Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Ордена Трудового Красного Знамени

федеральное государственное бюджетное образовательное учреждение высшего образования

МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ

Кафедра «Математической кибернетики и информационных технологий»

ИТиП

Лабораторная работа №3

Выполнил:

студент группы БВТ2203

Иляков С.А.

Москва 2023

**Вариант №3**

**Задание 1:**

1. Создайте класс HashTable, который будет реализовывать хэш-

таблицу с помощью метода цепочек.

2. Реализуйте методы put(key, value), get(key) и remove(key), которые

добавляют, получают и удаляют пары «ключ-значение» соответственно.

3. Добавьте методы size() и isEmpty(), которые возвращают

количество элементов в таблице и проверяют, пуста ли она.

**Задание 2:**

Реализация хэш-таблицы для хранения информации о заказах

в интернет-магазине. Ключом является номер заказа, а значением - объект

класса Product, содержащий поля дата заказа, список товаров и статус

заказа. Необходимо реализовать операции вставки, поиска и удаления

заказа по номеру. Также необходимо реализовать метод для изменения

статуса заказа.

**Реализация**

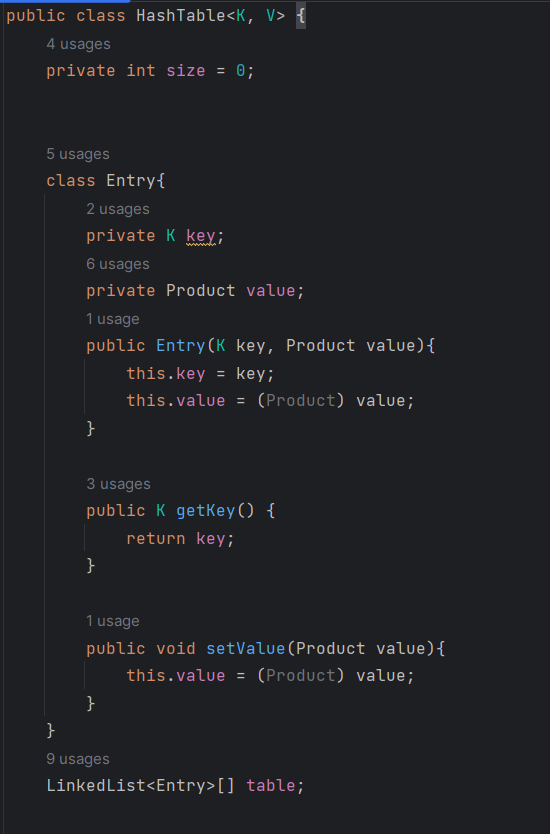
1) Создадим класс HashTable (здесь и далее Рис. 1). В нём добавим поле для подсчета числа подкласс Entry, который будет содержать в себе поля со значением и ключом, конструктор, а также методы для получения значения ключа и установки значения. Также создадим массив table из связных списков содержащих экземпляры подкласса Entry.

Рисунок 1 – Листинг класса Hashtable.

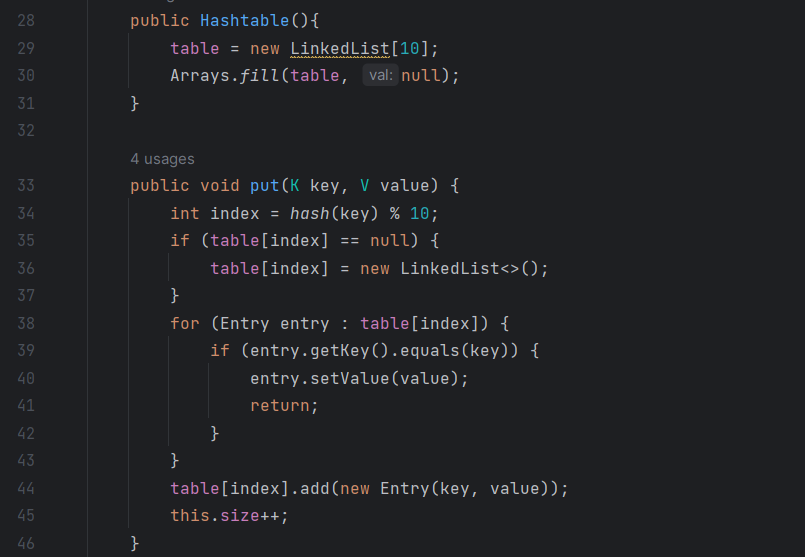
2) Далее идет конструктор по умолчанию, который создает массив из 10 элементов и заполняет их null значениями. Метод put, находит значение хэш функции для ключа, находит остаток от деления на длину массива и в полученный индекс массива добавляет новую пару ключ-значение в конец связанного листа и увеличивает количество элементов, если до этого ключа не было - перезапишет значение.

Рисунок 2 - Листинг класса Hashtable.

3) Метод get возвращает значение по ключу, для этого он вычисляет индекс массива, в котором нужно искать значение и возвращает его, в противном случаем, если нет такого элемента, он возвращает null. Метод remove в случае если элемент существует находит его и удаляет из связанного списка и, следовательно, из таблицы и при этом уменьшает количество элементов.



Рисунок 3 - Листинг класса Hashtable.

4) И последние два метода это isEmpty и size (Рис. 4), которые проверяют пустая ли таблица и возвращает ее размер.

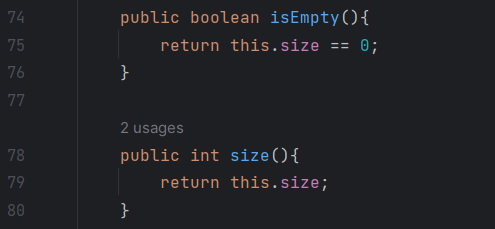


Рисунок 4 - Листинг класса Hashtable.

5) Для выполнения задания создадим класс Product, в котором будут поля для хранения даты, списка товаров и статуса доставки (Рис. 5).

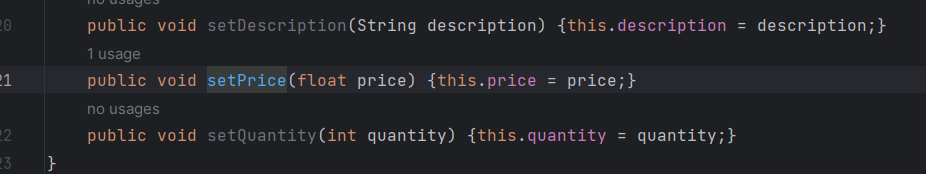
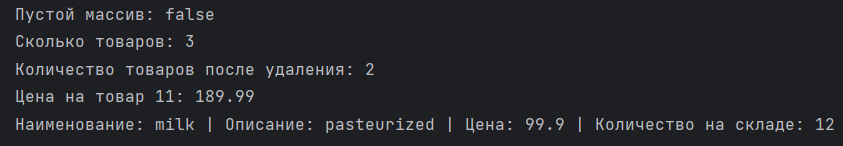


Рисунок 5, 6 – Листинг класса Order.

6)В классе Main (Рис. 6) реализуем созданную хэш таблицу. Для этого создадим пустую таблицу, добавим в нее 3 элемента, продемонстрируем работу методов isEmpty, size, remove, setStatus и get.



Рисунок 7 – Листинг Main.



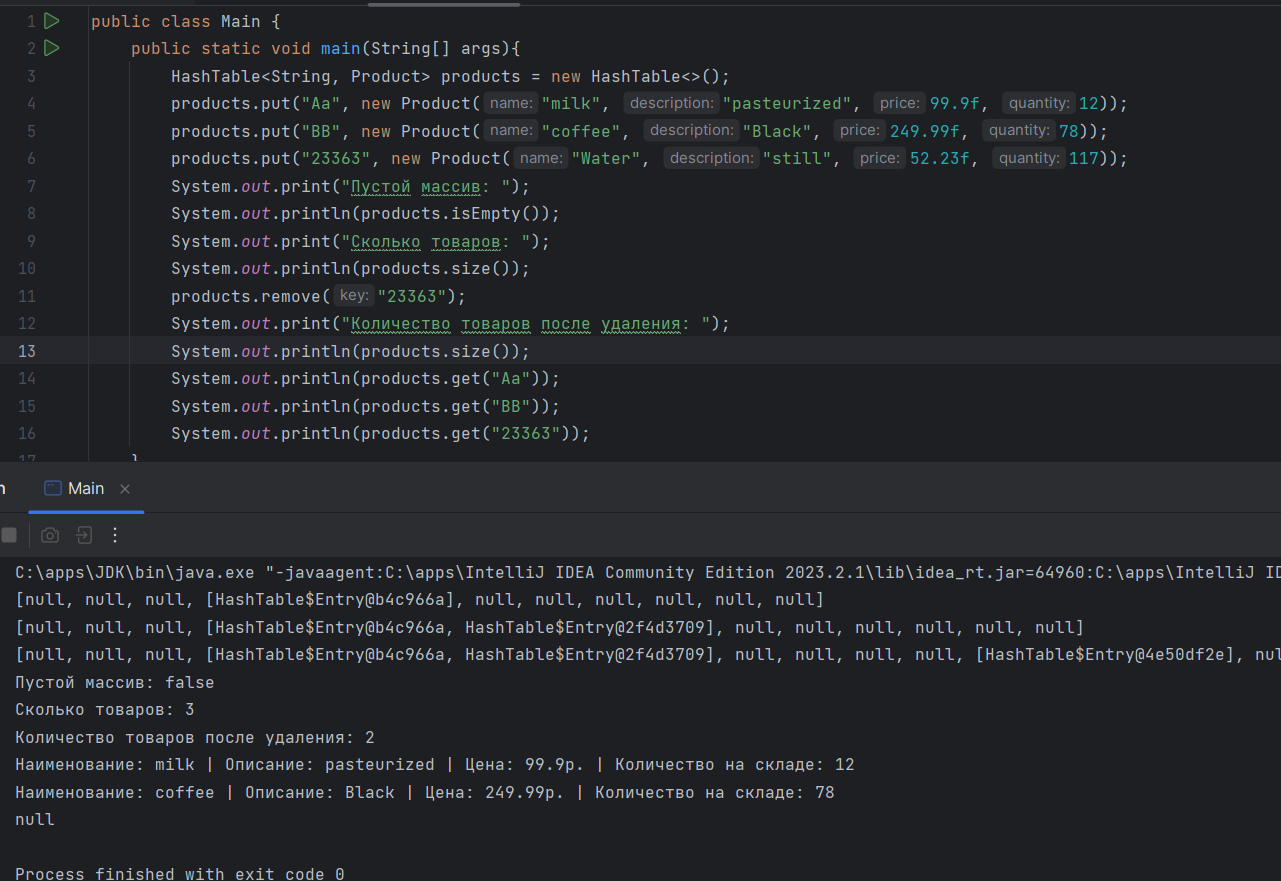


Рисунок 8, 9 – Результат работы.

**Вывод:**

1. Я освоил принцип работы хэш-таблицы, являющейся одной из самых востребованных в программировании.

2. Я написал свой класс HashTable, реализующий все основные методы стандартного класса, а также типизировал его для хранения информации о работниках.

Ссылка на Git - <https://github.com/Wex187/-.git>