

|  |  |
| --- | --- |
| **Министерство образования и науки**  **Российской Федерации**  **Государственное образовательное учреждение высшего образования «МОСКОВСКИЙ АВТОМОБИЛЬНО-ДОРОЖНЫЙ**  **ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  **(МАДИ)»** |  |

**Кафедра «Высшая математика»**

**Отчет по дисциплине «Структуры и алгоритмы обработки данных»**

**Практическая работа №6  
«Работа со списками»**

**Выполнил:**

Учебная группа 1бПМ\_\_\_\_\_\_\_\_\_  
ФИО Греча К. П. \_

**Принял:**

Должность Старший преподаватель

Звание \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ФИО \_ Кутейников И. А.\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023г.

**Цель:**

1 Реализовать в Java класс в соответствии со своим вариантом.

Предусмотреть не менее 3 параметров, одним из которых является

объект другого класса, 2 методов и 2 конструкторов (включая

конструктор по умолчанию). Предусмотреть счетчик экземпляров

классов.

Предусмотреть для классов, являющихся полями не менее 2

параметров, 2 методов и 2 конструкторов (включая конструктор по

умолчанию).

2 Реализовать хранение объектов классов в виде списка.

Предусмотреть следующие операции над списком:

1) Добавления элемента (доп. нескольких элементов) в конец или в

определенное место списка;

2) Удаление элемента (доп. нескольких элементов) в конце или в

определенном месте списка;

3) Изменение определенного элемента списка;

4) Поиск и вывод номера и информации введенного элемента по

разным полям;

5) Вывод на экран всех объектов списка или определенного

элемента списка;

6) Вывод числа элементов в списке.  
  
Алгоритм:

import java.util.ArrayList;  
import java.util.Scanner;  
  
class Person {  
 String name;  
  
 public Person(String name) {  
 this.name = name;  
 }  
}  
  
class Clothing {  
 static int *count* = 0;  
 String size;  
 String color;  
 String brand;  
 String type;  
 Person owner;  
  
 public Clothing(String size, String color, String brand, String type, Person owner) {  
 this.size = size;  
 this.color = color;  
 this.brand = brand;  
 this.type = type;  
 this.owner = owner;  
 Clothing.*count*++;  
 }  
  
 public void isAvailable() {  
 System.*out*.println("Эта одежда доступна для покупки");  
 }  
  
 public void isNotAvailable() {  
 System.*out*.println("Эта одежда уже куплена другим человеком по имени: " + this.owner.name);  
 }  
  
 @Override  
 public String toString() {  
 return "Размер: " + size + ", Цвет: " + color + ", Брэнд: " + brand + ", Тип: " + type + ", Владелец: " + owner.name;  
 }  
}  
  
public class Main {  
 static ArrayList<Clothing> *clothingItems* = new ArrayList<>();  
  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 int variant;  
 while (true) {  
 System.*out*.println("\n1. Добавить одежду");  
 System.*out*.println("2. Удалить одежду");  
 System.*out*.println("3. Редактировать одежду");  
 System.*out*.println("4. Поиск одежды");  
 System.*out*.println("5. Отобразить параметры одежды");  
 System.*out*.println("6. Отобразить все предметы одежды");  
 System.*out*.println("7. Показать количество предметов одежды");  
 System.*out*.println("8. Перемещение");  
 System.*out*.println("0. Выйти");  
  
 System.*out*.print("Выберите пункт меню: ");  
 try {  
 variant = Integer.*parseInt*(in.nextLine());  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный выбор");  
 continue;  
 }  
 switch (variant) {  
 case 1:  
 // Добавление предметов одежды в список  
 System.*out*.print("Введите количество предметов одежды: ");  
 int count1;  
 try { //используется для определения блока кода, в котором может произойти исключение  
 count1 = Integer.*parseInt*(in.nextLine());  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 continue;  
 }  
  
 for (int i = 1; i <= count1; i++) {  
 // Ввод параметров для каждого предмета одежды  
 System.*out*.print("Введите размер: ");  
 String size = in.nextLine();  
 System.*out*.print("Введите цвет: ");  
 String color = in.nextLine();  
 System.*out*.print("Введите брэнд: ");  
 String brand = in.nextLine();  
 System.*out*.print("Введите тип: ");  
 String type = in.nextLine();  
 System.*out*.print("Введите имя владельца: ");  
 String ownerName = in.nextLine();  
 Person owner = new Person(ownerName);  
 // Создание нового экземпляра одежды и добавление его в список  
 Clothing clothing = new Clothing(size, color, brand, type, owner);  
 *clothingItems*.add(clothing);  
 System.*out*.println("Одежда добавлена в список");  
 }  
 break;  
  
 case 2:  
 // Удаление одежды из списка  
 if (!*clothingItems*.isEmpty()) {  
 System.*out*.print("Введите индекс удаляемой одежды: ");  
 try {  
 int index = Integer.*parseInt*(in.nextLine());  
 if (index >= 0 && index < *clothingItems*.size()) {  
 *clothingItems*.remove(index);  
 System.*out*.println("Одежда удалена из списка");  
 } else {  
 System.*out*.println("Неверный индекс");  
 }  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 }  
 } else {  
 System.*out*.println("Список одежды пуст");  
 }  
 break;  
  
 case 3:  
 // Редактирование параметров одежды  
 if (!*clothingItems*.isEmpty()) {  
 System.*out*.print("Введите индекс редактируемой одежды: ");  
 try {  
 int index = Integer.*parseInt*(in.nextLine());  
 if (index >= 0 && index < *clothingItems*.size()) {  
 Clothing clothing = *clothingItems*.get(index);  
 System.*out*.print("Введите новый размер: ");  
 clothing.size = in.nextLine();  
 System.*out*.print("Введите новый цвет: ");  
 clothing.color = in.nextLine();  
 System.*out*.print("Введите новый брэнд: ");  
 clothing.brand = in.nextLine();  
 System.*out*.print("Введите новый тип: ");  
 clothing.type = in.nextLine();  
 System.*out*.print("Введите новое имя владельца: ");  
 clothing.owner.name = in.nextLine();  
 System.*out*.println("Одежда обновлена");  
 } else {  
 System.*out*.println("Неверный индекс");  
 }  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 }  
 } else {  
 System.*out*.println("Список одежды пуст");  
 }  
 break;  
  
 case 4:  
 // Поиск одежды по различным параметрам  
 if (!*clothingItems*.isEmpty()) {  
 System.*out*.println("Выберите параметр для поиска:");  
 System.*out*.println("1. По цвету");  
 System.*out*.println("2. По индексу");  
 System.*out*.println("3. По типу");  
 int searchOption;  
 try {  
 searchOption = Integer.*parseInt*(in.nextLine());  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 continue;  
 }  
 switch (searchOption) {  
 case 1:  
 // Поиск по цвету  
 System.*out*.print("Введите цвет для поиска: ");  
 String colorToSearch = in.nextLine();  
 for (int index = 0; index < *clothingItems*.size(); index++) {  
 Clothing clothing = *clothingItems*.get(index);  
 if (colorToSearch.equalsIgnoreCase(clothing.color)) {  
 System.*out*.println("Индекс: " + index + ", Одежда: " + clothing);  
 }  
 }  
 break;  
  
 case 2:  
 // Поиск по индексу  
 System.*out*.print("Введите индекс для поиска: ");  
 try {  
 int indexToSearch = Integer.*parseInt*(in.nextLine());  
 if (indexToSearch >= 0 && indexToSearch < *clothingItems*.size()) {  
 Clothing clothing = *clothingItems*.get(indexToSearch);  
 System.*out*.println("Одежда: " + clothing);  
 } else {  
 System.*out*.println("Неверный индекс");  
 }  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 }  
 break;  
  
 case 3:  
 // Поиск по типу  
 System.*out*.print("Введите тип для поиска: ");  
 String typeToSearch = in.nextLine();  
 for (int index = 0; index < *clothingItems*.size(); index++) {  
 Clothing clothing = *clothingItems*.get(index);  
 if (typeToSearch.equalsIgnoreCase(clothing.type)) {  
 System.*out*.println("Индекс: " + index + ", Одежда: " + clothing);  
 }  
 }  
 break;  
  
 default:  
 System.*out*.println("Неверный выбор параметра для поиска");  
 }  
 } else {  
 System.*out*.println("Список одежды пуст");  
 }  
 break;  
  
 case 5:  
 // Вывод параметров конкретной одежды  
 if (!*clothingItems*.isEmpty()) {  
 System.*out*.print("Введите индекс одежды для отображения: ");  
 try {  
 int index = Integer.*parseInt*(in.nextLine());  
 if (index >= 0 && index < *clothingItems*.size()) {  
 Clothing clothing = *clothingItems*.get(index);  
 System.*out*.println("Параметры одежды: " + clothing);  
 } else {  
 System.*out*.println("Неверный индекс");  
 }  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 }  
 } else {  
 System.*out*.println("Список одежды пуст");  
 }  
 break;  
 case 6:  
 // Вывод списка одежды с указанием индекса каждого элемента  
 if (!*clothingItems*.isEmpty()) {  
 for (int index = 0; index < *clothingItems*.size(); index++) {  
 System.*out*.println("Индекс: " + index + ", Одежда: " + *clothingItems*.get(index));  
 }  
 } else {  
 System.*out*.println("Список одежды пуст");  
 }  
 break;  
  
 case 7:  
 // Вывод количества предметов одежды в списке  
 System.*out*.println("Количество предметов одежды в списке: " + *clothingItems*.size());  
 break;  
  
 case 8:  
 // Перемещение одежды в списке  
 if (!*clothingItems*.isEmpty()) {  
 System.*out*.print("Введите индекс перемещаемой одежды: ");  
 try {  
 // Ввод индекса элемента для перемещения  
 int index = Integer.*parseInt*(in.nextLine());  
  
 // Проверка на корректность введенного индекса  
 if (index >= 0 && index < *clothingItems*.size()) {  
 // Удаление одежды из текущей позиции  
 Clothing clothing = *clothingItems*.remove(index);  
  
 // Ввод количества позиций для перемещения  
 System.*out*.print("Введите количество позиций для перемещения: ");  
 int shiftAmount = Integer.*parseInt*(in.nextLine());  
  
 // Вычисление новой позиции после перемещения  
 int newIndex = index + shiftAmount;  
  
 // Добавление одежды в новую позицию  
 *clothingItems*.add(Math.*max*(0, Math.*min*(newIndex, *clothingItems*.size())), clothing);  
  
 // Вывод измененного расположения одежды  
 System.*out*.println("Одежда перемещена. Измененное расположение:");  
 for (Clothing item : *clothingItems*) {  
 System.*out*.println(item);  
 }  
 } else {  
 System.*out*.println("Неверный индекс");  
 }  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный ввод");  
 }  
 } else {  
 System.*out*.println("Список одежды пуст");  
 }  
 break;  
 }  
 }  
 }

}

**Результат:**

1. Добавить одежду

2. Удалить одежду

3. Редактировать одежду

4. Поиск одежды

5. Отобразить параметры одежды

6. Отобразить все предметы одежды

7. Показать количество предметов одежды

8. Перемещение

0. Выйти

Выберите пункт меню: 1

Введите количество предметов одежды: **2**

Введите размер: **45**

Введите цвет: **черный**

Введите брэнд: **адидас**

Введите тип: **кроссовки**

Введите имя владельца: **Ваня**

**Одежда добавлена в список**

Введите размер: **41**

Введите цвет: **белый**

Введите брэнд: **найк**

Введите тип: **кроссовки**

Введите имя владельца: **Антон**

**Одежда добавлена в список**

1. Добавить одежду

2. Удалить одежду

3. Редактировать одежду

4. Поиск одежды

5. Отобразить параметры одежды

6. Отобразить все предметы одежды

7. Показать количество предметов одежды

8. Перемещение

0. Выйти

Выберите пункт меню: 6

**Индекс: 0, Одежда: Размер: 45, Цвет: черный, Брэнд: адидас, Тип: кроссовки, Владелец: Ваня**

**Индекс: 1, Одежда: Размер: 41, Цвет: белый, Брэнд: найк, Тип: кроссовки, Владелец: Антон**

Выберите пункт меню: **7**

Количество предметов одежды в списке: **2**

Выберите пункт меню: **4**

Выберите параметр для поиска:

1. По цвету

2. По индексу

3. По типу

**1**

Введите цвет для поиска: **белый**

**Индекс: 1, Одежда: Размер: 41, Цвет: белый, Брэнд: найк, Тип: кроссовки, Владелец: Антон**

Выберите пункт меню: **3**

Введите индекс редактируемой одежды: **1**

Введите новый размер: **43**

Введите новый цвет: белый

Введите новый брэнд: найк

Введите новый тип: кроссовки

Введите новое имя владельца: Антон

**Одежда обновлена**

Выберите пункт меню: **5**

Введите индекс одежды для отображения: **1**

Параметры одежды: **Размер: 43, Цвет: белый, Брэнд: найк, Тип: кроссовки, Владелец: Антон**

Выберите пункт меню: **2**

Введите индекс удаляемой одежды: **0**

**Одежда удалена из списка**

Выберите пункт меню: **5**

Введите индекс одежды для отображения: **0**

Параметры одежды: **Размер: 43, Цвет: белый, Брэнд: найк, Тип: кроссовки, Владелец: Антон**

**Заключение:**

В процессе выполнения лабораторной работы мы познакомились с реализацией классов, а также с реализацией объектов класса и методов, конструкторов. В лабораторной работе использовались навыки создания полей классов, а так же добавление элемента класса, удаление элемента, поиск и вывод информации, изменение определенного элемента.