

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ**

Ордена Трудового Красного Знамени

**Федеральное государственное бюджетное образовательное учреждение
высшего образования**

«Московский технический университет связи и информатики»

Математическая кибернетика и информационные технологии

Отчёт по лабораторной работе №3

по дисциплине «ИТиП» на тему:

«Основы синтаксиса Java»

Выполнили: студенты группы

БВТ2104

Старостин Игорь Дмитриевич

Руководитель:

Мосева Марина Сергеевна

Москва 2022

1. Цель работы: изучение основ синтаксиса Java, изучение основ использования компилятора Java и виртуальной машины Java.

2. Задание:

1. Создать файл Primes.java и
 - a) Описать класс Primes.
 - b) Написать функцию isPrime класса Primes.
 - c) Произвести перебор чисел от 2 до 100 функцией isPrime.
 - d) Запустить программу.
2. Создать файл Palindrome.java и
 - a) Написать функцию реверса строки класса Palindrome.
 - b) Написать функцию isPalindrome класса Palindrome.
 - c) Произвести перебор аргументов командной строки функцией isPalindrome.
 - d) Запустить программу.

3.Ход работы:

- 1.a) Создаём файл Primes и описываем класс Primes(См. Рис. 1)

```
public class Primes{  
    Barsux *  
    public static void main(String[] args){  
  
    }  
}
```

Рисунок 1 - Класс Primes

В функции присутствует цикл, который передаёт все числа в функцию isPrime в диапазоне от 2 до 100.

- b) Напишем функцию isPrime(См. Рис. 2)

```

public static boolean isPrime(int n){
    //Если число меньше 0 или 0 или 1 то оно будет точно непростым.
    if(n < 2)return false;
    //Нет смысла перебирать все числа в диапазоне от 1 до n, достаточно от 2 до n // 2 + n % 2
    for(int i = 2; i < (n / 2 + n % 2); i++) {
        if(n % i == 0)return false;
    }
    return true;
}

```

Рисунок 2 - Функция isPrime

Функция перебирает числа от 2 до $n \div 2 + n \bmod 2$, если она находит делитель, то это число уже не будет являться простым.

с) Переберём числа в ф-ции main(См. Рис. 3)

```

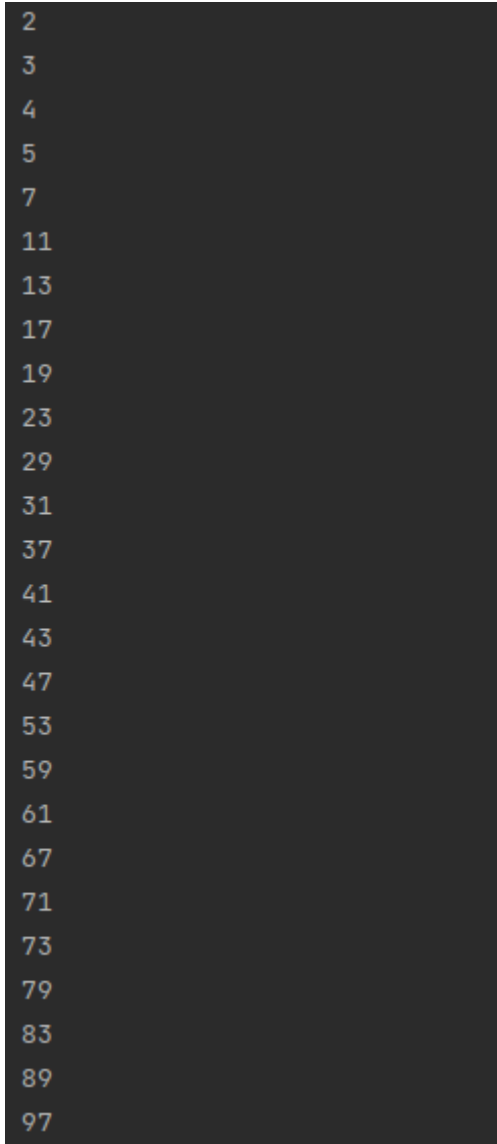
public class Primes{
    Barsux
    public static void main(String[] args){
        for(int i = 2; i <= 100; i++){
            if(isPrime(i))System.out.println(i);
        }
    }
}

```

Рисунок 3 - Перебор чисел в ф-ции main

Допишем все комментарии и приступим к запуску программы.

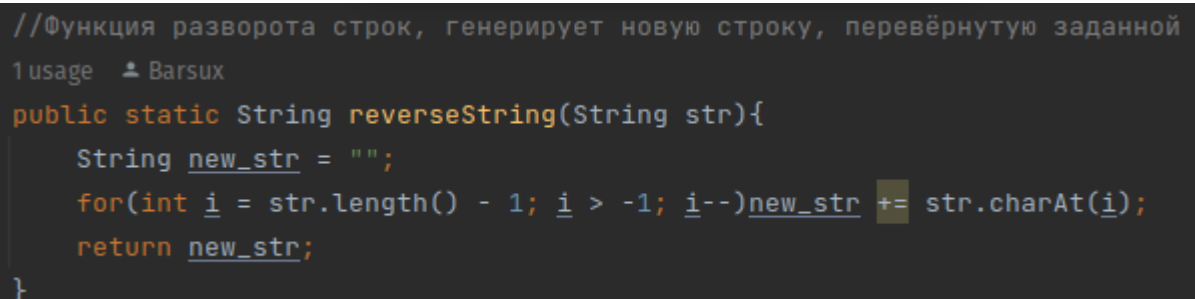
d) Компилируем и запускаем программу(См. Рис. 4)



2
3
4
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97

Рисунок 4 - Результат работы программы 1.

2.a) Напишем функцию stringReverse класса Palindrome(См. Рис. 5)



```
//Функция разворота строк, генерирует новую строку, перевёрнутую заданной;
1 usage  @ Barsux;
public static String reverseString(String str){
    String new_str = "";
    for(int i = str.length() - 1; i > -1; i--) new_str += str.charAt(i);
    return new_str;
}
```

Рисунок 5 - Функция reverseString

Функция в цикле перебирает входную строку с конца, добавляя символы в новую.

б) Напишем функцию isPalindrome(См. Рис. 6)

```
//Получился декоратор, функция определяет является ли строка(аргумент) палиндромом, сравнивая исходную и перевёрнутую строки
//usage  ⚡ Barsux
public static boolean isPalindrome(String str){
    return str.equals(reverseString(str));
}
```

Рисунок 6 - Функция isPalindrome

Функция - декоратор, проверяющая является ли строка палиндромом.

с) В функции main произведём перебор аргументов командной строки(См. Рис. 7)

```
//Точка входа в программу, перебирает аргументы командной строки, проверяя является ли палиндромом каждый из них.
//Barsux
public static void main(String[] args) {
    for (int i = 0; i < args.length; i++) {
        String s = args[i];
        System.out.println(s + " isPalindrome: " + isPalindrome(s));
    }
}
```

Рисунок 7 - Перебор аргументов командной строки

д) Произведём запуск программы(См. Рис. 8)

```
java isPalindrome: false
Palindrome isPalindrome: false
madam isPalindrome: true
racecar isPalindrome: true
apple isPalindrome: false
kayak isPalindrome: true
song isPalindrome: false
noon isPalindrome: true
```

Рисунок 8 - Результат работы программы 2

4. Вывод: Мы изучили основы синтаксиса Java, научились работать с компилятором Java и виртуальной машиной Java.

5. Литература:

- a) !Методичка ТП.pdf
- b) лаб1.pdf

Репозиторий с написанным кодом находится по адресу:

https://github.com/Barsux/TTnP_lab_java1