

Факультет программной инженерии и компьютерной техники

Дисциплина: Программирование

Лабораторная работа №5

Вариант № 311294

Выполнил: Васильев А.Ю.

№ группы: P3115

Преподаватель: Пашнин А.Д.

2021 год, 2 семестр

Текст задания.

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса Worker, описание которого приведено ниже.

**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.TreeMap
* 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.util.Scanner
* 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
* insert null {element} : добавить новый элемент с заданным ключом
* update id {element} : обновить значение элемента коллекции, id которого равен заданному
* remove\_key null : удалить элемент из коллекции по его ключу
* 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\_greater {element} : удалить из коллекции все элементы, превышающие заданный
* replace\_if\_greater null {element} : заменить значение по ключу, если новое значение больше старого
* remove\_greater\_key null : удалить из коллекции все элементы, ключ которых превышает заданный
* max\_by\_coordinates : вывести любой объект из коллекции, значение поля coordinates которого является максимальным
* print\_descending : вывести элементы коллекции в порядке убывания
* print\_field\_descending\_position : вывести значения поля position всех элементов 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.

**Описание хранимых в коллекции классов:**

public class Worker {

private Long id; //Поле не может быть null, Значение поля должно быть больше 0, Значение этого поля должно быть уникальным, Значение этого поля должно генерироваться автоматически

private String name; //Поле не может быть null, Строка не может быть пустой

private Coordinates coordinates; //Поле не может быть null

private java.time.LocalDateTime creationDate; //Поле не может быть null, Значение этого поля должно генерироваться автоматически

private Integer salary; //Поле может быть null, Значение поля должно быть больше 0

private Position position; //Поле не может быть null

private Status status; //Поле не может быть null

private Person person; //Поле не может быть null

}

public class Coordinates {

private Integer x; //Значение поля должно быть больше -716, Поле не может быть null

private Double y; //Максимальное значение поля: 943, Поле не может быть null

}

public class Person {

private java.time.LocalDateTime birthday; //Поле может быть null

private float height; //Значение поля должно быть больше 0

private String passportID; //Длина строки не должна быть больше 44, Строка не может быть пустой, Значение этого поля должно быть уникальным, Поле не может быть null

}

public enum Position {

DIRECTOR,

LABORER,

HUMAN\_RESOURCES,

HEAD\_OF\_DEPARTMENT,

MANAGER\_OF\_CLEANING;

}

public enum Status {

HIRED,

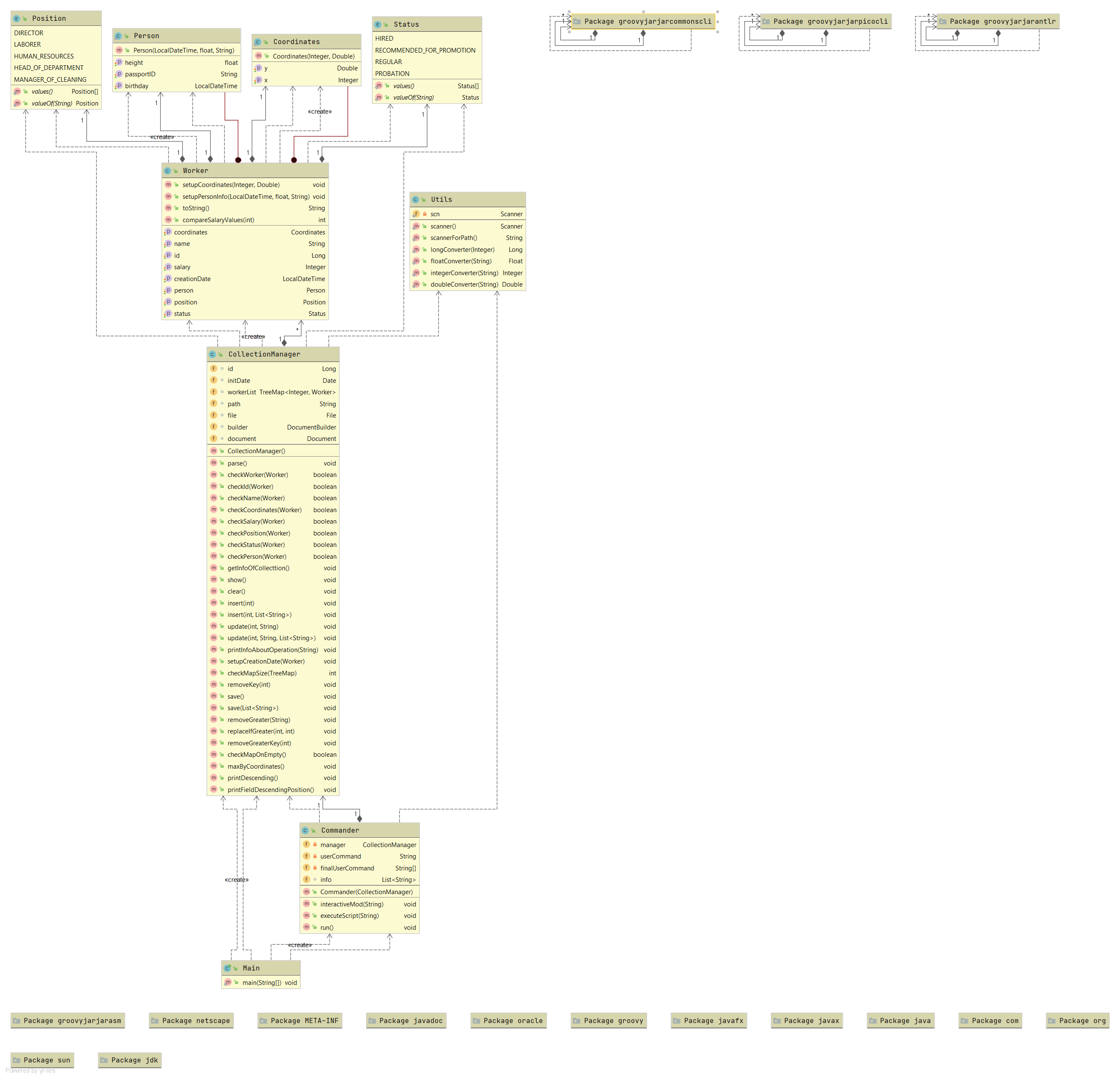
RECOMMENDED\_FOR\_PROMOTION,

REGULAR,

PROBATION;

}

Диаграмма классов разработанной программы.



Исходный код можно посмотреть по ссылке <https://github.com/wizarsi/lab5>

Выводы по работе.

В данной лабораторной работе, я познакомился с коллекциями, работал с потоками ввода вывода, осуществлял чтение и запись файлов, обрабатывал исключения, строил архитектуру консольного приложения. Все это дало мне практические навыки, которые пригодятся при работе с базами данных.