**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

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

Кафедра «Математическая кибернетика и информационные технологии»

**Отчет по лабораторной работе №1**

по дисциплине «Информационные технологии и программирование»

на тему:

Основы синтаксиса Java

Выполнил: студент группы БФИ2101

Иванов Глеб

Проверил преподаватель:

Мосева Марина Сергеевна

Москва, 2022

**Оглавление**

[Постановка задачи 3](#_Toc122856720)

[Задания 4](#_Toc122856721)

[Задание №1 4](#_Toc122856722)

[Задача №2 6](#_Toc122856723)

[Код 8](#_Toc122856724)

[Вывод 11](#_Toc122856725)

[Список источников 12](#_Toc122856726)

**Постановка задачи**

**Задание 1.** Необходимо создать программу, которая будет выводить все простые числа до 100 (включительно).

**Задание 2.** Необходимо создать программу, которая будет показывать является ли строка палиндромом.

**Задания**

**Задание №1**

Для выполнения поставленной задачи было необходимо создать новый класс Primes. Далее внутри этого класса создать метод, который возвращает bool-значение в зависимости от того является ли число простым. Следующим шагом было создание цикла от 1 до 100 внутри main. Внутри этого цикла вызывается наш метод. При возврате true число выводится на экран консоли, если возвращается false, то ничего не происходит. Cкриншоты выполненной задачи предоставлен ниже. (Рисунок №1, Рисунок №2).



Рисунок 1 – Запуск файла Primes.java



Рисунок 2 – Проверка вывода всех простых чисел

**Задача №2**

Для выполнения поставленной задачи было необходимо создать новый класс Pcalindrome. Далее внутри него нужно было создать метод, который на вход получает строки и возвращает её обработанную версию (строка будет записана задом-наперёд). Следующим шагом создаём метод, который возвращает boll-значение, а также принимает на вход строчку, обрабатывает её с помощью выше описанного метода и производит проверку идентичности. Далее в main создаём цикл, который перебирает все элементы массива, которые поступили на вход при запуске программы, также внутри данного цикла вызывается метод который обрабатывает строчки и возвращает true, если строчки совпали после чего слово выводится на экран консоли, если строчки не совпали то возвращает false и ничего не происходит. Для запуска программы необходимо в консоли прописать java <название файла> аргумент1 аргумент2 … . Скриншот выполненной работы предоставлен ниже. (Рисунок №3).

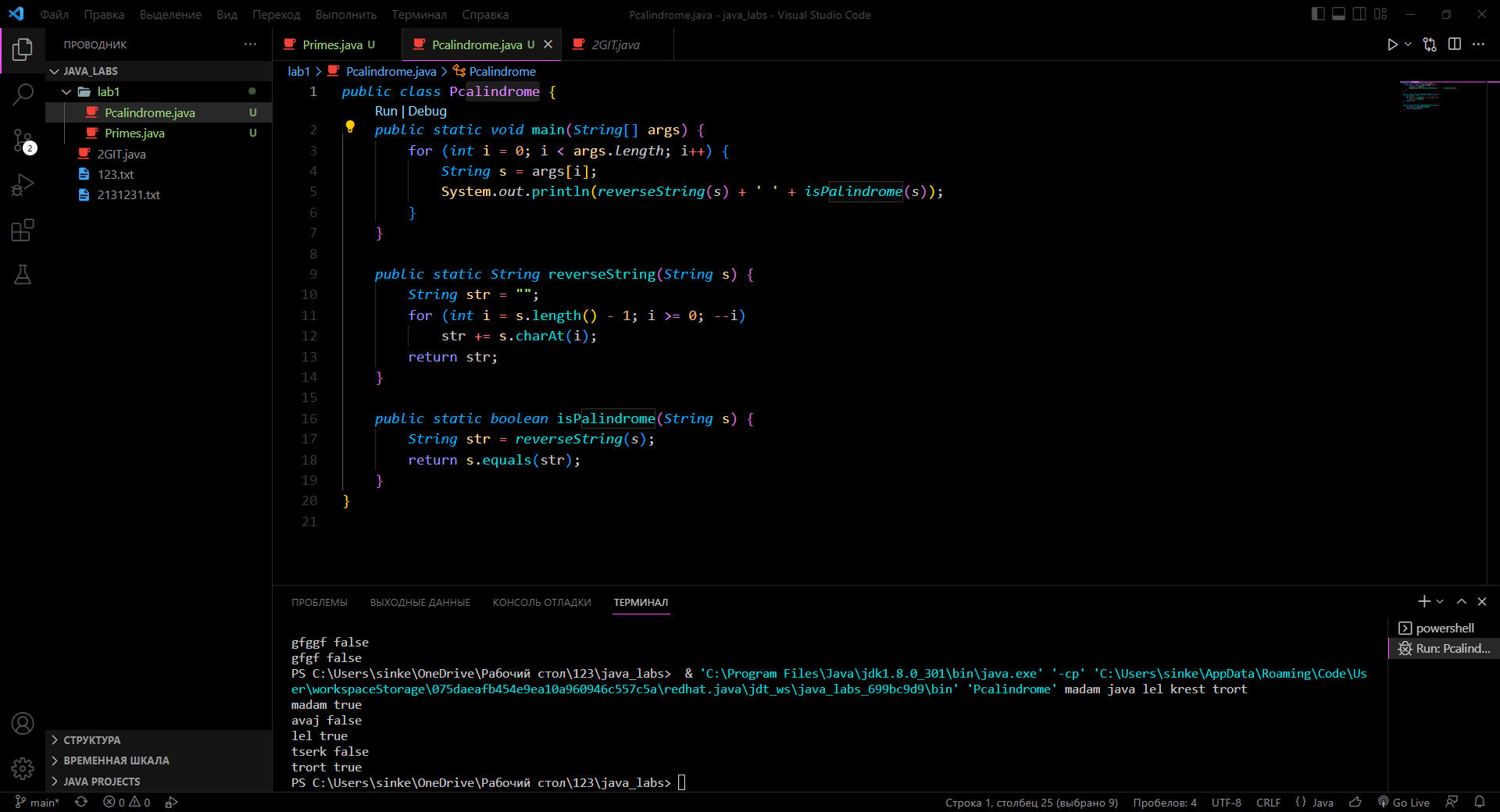


Рисунок №3 – Выполнение работы программой

Для сохранения выполненной работы необходимо запушить файлы на github. (Рисунок №4).

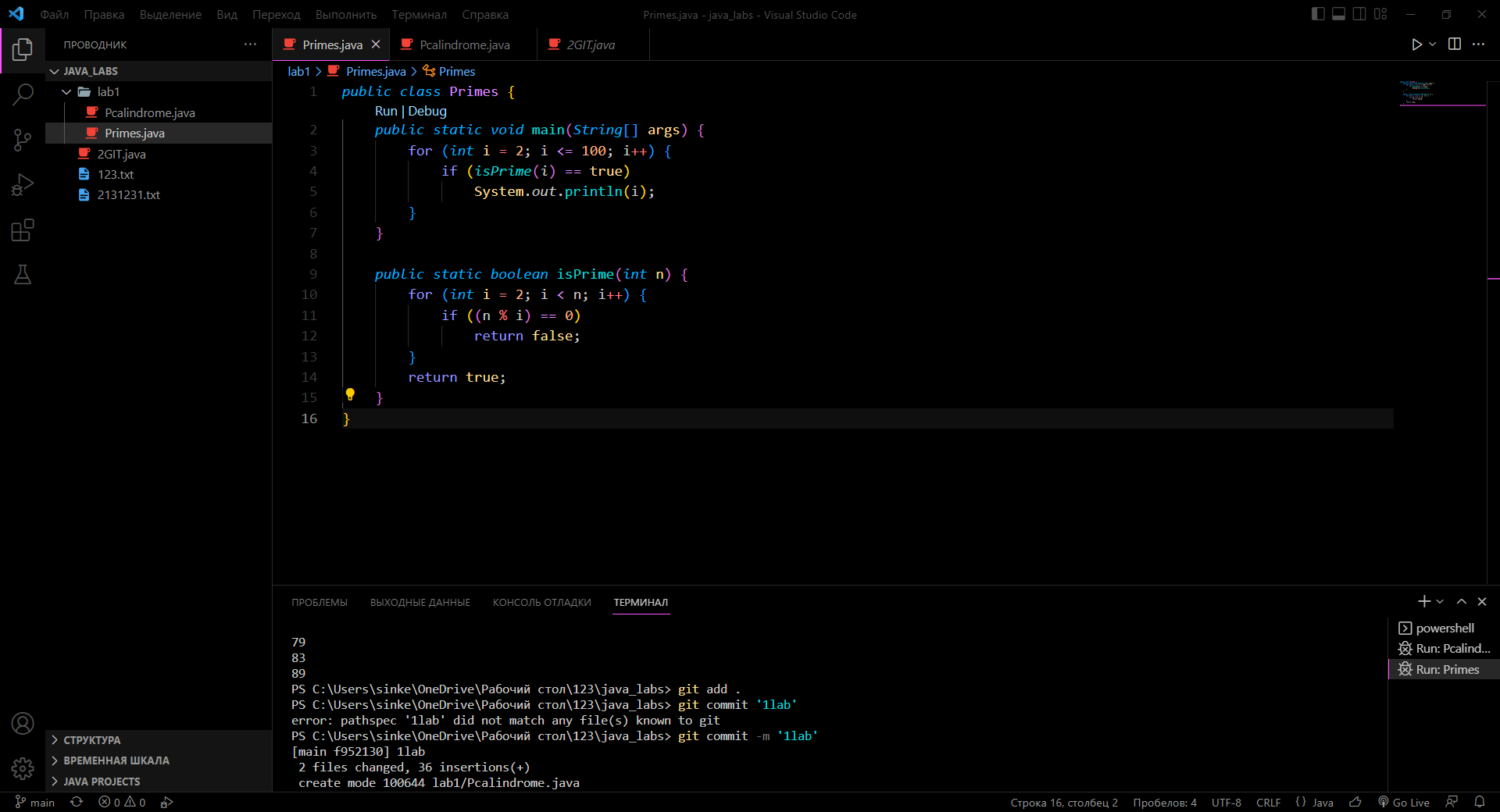


Рисунок 4 – Отправка файла на удалённый репозиторий

**Код**

*public* *class* Pcalindrome {

*public* *static* *void* main(*String*[] args) {

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

*String* s = args[i];

            if (*isPalindrome*(*s*) == true)

                System.*out*.println(*reverseString*(*s*) + ' ' + *isPalindrome*(*s*));

        }

    }

*public* *static* *String* reverseString(*String* s) {

*String* str = "";

        for (*int* i = s.length() - 1; i >= 0; --i)

            str += s.charAt(i);

        return str;

    }

*public* *static* *boolean* isPalindrome(*String* s) {

*String* str = *reverseString*(*s*);

        return s.equals(str);

    }

}

Код файла Pcalindrime.java

*public* *class* Primes {

*public* *static* *void* main(*String*[] args) {

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

            if (*isPrime*(*i*) == true)

                System.*out*.println(i);

        }

*int*[] arr = new *int*[] { 234234, 2, 3, 5, 6, 6, 223, 24, 2342, 2342, 12, 233, 4, 127, 188, 129, 135, 144 };

*checkSort*(*arr*);

*String* str = "";

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

            str += " " + arr[i];

        }

        System.*out*.println(str);

    }

*public* *static* *boolean* isPrime(*int* n) {

        for (*int* i = 2; i < n; i++) {

            if ((n % i) == 0)

                return false;

        }

        return true;

    }

*public* *static* *int*[] checkSort(*int*[] arr) {

*int* x;

*int* k;

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

            for (*int* j = 0; j < arr.*length*; j++) {

                if (arr[j] - 128 > 32) {

                    if (!(arr[j] - 160 > 32) && !(arr[j] - 160 < 0)) {

                        x = arr[j];

                        if (arr[j] - 160 < arr[i] - 160) {

                            arr[j] = arr[i];

                            arr[i] = x;

                            k = 160;

                        } else {

                            x = arr[arr.*length* - i - 1];

                            arr[arr.*length* - i - 1] = arr[i];

                            arr[i] = x;

                        }

                    } else {

                        x = arr[arr.*length* - i - 1];

                        arr[arr.*length* - i - 1] = arr[i];

                        arr[i] = x;

                    }

                } else if (!(arr[j] - 128 < 0)) {

                    x = arr[j];

                    if (arr[j] - 128 < arr[i] - 128) {

                        arr[j] = arr[i];

                        arr[i] = x;

                    }

                } else {

                    x = arr[arr.*length* - i - 1];

                    arr[arr.*length* - i - 1] = arr[i];

                    arr[i] = x;

                }

            }

        }

        return arr;

    }

}

Код файла Primes.java

**Вывод**

В ходе выполненной работы я ознакомился с основами синтаксиса языка-программирования java. Смог создать свои классы, свои методы, познакомился со встроенными методами: length(), charAt(), equlas(). Научился запускать java файл в консоли.

**Список источников**

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа, 2006.

2. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004.

3. Scott Chacon, Ben Straub «Pro Git»

4. [git-scm.com](https://git-scm.com/)