

Домашнее задание по теме «Java Collections. Stream API»

Формулировка задания:

1. Реализовать класс Автомобиль. У класса есть поля, свойства и методы.

Поля класса:

- а) Номер автомобиля;
- б) Модель;
- в) Цвет;
- г) Пробег;
- д) Стоимость.

Обратить внимание на переопределение метода toString, на сеттеры и геттеры, модификаторы доступа полей.

2. Проверить работу в классе Main, методе main.

3. Создать объект Java Collections со списком автомобилей.

4. Используя Java Stream API, вывести (можно сделать любые 2 пункта из 4):

1) Номера всех автомобилей, имеющих заданный в переменной цвет colorToFind или нулевой пробег mileageToFind.

2) Количество уникальных моделей в ценовом диапазоне от n до m тыс.

3) Вывести цвет автомобиля с минимальной стоимостью.

4) Среднюю стоимость искомой модели modelToFind

Входные данные:

[НОМЕР_АВТОМОБИЛЯ][МОДЕЛЬ][ЦВЕТ][ПРОБЕГ][ЦЕНА]

a123me|Mercedes|White|0|8300000

b873of|Volga|Black|0|673000

w487mn|Lexus|Grey|76000|900000

p987hj|Volga|Red|610|704340

c987ss|Toyota|White|254000|761000

o983op|Toyota|Black|698000|740000
p146op|BMW|White|271000|850000
u893ii|Toyota|Purple|210900|440000
l097df|Toyota|Black|108000|780000
y876wd|Toyota|Black|160000|1000000

Black, 0L

700_000L, 800_000L

Toyota

Volvo

Выходные данные:

Автомобили в базе:

Number	Model	Color	Mileage	Cost
a123me	Mercedes	White	0	8300000
b873of	Volga	Black	0	673000
w487mn	Lexus	Grey	76000	900000
p987hj	Volga	Red	610	704340
c987ss	Toyota	White	254000	761000
o983op	Toyota	Black	698000	740000
p146op	BMW	White	271000	850000
u893ii	Toyota	Purple	210900	440000
l097df	Toyota	Black	108000	780000
y876wd	Toyota	Black	160000	1000000

Номера автомобилей по цвету или пробегу: a123me b873of o983op l097df
y876wd

Уникальные автомобили: 4 шт.

Цвет автомобиля с минимальной стоимостью: Purple

Средняя стоимость модели Toyota: 744200,00

Средняя стоимость модели Volvo: 0,00

Программа реализуется в отдельной ветке git homeworks/homework011. При сохранении состояния программы (коммиты) пишется сообщение с описанием хода работы по задаче.

В корне папки с программой должен быть файл .gitignore.

Программа локально коммитится и публикуется в репозиторий GitHub на проверку.

Дополнительно:

1. Реализовать ввод и вывод программы в файл *.txt.
2. Вынести методы работы с автомобилем в папку repository интерфейс CarsRepository и его реализацию CarsRepositoryImpl.
3. Доработать программу до следующей структуры:

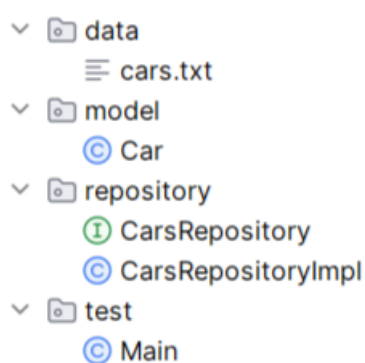


Рисунок 1. Структура проекта

Программа реализуется в отдельной ветке git homeworks/homework011Addition. При сохранении состояния программы (коммиты) пишется сообщение с описанием хода работы по задаче.

В корне папки с программой должен быть файл .gitignore.

Планируемый результат:

1. Ссылка на программу в репозитории GitHub;

2. Отчёт со скринами выполнения задач - постановка задачи, код задачи и результат в консоли IntelliJ Idea.

Описания плана работы:

Изучение функций Stream API с последующей реализацией функциональности.

Перечень инструментов, необходимых для реализации деятельности:

Персональный компьютер, JDK 17 (либо OpenJDK 17), IntelliJ Idea для разработки на Java, GIT, Tortoise GIT.