коллекции в порядке возрастания
* print\_descending : вывести элементы коллекции в порядке убывания

**Формат ввода команд:**

* Все аргументы команды, являющиеся стандартными типами данных (примитивные типы, классы-оболочки, String, классы для хранения дат), должны вводиться в той же строке, что и имя команды.
* Все составные типы данных (объекты классов, хранящиеся в коллекции) должны вводиться по одному полю в строку.
* При вводе составных типов данных пользователю должно показываться приглашение к вводу, содержащее имя поля (например, "Введите дату рождения:")
* Если поле является enum'ом, то вводится имя одной из его констант (при этом список констант должен быть предварительно выведен).
* При некорректном пользовательском вводе (введена строка, не являющаяся именем константы в enum'е; введена строка вместо числа; введённое число не входит в указанные границы и т.п.) должно быть показано сообщение об ошибке и предложено повторить ввод поля.
* Для ввода значений null использовать пустую строку.
* Поля с комментарием "Значение этого поля должно генерироваться автоматически" не должны вводиться пользователем вручную при добавлении.

**Исходный код**

Доступен по ссылке:

<https://github.com/FJPO/ifmo_lab5>

**Диаграмма классов**

См Приложение.

**Документация**

Доступна по ссылке:

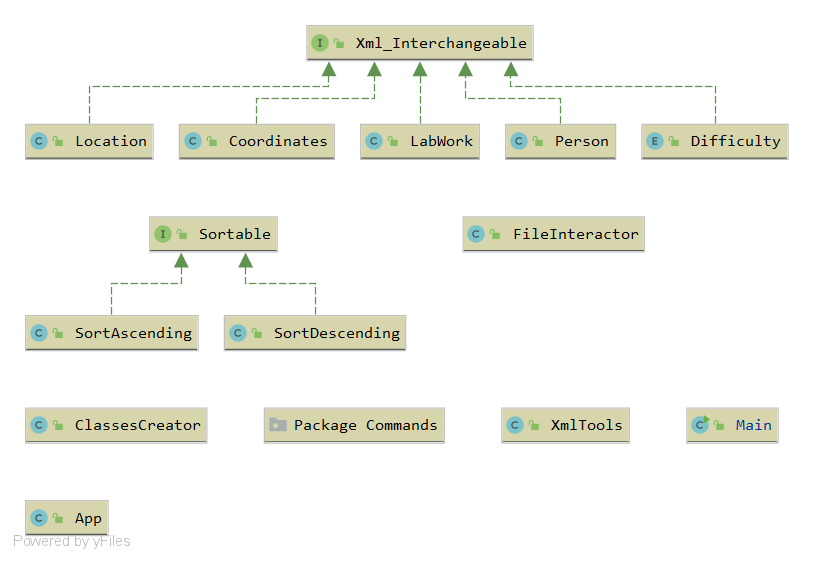
<https://github.com/FJPO/ifmo_lab5/tree/master/doc>

**Вывод**

Данная лабораторная работа на сегодняшний момент – самая объемная по количеству кода и приложенных усилий для ее написания. Это был мой первый опыт работы не только с удобными в плане возможности работы с переменным количеством элементов коллекциями, но и с несомненно полезными для долгосрочного хранения данных классами работы с файлами.

**Приложение**

UML-диаграмма:



UML-диаграмма пакета Commands:

