Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Ордена Трудового Красного знамени

Федеральное государственное бюджетное учреждение

высшего образования

«Московский технический университет связи и информатики»

Кафедра МКиТ

Лабораторная работа № 1

«Основы синтаксиса Java»

Выполнил студент

Группы БСТ1903

Бритвин Д.В.

Вариант № 3

Москва 2021

**Оглавление**

[**1.** **Цель работы** 3](#_Toc72363856)

[**2.** **Задание на лабораторную работу** 3](#_Toc72363857)

[**3.** **Код программы** 3](#_Toc72363858)

[**4.** **Результаты выполнения программы** 5](#_Toc72363859)

[**Вывод:** 6](#_Toc72363860)

# **Цель работы**

Целью лабораторной работы является изучение основ синтаксиса Java с помощью нескольких простых задач программирования, а также изучение работы компилятора Java.

# **Задание на лабораторную работу**

1. «Простые числа» Создайте программу, которая находит и выводит все простые числа меньше 100.
2. «Палиндромы» Вторая программа, которую вам необходимо будет написать, показывает, является ли строка палиндромом.

# **Код программы**

Ниже представлен код для первой программы:

public class Palindrome {

public static void main(String[] args) {

String s = "";

for (int i = 0; i < args.length; i++) {

s = args[i];

boolean b = isPalindrome(s); //Вызов функции isPalindrome

if (b == true)

System.out.println("Is Palindrome -> " + s);//вывод результата если слово палиндром

else

System.out.println("Is not Palindrome -> " + s);//вывод результата если слово не палиндром

}

}

public static String reverseString(String s){

int len = s.length();

String str = ""; //Функция создает новую строку

while (len > 0){

str = str + s.charAt(len - 1);//начиная с конца

len--;//записывает в нее поочередно все элементы входящей строки

}

return str;

}

public static boolean isPalindrome(String s){

String str = reverseString(s);//Вызываем функцию, переворачивающую нашу строку

if (s.equals(str))//Проверяем, является ли она равной нашей изначальной строке

return true;

else

return false;

}

}

Для второй программы:

public class Primes {

public static void main(String[] args) { //основная функция

System.out.println("Prime counts from 2 to 100:");

for (int i = 2; i < 100; i++){ //Начало цикла (от 2 до 100 (100 не включительно тк оно не простое))

if (isPrime(i) == true) //Если число простое - выводим его

System.out.println(i);

}

}

public static boolean isPrime(int n){

int count = 0; //Это количество чисел, на которые делится проверяемое нами число

for(int j = 2; j <=n && count < 2;++j){ //Начало цикла

if(n % j == 0){ //проверка условия

++count; //увеличение нашего счетчика если число делится на какое-либо

}

}

if(count<2) //Наш счетчик может быть равен только 1, тк число должно делиться только на себя

return true; //выводим true если число простое

else

return false; //выводим false если число не простое

}

}

# **Результаты выполнения программы**

Результаты работы программы Palindrome представлены на рисунке 1:

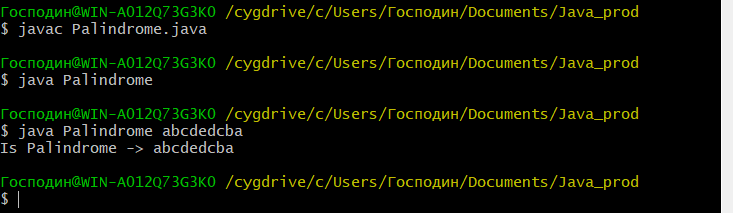
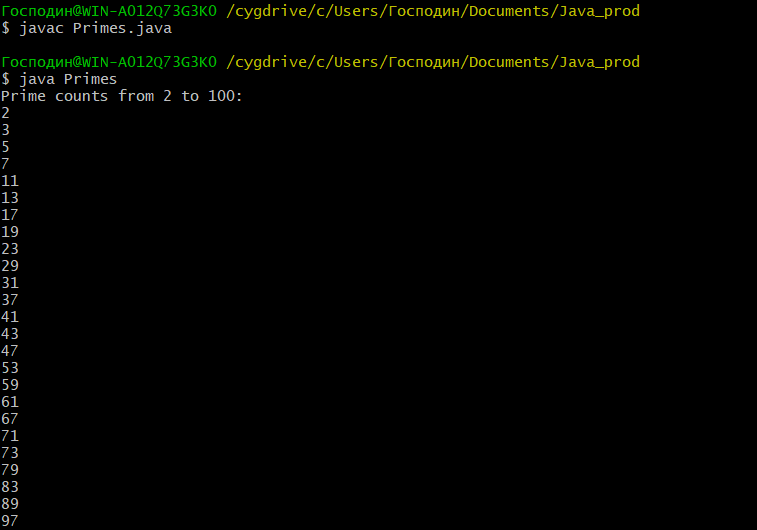


Рисунок 1 – Результат работы программы Palindrome

Результаты работы программы Primes представлены на рисунке 2:

  
Рисунок 2 – результаты работы программы Primes

# **Вывод:**

В ходе проделанной работы были выполнены задачи на нахождение всех простых чисел до 100 и проверки является ли слово палиндромом, также были изучены принципы работы компилятора Java