**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»**

Факультет «Информационные технологии»

Кафедра «Математическая кибернетика и информационные технологии»

Дисциплина: «Информационные технологии и программирование»

**Лабораторная работа №3**

**«Класс Object. Работа с хэш-таблицами»**

Выполнил:

студент группы БВТ2402

Аралушкин Максим Дмитриевич

Руководитель:

Мосева Марина Сергеевна

Москва

2025

# Цель работы

Освоение работы с хэш-таблицами на Java. Изучение класса Object. Создание хэш-таблиц.

# Индивидуальное задание

* 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.

2) Реализация хэш-таблицы для хранения информации о студентах. Ключом является номер зачетной книжки, а значением — объект класса Student, содержащий поля имя, фамилия, возраст и средний балл. Необходимо реализовать операции вставки, поиска и удаления студента по номеру зачетки.

# Ход работы

Задание №1:

HashTable:

package lab3;

import java.util.LinkedList;

public class HashTable<K, V> {

    private LinkedList<Entry<K, V>>[] table;

    private int size;

    private int capacity;

    private static class Entry<K, V> {

        K key;

        V value;

        Entry(K key, V value) {

            this.key = key;

            this.value = value;

        }

    }

    @SuppressWarnings("unchecked")

    public HashTable(int capacity) {

        this.capacity = capacity;

        this.size = 0;

        table = new LinkedList[capacity];

        for (int i = 0; i < capacity; i++) {

            table[i] = new LinkedList<>();

        }

    }

    public HashTable() {

        this(16);

    }

    private int hash(K key) {

        return Math.abs(key.hashCode()) % capacity;

    }

    public void put(K key, V value) {

        int index = hash(key);

        for (Entry<K, V> entry: table[index]) {

            if (entry.key.equals(key)) {

                entry.value = value;

                return;

            }

        }

        table[index].add(new Entry<>(key, value));

        size++;

    }

    public V get(K key) {

        int index = hash(key);

        for (Entry<K, V> entry: table[index]) {

            if (entry.key.equals(key)) {

                return entry.value;

            }

        }

        return null;

    }

    public void remove(K key) {

        int index = hash(key);

        for (int i = 0; i < table[index].size(); i++) {

            Entry<K, V> entry = table[index].get(i);

            if (entry.key.equals(key)) {

                table[index].remove(i);

                size--;

                return;

            }

        }

    }

    public int size() {

        return size;

    }

    public boolean isEmpty() {

        return size == 0;

    }

}

Пример для HashTable:

package lab3;

public class TaskOne {

    public static void main(String[] args) {

        HashTable<Integer, String> studtable = new HashTable<>(10);

        System.out.println("Таблица пуста: " + studtable.isEmpty());

        System.out.println("Размер таблицы: " + studtable.size());

        System.out.println();

        studtable.put(101, "Иван Петров");

        studtable.put(102, "Мария Сидорова");

        studtable.put(103, "Алексей Козлов");

        studtable.put(104, "Елена Новикова");

        studtable.put(105, "Дмитрий Волков");

        System.out.println("После добавления 5 студентов:");

        System.out.println("Таблица пуста: " + studtable.isEmpty());

        System.out.println("Размер таблицы: " + studtable.size());

        System.out.println();

        System.out.println("Поиск студентов по ID:");

        System.out.println("ID 101: " + studtable.get(101));

        System.out.println("ID 103: " + studtable.get(103));

        System.out.println("ID 106: " + studtable.get(106));

        System.out.println();

        System.out.println("Обновление данных студента с ID 102:");

        studtable.put(102, "Мария Иванова");

        System.out.println("ID 102 после обновления: " + studtable.get(102));

        System.out.println("Размер таблицы после обновления: " + studtable.size());

        System.out.println();

        HashTable<String, String> subtable = new HashTable<>();

        subtable.put("MATH101", "Математический анализ");

        subtable.put("PHYS202", "Общая физика");

        subtable.put("CS303", "Программирование на Java");

        subtable.put("ENG404", "Английский язык");

        System.out.println("Таблица предметов:");

        System.out.println("MATH101: " + subtable.get("MATH101"));

        System.out.println("CS303: " + subtable.get("CS303"));

        System.out.println("Размер таблицы предметов: " + subtable.size());

        System.out.println();

        System.out.println("Пример удаления:");

        System.out.println("Размер таблицы студентов до удаления: " + studtable.size());

        System.out.println();

        studtable.remove(103);

        System.out.println("После удаления ID 103:");

        System.out.println("ID 103: " + studtable.get(103));

        System.out.println("Размер таблицы студентов после удаления: " + studtable.size());

        System.out.println();

        studtable.remove(101);

        studtable.remove(105);

        System.out.println("После удаления ID 101 и 105:");

        System.out.println("Размер таблицы студентов: " + studtable.size());

        System.out.println("ID 101: " + studtable.get(101));

        System.out.println("ID 102: " + studtable.get(102));

        System.out.println("ID 104: " + studtable.get(104));

        System.out.println();

        System.out.println("Таблица студентов пуста: " + studtable.isEmpty());

        studtable.remove(102);

        studtable.remove(104);

        System.out.println("После удаления всех студентов:");

        System.out.println("Размер таблицы студентов: " + studtable.size());

        System.out.println("Таблица студентов пуста: " + studtable.isEmpty());

    }

}

Задание №2:

package lab3;

public class TaskTwo {

    public static class Student {

        String firstName;

        String lastName;

        int age;

        double averageGrade;

        public Student(String firstName, String lastName, int age, double averageGrade) {

            this.firstName = firstName;

            this.lastName = lastName;

            this.age = age;

            this.averageGrade = averageGrade;

        }

        @Override

        public String toString() {

            return "Student: " + firstName + " " + lastName + ", age: " + age + ", averageGrade: " + averageGrade;

        }

    }

    public static void main(String[] args) {

        HashTable<Integer, Student> st = new HashTable<>();

        System.out.println("Создание студентов:");

        st.put(202501, new Student("Misha", "Golikov", 17, 4.8));

        st.put(202502, new Student("Stepan", "Aniskov", 18, 3.6));

        st.put(202503, new Student("Artem", "Vladimirovich", 23, 4.6));

        st.put(202504, new Student("Tanya", "Zorkina", 17, 3.2));

        st.put(202505, new Student("Vadim", "Aniskov", 19, 2.6));

        System.out.println();

        System.out.println("Количество студентов до изменений: " + st.size());

        System.out.println();

        System.out.println("Изменяем студента с номером зачётки: 202502: ");

        System.out.println("Старый студент: " + st.get(202502));

        st.put(202502, new Student("Anna", "Socolova", 23, 4.7));

        System.out.println("Новый студент: " + st.get(202502));

        System.out.println();

        System.out.println("Поиск студентов:");

        System.out.println("Студент по зачётке 202503: " + st.get(202503));

        System.out.println("Студент по зачётке 202501: " + st.get(202501));

        System.out.println("Студент по зачётке 202505: " + st.get(202505));

        System.out.println("Студент по зачётке 202506: " + st.get(202506));

        System.out.println();

        System.out.println("Удаление студентов:");

        System.out.println("Удаляем студентов с зачёткой 202502 и 202501:");

        System.out.println("Студенты до удаления:");

        System.out.println(st.get(202502));

        System.out.println(st.get(202501));

        st.remove(202502);

        st.remove(202501);

        System.out.println();

        System.out.println("После удаления:");

        System.out.println("Студент зачётка 202502: " + st.get(202502));

        System.out.println("Студент зачётка 202501: " + st.get(202501));

    }

}

# Вывод

В ходе выполнения данной лабораторной работы мы освоили работу с хэш-таблицами на Java. Изучили класс Object. Создали хэш-таблицы.