МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВПО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Отчет по лабораторной работе №2

«Использование средств автоматической генерации документации и рефакторинга программного кода»

по дисциплине

«Разработка программных систем»

Выполнила студент группы ИВТ-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Седов М.Д./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Долженкова М.Л./

Киров 2020

1. Задание

В ходе выполнения лабораторной работы 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;

1. Листинг программы

Пример кода с комментариями приведен в приложении А

1. Документация

Пример документации к разработанному классу приведен в приложении Б.

1. Вывод

В ходе выполнения лабораторной работы были изучены основы Javadoc – стандарта документирования классов Java, получены навыки использования встроенных средств рефакторинга и генерации HTML-документации из Javadoc.

Приложение А

(обязательное)

Пример кода с комментариями

/\*\*

\* Returns true, and only if, second string is prefix of first string

\* @param str First string

\* @param sub Second string

\* @return true if second string is prefix of first string, otherwise false

\*/

public static boolean isPrefix(String str, String sub) {

int n = str.length(), m = sub.length();

if (m > n) return false;

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

if (sub.charAt(i) != str.charAt(i)) return false;

}

return true;

}

/\*\*

\* Returns true, and only if, second string is suffix of first string

\* @param str First string

\* @param sub Second string

\* @return true if second string is suffix of first string, otherwise false

\*/

public static boolean isSuffix(String str, String sub) {

int n = str.length(), m = sub.length();

if (m > n) return false;

for (int i = m - 1; i >= 0; i--) {

if (sub.charAt(i) != str.charAt(i + n - m)) return false;

}

return true;

}

/\*\*

\* Returns true, and only if, second string is substring of first string

\* @param str First string

\* @param sub Second string

\* @return true if second string is substring of first string, otherwise false

\*/

public static boolean isSubstring(String str, String sub) {

int m = sub.length(), n = str.length();

if (m > n) return false;

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

if (str.substring(i, i + m).equals(sub)) return true;

}

return false;

}

/\*\*a

\* Returns true, and only if, second string is subsequence of first string

\* @param str First string

\* @param sub Second string

\* @return true if second string is subsequence of first string, otherwise false

\*/

public static boolean isSubsequence(String str, String sub) {

int n = str.length(), m = sub.length();

int p = 0, t = 0, cnt = 0;

while (p < m && m - p <= n - t) {

cnt++;

if (sub.charAt(p) == str.charAt(t))

p++;

t++;

}

return p == sub.length();

}

Приложение Б

(обязательное)

Пример документации

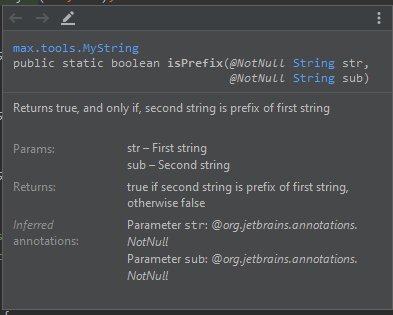


Рисунок 1 – Документация к функции isPrefix

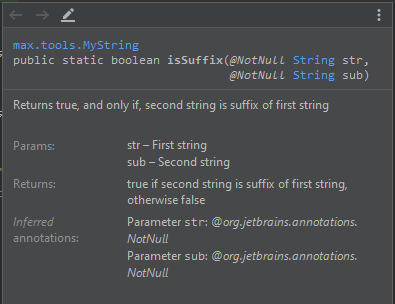


Рисунок 2 – Документация к функции isSuffix

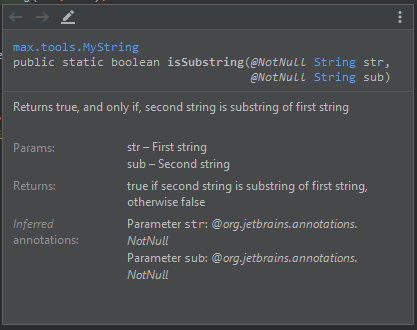


Рисунок 3 – Документация к функции isSubstring

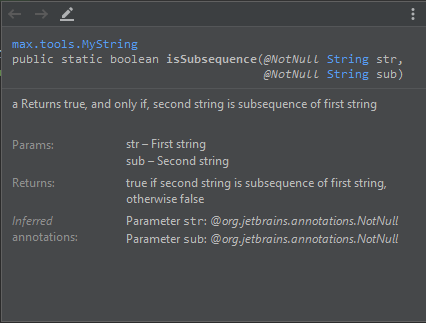


Рисунок 4 – Документация к функции isSubsequence