**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ**

**КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Лабораторная работа № 1

Выполнила: Студентка группы БВТ2306

Максимова Мария

Москва 2024

Ссылка на Git-hub: <https://github.com/Mashaaaaa7/ITiP-1-lab> файл read.me

Для выполнения лабораторной работы №1 нам понадобится выполнить 2 задания:

Задание 1. Создайте программу, которая находит и выводит все простые числа меньше 100.

Задание 2. Создайте программу, которая определяет, является ли введенная строка палиндромом.

Для начала, запустим приложение IntellIJ Idea и создадим проект. Назовём его JavaHelloWordProgram.java и напишем код для выполнения команды «Hello, word!» и сохраним программу под именем JavaHelloWorldProgram.java в каталоге для первой лабораторной работы.(См. рисунок 1)

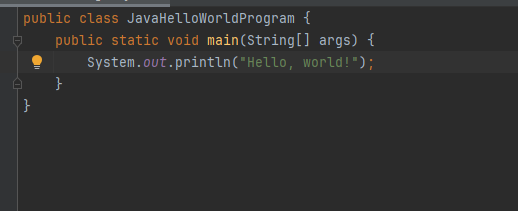


Рисунок 1

Далее, запускаю код:

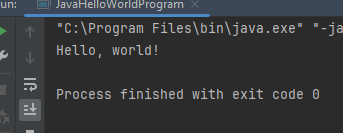


Рисунок 2

Открою командную строку PowerShell и перейду в каталог, в котором сохранен файл программы. Для начала, мы можем набрать следующие команды чтобы скомпилировать и запустить программу:

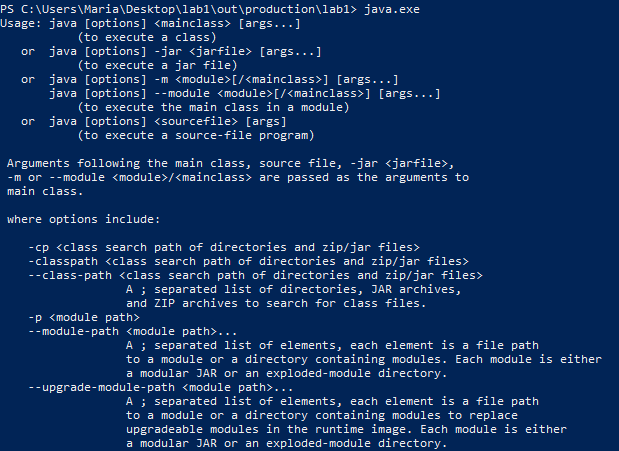


Рисунок 3

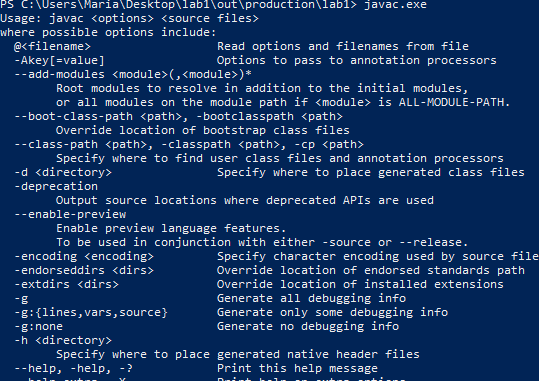


Рисунок 4

В первом случае будет вызван компилятор javac.exe, а во втором случае – запускалка java.exe, которая стартует нашу программу. Эти файлы лежат в папке bin нашего JDK

Скомпилируем код:

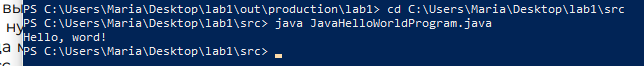


Рисунок 5



Рисунок 6

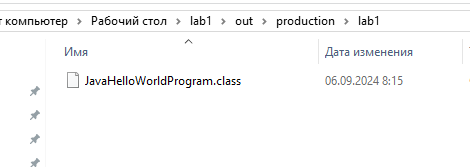


Рисунок 7

Эта команда вызовет компилятор, который создаст файл **JavaHelloWorldProgram.class**, содержащий скомпилированный код нашей java программы.

Чтобы запустить ее, нужно ввести команду **java** с именем класса в качестве параметра:



Рисунок 8

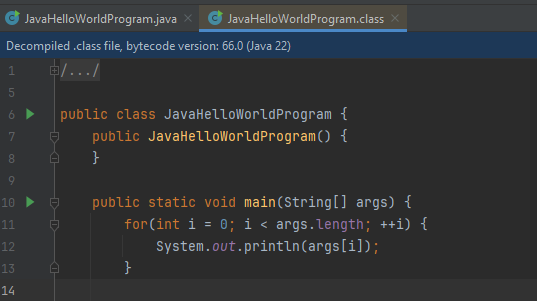
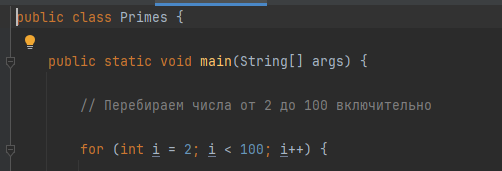


Рисунок 9

Далее, приступим к выполнению заданий. Начнём с задания 1. Нам нужно создать программу, которая находит и выводит все простые числа меньше 100. До 100 встречается 25 простых чисел:

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97.

Начнём с понятия простого числа. **Простое число — это натуральное число больше 1, у которого есть всего два делителя: единица и само число. Таким образом, нам нужно перебирать простые числа от 2 до 100 включительно. Создадим файл с именем Primes.java и опишем класс:**

****

**Рисунок 10**

Разберём подробней:

1. **public class Primes**:

* Здесь мы объявляем класс с именем Primes. В Java все кодовые структуры организованы в классы, и каждая программа должна содержать хотя бы один класс. Класс является шаблоном или чертежом для создания объектов.

1. **public static void main(String[] args)**:

* Это объявление метода main, который является точкой входа для любой Java-программы.
* **public**: Это модификатор доступа, который указывает, что метод main доступен из любого другого класса.
* **static**: Этот модификатор указывает, что метод принадлежит классу Primes, а не объекту класса. Это означает, что метод может быть вызван без создания экземпляра класса.
* **void**: Это возвращаемый тип метода, указывающий, что метод не возвращает никакого значения.
* **String[] args**: Это параметр метода, представляющий массив строк. Он используется для передачи аргументов командной строки в программу. Когда вы запускаете программу, вы можете передать в нее параметры через командную строку, и они будут доступны в этом массиве.

1. **for (int i = 2; i < 100; i++) { :**

объявить переменную i целого типа (int) и присвоить ей начальное значение 2;

пока i меньше 100, выполнять код в фигурных скобках {...} после оператора for;

увеличивать значение переменной i на единицу после каждого выполнения кода в фигурных скобках.

1. **if (isPrime(i)) {** // Проверяем, является ли число простым

**System.out.println(i);** // Выводим простое число

После запуска данной программы, на выходе будет представлены 25 простых чисел от 2 до 100. Вот часть из них:

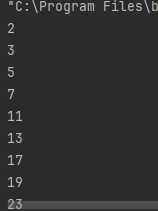


Рисунок 11

Для второго задания надо сделать код, который скажет, палиндром слово или нет

Пишем код и запускаем программу:

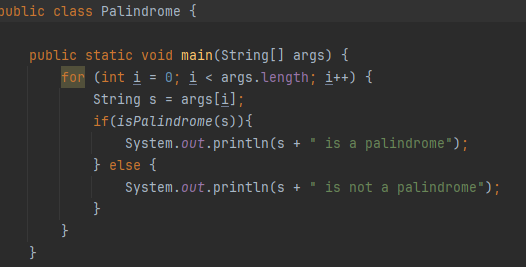


Рисунок 12

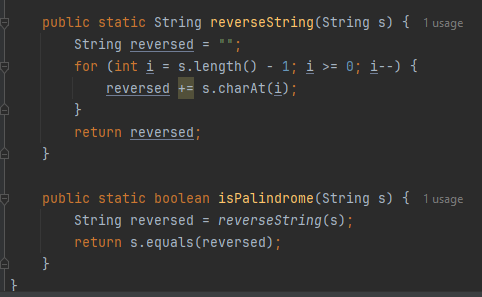


Рисунок 13

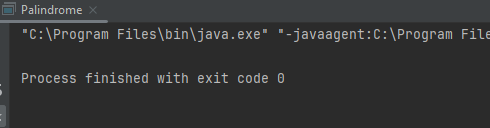


Рисунок 14

Разберём код подробно:

1. public class Palindrome — это класс с именем Palindrome, который является публичным (доступным для использования другими классами).

2. Метод main — это точка входа в программу. Метод принимает массив строк args и выполняет цикл for для каждой строки в этом массиве.

3. Внутри цикла for:

String s = args[i] — строка s присваивается значению из массива args по индексу i.

isPalindrome(s) — вызывается метод isPalindrome со строкой s в качестве аргумента. Этот метод возвращает true, если строка является палиндромом, и false в противном случае.

Если строка является палиндромом, то выводится сообщение «s is a palindrome».

Иначе выводится сообщение «s is not a palindrome».

4. Метод reverseString — этот метод принимает строку s и возвращает её в обратном порядке. Он использует цикл for и добавляет символы строки s в обратном порядке к строке reversed.

5. Метод isPalindrome — этот статический метод проверяет, является ли строка s палиндромом. Для этого он вызывает метод reverseString, чтобы получить обратную строку, а затем сравнивает исходную строку с обратной. Если они равны, то строка является палиндромом. В противном случае она не является палиндромом.

Этот код позволяет проверить, является ли каждая строка в массиве args палиндромом или нет.

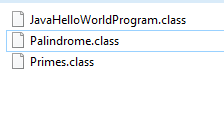


Рисунок 15

Для окончания лабораторной работы, нам понадобится программа PowerShell. Мы компилируем и тестируем программу, как указано в задании. На выходе, у нас выводятся наши слова и определяется, палиндром слово или нет.

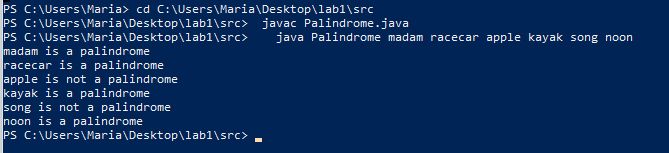


Рисунок 16