**МОЛДАВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет Математики и Информатики**

**Департамент Информатики**

Лабораторная работа № 1

Программирование на JAVA

Наследование. Иерархии классов.

Проверил: профессор, др. Gheorghe Latu

Выполнил: Mamaliga Artur grupa I2302-1

Кишинев, 2024

## Цель работы:

Создать иерархию классов для представления геометрических фигур, включая класс для прямоугольника и параллелепипеда. Реализовать методы для вычисления площади прямоугольника, объема и веса параллелепипеда, а также провести сравнение двух объектов по объему и весу.

## Задание:

1. Создать классы Сторона, Прямоугольник и Параллелепипед.  
2. Класс Прямоугольник должен содержать метод для вычисления площади.  
3. Класс Параллелепипед должен содержать методы для вычисления объема и веса.  
4. Классы Прямоугольник и Параллелепипед должны иметь конструкторы.  
5. Создать класс MAIN, в котором создаются два параллелепипеда, и определяется, какой из них больший по объему и легче по весу. Вывести на экран все характеристики создаваемых объектов.

## Ход работы:

### 1. Создание классов

Класс Side описывает сторону фигуры и содержит одно поле - длину стороны (length), а также методы доступа и изменения значения длины.  
Класс Rectangle наследуется от класса Side и добавляет новое поле width, представляющее ширину прямоугольника. Этот класс содержит конструктор для инициализации длины и ширины прямоугольника, а также метод calcArea(), который вычисляет площадь прямоугольника.  
Класс Parallelepiped наследуется от класса Rectangle и добавляет новое поле height, представляющее высоту параллелепипеда, а также поле density, которое описывает плотность материала.

### 2. Реализация класса MAIN

В классе Program создаются два объекта класса Parallelepiped с различными характеристиками. После этого выводятся их параметры, и выполняется сравнение по объему и весу.

### 3. Результаты выполнения программы

Характеристики первого параллелепипеда:  
Side [length=4] Rectangle [width=4, area=16] Parallelepiped [height=6, density=2.5, volume=96, weight=240.0]  
  
Характеристики второго параллелепипеда:  
Side [length=3] Rectangle [width=6, area=18] Parallelepiped [height=7, density=3.0, volume=126, weight=378.0]  
  
Сравнение параллелепипедов:  
- Параллелепипед 2 имеет больший объем.  
- Параллелепипед 1 легче.

## 4. Исходный код программы

### Класс Program.java:

public class Program {  
 public static void main(String[] args) {  
 Form form = new Form();  
  
 Form.Parallelepiped p1 = form.new Parallelepiped(4, 4, 6, 2.5);  
 Form.Parallelepiped p2 = form.new Parallelepiped(3, 6, 7, 3.0);  
  
 System.out.println("[ + ] Parallelepiped 1 characteristics:");  
 System.out.println(p1);  
  
 System.out.println("\n[ + ] Parallelepiped 2 characteristics:");  
 System.out.println(p2);  
  
 form.compareParallelepipeds(p1,p2);  
 }  
}

### Класс Form.java:

public class Form {  
  
 public class Side {  
 public int length;  
  
 public int getLen() {  
 return length;  
 }  
  
 public void setLen(int length) {  
 this.length = length;  
 }  
  
 @Override  
 public String toString() {  
 return "Side [length=" + length + "]";  
 }  
 }  
  
 public class Rectangle extends Side {  
  
 public int width;  
  
 public Rectangle(int length, int width) {  
 setLen(length);  
 this.width = width;  
 }  
  
 public int calcArea() {  
 return getLen() \* width;  
 }  
  
 @Override  
 public String toString() {  
 return super.toString() + "Rectangle [width=" + width + ", area=" + calcArea() + "] ";  
 }  
 }  
  
 public class Parallelepiped extends Rectangle {  
  
 public int height;  
 public double density;  
  
 public Parallelepiped(int length, int width, int height, double density) {  
 super(length, width);  
 this.height = height;  
 this.density = density;  
 }  
  
 public int calcVolume() {  
 return calcArea() \* height;  
 }  
  
 public double calcWeight() {  
 return calcVolume() \* density;  
 }  
  
 @Override  
 public String toString() {  
 return super.toString() + "Parallelepiped [height=" + height + ", density=" + density + ", volume=" + calcVolume() + ", weight=" + calcWeight() + "] ";  
 }  
 }  
  
 public void compareParallelepipeds(Parallelepiped p1, Parallelepiped p2) {  
 if (p1.calcVolume() > p2.calcVolume()) {  
 System.out.println("\n[ + ] Parallelepiped 1 is larger.");  
 } else if (p1.calcVolume() < p2.calcVolume()) {  
 System.out.println("\n[ + ] Parallelepiped 2 is larger.");  
 } else {  
 System.out.println("\n[ + ] Both parallelepipeds have the same volume.");  
 }  
  
 if (p1.calcWeight() < p2.calcWeight()) {  
 System.out.println("[ + ] Parallelepiped 1 is lighter.\n");  
 } else if (p1.calcWeight() > p2.calcWeight()) {  
 System.out.println("[ + ] Parallelepiped 2 is lighter.\n");  
 } else {  
 System.out.println("[ + ] Both parallelepipeds have the same weight.\n");  
 }  
 }  
}

## 5. Заключение:

В ходе выполнения лабораторной работы была создана иерархия классов для работы с прямоугольником и параллелепипедом. Были реализованы методы для расчета площади, объема и веса объектов. Программа успешно выводит характеристики двух параллелепипедов и корректно сравнивает их по объему и весу.