Лабораторна робота 2.

Основи мови програмування Kotlin.

**Завдання**

Реалізуйте батьківський клас Геометрична фігура з методами для визначення площі та периметра. Створіть підкласи для різних типів геометричних фігур (квадрат, коло, трикутник), які успадковують методи батьківського класу та реалізують власні алгоритми обчислення площі та периметра.

**Хід роботи**

Для створення програми, що обчислює площу та периметр різних геометричних фігур, ми починаємо з визначення батьківського класу GeometricFigure. Цей клас містить два методи: area() та perimeter(), які повертають значення за замовчуванням (0.0). Цей підхід дозволяє визначити базовий інтерфейс для всіх геометричних фігур, що будуть створені у майбутньому.

Далі ми створюємо підклас для квадрата, який називається Square. Цей клас успадковує GeometricFigure та має параметр sideLength, який зберігає довжину сторони квадрата. У класі Square ми перевизначаємо методи area() та perimeter(), щоб вони обчислювали площу та периметр квадрата відповідно. Площа квадрата обчислюється як квадрат довжини його сторони, а периметр - як чотириразове значення довжини сторони.

Наступним кроком є створення підкласу для кола, який називається Circle. Цей клас також успадковує GeometricFigure та має параметр radius, який зберігає радіус кола. У класі Circle ми перевизначаємо методи area() та perimeter(). Площа кола обчислюється за формулою πr², де r - радіус кола, а периметр (або довжина кола) обчислюється як 2πr.

Підклас для трикутника називається Triangle. Цей клас має три параметри: sideA, sideB та sideC, які зберігають довжини сторін трикутника. Метод perimeter() у цьому класі обчислює периметр трикутника як суму довжин його сторін. Метод area() використовує формулу Герона для обчислення площі трикутника. Формула Герона включає обчислення напівпериметра s = (sideA + sideB + sideC) / 2, після чого площа обчислюється як √(s(s - sideA)(s - sideB)(s - sideC)).

Основна програма забезпечує інтерфейс взаємодії з користувачем. Вона використовує клас Scanner для зчитування вводу користувача. У циклі while програма пропонує користувачу вибрати одну з опцій: обчислити площу та периметр квадрата, кола або трикутника, або завершити програму. В залежності від вибору користувача, програма запитує необхідні дані (довжину сторони для квадрата, радіус для кола, довжини сторін для трикутника), створює відповідний об'єкт фігури та виводить обчислені площу і периметр. Якщо користувач обирає опцію завершення, програма припиняє свою роботу.

Повний код програми:

import java.util.Scanner

import kotlin.math.PI

import kotlin.math.sqrt

// Батьківський клас

open class GeometricFigure {

open fun area(): Double {

return 0.0

}

open fun perimeter(): Double {

return 0.0

}

}

// Підклас для квадрата

class Square(private val sideLength: Double) : GeometricFigure() {

override fun area(): Double {

return sideLength \* sideLength

}

override fun perimeter(): Double {

return 4 \* sideLength

}

}

// Підклас для кола

class Circle(private val radius: Double) : GeometricFigure() {

override fun area(): Double {

return PI \* radius \* radius

}

override fun perimeter(): Double {

return 2 \* PI \* radius

}

}

// Підклас для трикутника

class Triangle(private val sideA: Double, private val sideB: Double, private val sideC: Double) : GeometricFigure() {

override fun area(): Double {

// Використовуємо формулу Герона для обчислення площі трикутника

val s = perimeter() / 2

return sqrt(s \* (s - sideA) \* (s - sideB) \* (s - sideC))

}

override fun perimeter(): Double {

return sideA + sideB + sideC

}

}

fun main() {

val scanner = Scanner(System.`in`)

var continueProgram = true

while (continueProgram) {

println("Виберіть опцію:")

println("1. Обчислити площу та периметр квадрата")

println("2. Обчислити площу та периметр кола")

println("3. Обчислити площу та периметр трикутника")

println("4. Вийти з програми")

// Зчитування вибору користувача

val choice = scanner.nextInt()

when (choice) {

1 -> {

println("Введіть довжину сторони квадрата:")

val sideLength = scanner.nextDouble()

val square = Square(sideLength)

println("Площа квадрата: ${square.area()}")

println("Периметр квадрата: ${square.perimeter()}")

}

2 -> {

println("Введіть радіус кола:")

val radius = scanner.nextDouble()

val circle = Circle(radius)

println("Площа кола: ${circle.area()}")

println("Довжина кола: ${circle.perimeter()}")

}

3 -> {

println("Введіть довжини сторін трикутника (через пробіл):")

val sideA = scanner.nextDouble()

val sideB = scanner.nextDouble()

val sideC = scanner.nextDouble()

val triangle = Triangle(sideA, sideB, sideC)

println("Площа трикутника: ${triangle.area()}")

println("Периметр трикутника: ${triangle.perimeter()}")

}

4 -> {

println("Програма завершує роботу.")

continueProgram = false

}

else -> {

println("Невірний вибір опції. Будь ласка, спробуйте ще раз.")

}

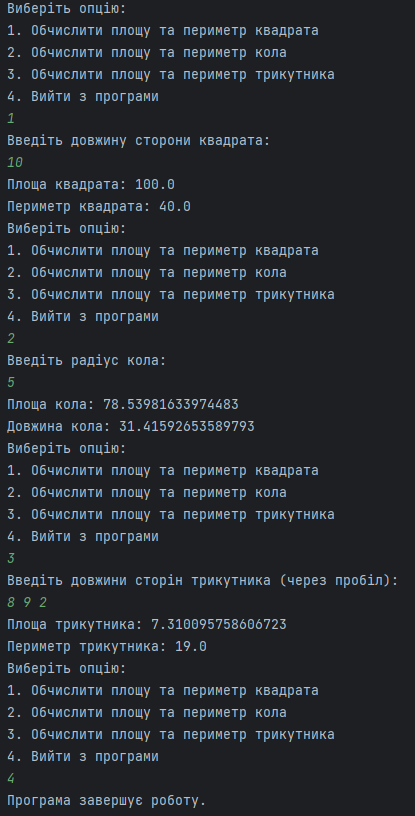
}

}

scanner.close()

}

**Результати роботи**

****

Тестування програми