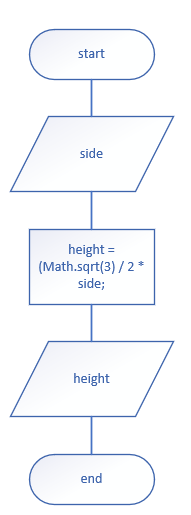
**Лабораторная работа №6**

**Основы программирования на Java: модификаторы доступа и класса**

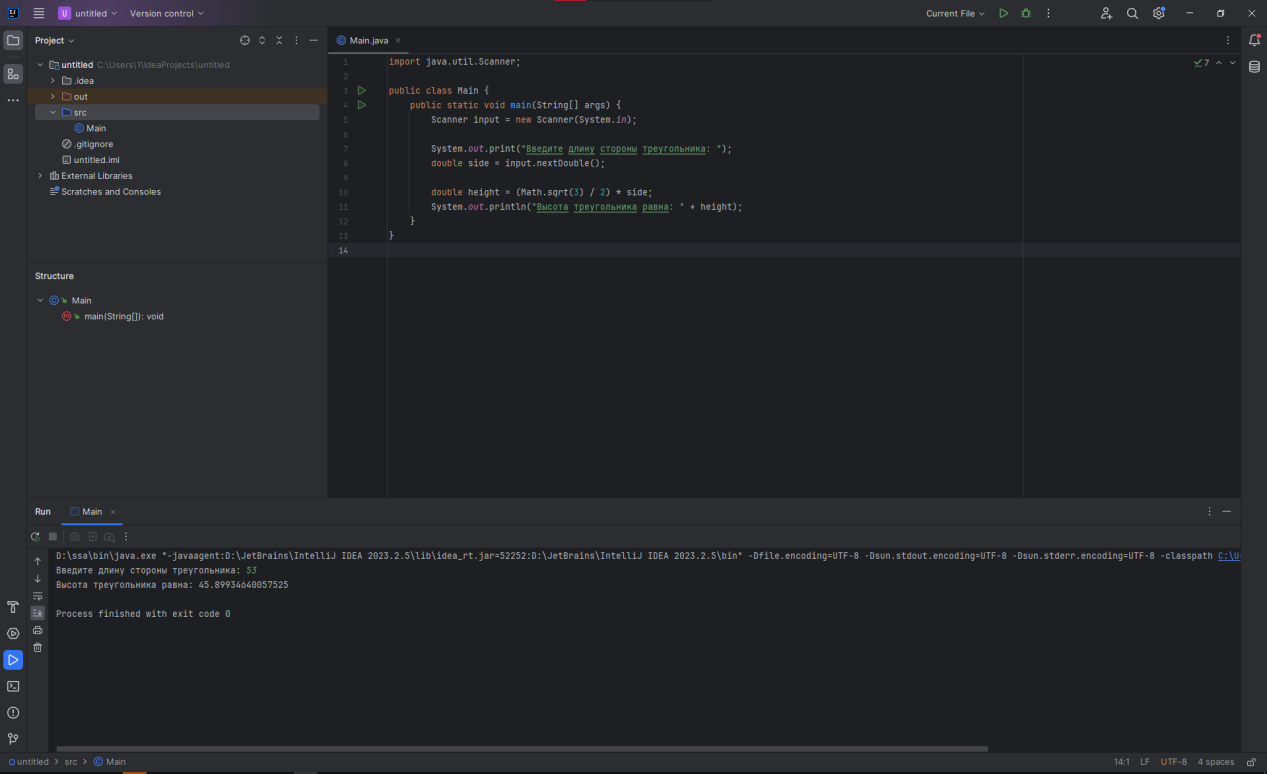
**Цель работы:** освоить основные способы создания Java-программ либо с помощью обычного редактора, либо с помощью среды разработки. Приобрести навыки работы с модификаторами доступа в Java.

Вариант 13



|  |  |  |
| --- | --- | --- |
| Наименование переменной | Тип данных | Назначение |
| side | double | Переменная, вводимая пользователем |
| height | double | Содержит результат |

import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.in);  
  
 System.out.print("Введите длину стороны треугольника: ");  
 double side = input.nextDouble();  
  
 double height = (Math.sqrt(3) / 2) \* side;  
 System.out.println("Высота треугольника равна: " + height);  
 }  
}



1. Модификатор в программировании - это ключевое слово, которое определяет видимость и доступность элемента программы (например, переменной, метода, класса) для других элементов программы.

2. В Java существуют следующие модификаторы:

- public: элемент доступен из любого места программы.

Пример: public class MyClass { ... }

- private: элемент доступен только внутри класса, в котором он объявлен.

Пример: private int x;

- protected: элемент доступен внутри класса, в котором он объявлен, а также внутри подклассов этого класса и других классов в том же пакете.

Пример: protected void myMethod() { ... }

- default (отсутствие модификатора): элемент доступен только внутри класса и внутри других классов в том же пакете.

Пример: int y;

3. Модификатор доступа private, public, protected определяют уровень доступа к элементам класса:

- private: элемент доступен только внутри класса, в котором он объявлен.

- public: элемент доступен из любого места программы.

- protected: элемент доступен внутри класса, в котором он объявлен, а также внутри подклассов этого класса и других классов в том же пакете.

4. Вспомогательные модификаторы в Java могут использоваться вместе с основными модификаторами и изменять их поведение. Примеры вспомогательных модификаторов:

- static: элемент принадлежит классу, а не экземпляру класса.

Пример: public static int count;

- final: элемент является неизменяемым и его нельзя переопределить или наследовать.

Пример: public final String name;

5. В моей работе я использовал модификаторы public, static