Министерство образования и науки Российской Федерации Федеральное агентство по образованию Федеральное государственное бюджетное образовательное учреждение высшего образования «Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Лабораторная работа №2

по курсу «Разработка программных систем»

Выполнил студент группы ИВТ-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Птахова А.М/

Проверил доцент кафедры ЭВМ \_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Чистяков Г.А./

Киров 2023

1. Задание

Подготовить комплект технической документации на разработанный ранее набор классов. Для выполнения лабораторной работы необходимо решить следующие задачи:

- Провести ряд преобразований программного кода, полученного в ходе выполнения предыдущей работы, с использованием встроенных средств рефакторинга.

- Сопроводить код комментариями с использованием Javadoc.

- Сгенерировать документацию к разработанным классам.

2. Листинг кода с комментариями

package lab2\_refactor;

import java.util.Arrays;

import java.util.Random;

public class PrimeGener {

private Random \_random;

private static final int LAST\_PRIME\_NUMBER = 999999937;

/\*\*

\*

\* This method determines whether a number is prime

\*

\* @param number

\* @return true or false

\*/

private boolean isPrime(int number)

{

if (number == 2) return true;

else

{

//System.out.println(pollardP1Test(number));

return pollardP1Test(number)==1 &&

fermatTest(number)==1;

}

}

/\*\*

\*

\* This method determines whether a number is prime

\* using fermat test

\*

\*

\* @param number

\* @return 1 (if number is prime) or number

\*/

private int fermatTest(int number) {

int x = (int) Math.sqrt(number), y = 0;

int r = x \* x - y \* y - number;

for (;;) {

if (r == 0) {

return x != y ? x - y : x + y;

} else {

if (r > 0) {

r -= y + y + 1;

++y;

} else {

r += x + x + 1;

++x;

}

}

}

}

private int gcd(int a, int b) {

if (a == 0) {

return b;

} else {

return gcd(b % a, a);

}

}

private int mulmod(long a, long b, long m) {

return (int) ((a \* b) % m);

}

private int powmod(int a, int n, int m) {

int res = 1;

for (int i = 0; i < n; ++i) {

res = (res \* a) % m;

}

return res;

}

/\*\*

\*

\* This method determines whether a number is prime

\* using pollard p-1 test

\*

\*

\* @param number

\* @return 1 (if number is prime) or number

\*/

private long pollardP1Test(int number) {

final int b = 13;

final int[] q = { 2, 3, 5, 7, 11, 13 };

int a = 5 % number;

for (int j = 0; j < 10; ++j) {

while (gcd(a, number) != 1) {

a = mulmod(a, a, number);

a += 3;

a %= number;

}

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

int qq = q[i];

int e = (int) Math.floor(Math.log((double) b) /

Math.log((double) qq));

int aa = (int) powmod(a, powmod(qq, e, number),

number);

if (aa == 0) {

continue;

}

int g = gcd(aa - 1, number);

if (1 < g && g < number) {

return g;

}

}

}

return 1;

}

private int getNext(int number) {

for (int i = number + 1; i < 1000000000; ++i) {

if (isPrime(i)) {

return i;

}

}

return -1;

}

/\*\*

\* Constructor

\*

\*/

public PrimeGener()

{

\_random = new Random();

}

/\*\*

\* This is method get a random prime number

\*

\* @return n - random prime number

\*/

public int getRandomNumber() {

int n = 2 + \_random.nextInt(LAST\_PRIME\_NUMBER - 1);

return isPrime(n) ? n : getNext(n);

}

/\*\*

\* This method get an array of random primes of a given length

\*

\* @param length - array length

\* @return array

\*/

public int[] getRandomArray(int length)

{

int[] array = new int [length];

for(int i=0; i<length; i++)

{

array[i]=getRandomNumber();

}

Arrays.sort(array);

return array;

}

}

3. Документация

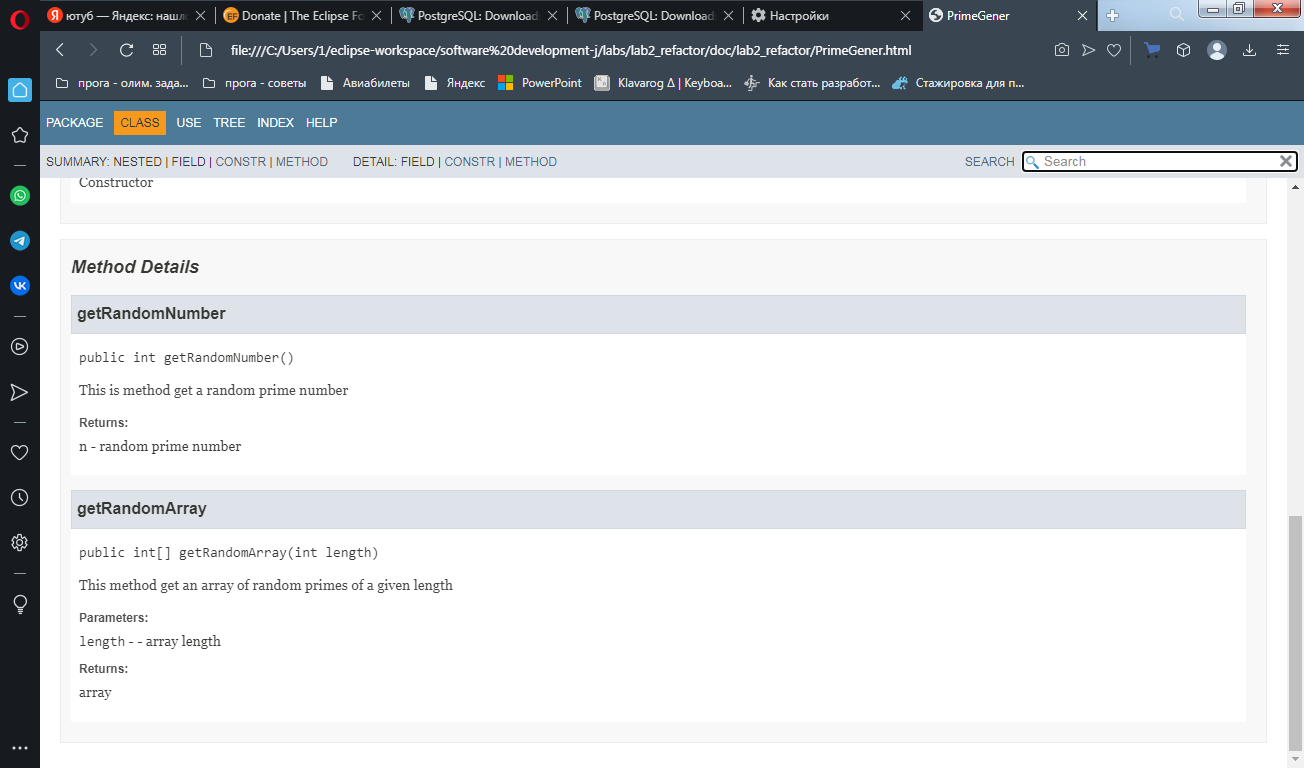


Рисунок 1 – пример разработанной документации

4. Вывод

В ходе выполнения лабораторной работы были изучены основы документирования классов Java с использованием Javadoc. А также были получены навыки использования встроенных средств рефакторинга, реализованных в среде разработки Eclipse.