

Основи програмування
Лабораторна робота №14
Тема: Потоки вводу-виводу

Виконали студенти групи ІА-23:

Содолинський Вадим
Калина Сергій

Шрубович Наталія

Перевірив:

Колеснік В. М.

Хід роботи:

1. Ознайомитись з API класів та інтерфейсів для здійснення операцій вводу-виводу. Особливу увагу звернути на такі класи та інтерфейси:

- InputStream
 - FileInputStream
- OutputStream
 - FileOutputStream
- Reader
 - FileReader
- Writer
 - FileWriter
- AutoCloseable
 - Closable
- IOException

2. Виконати завдання з таблиці 2 відповідно до свого варіанту у таблиці 1.

- Кожне завдання має бути реалізовано як окремий клас.
- Кожен клас має складатись щонайменше з двох методів:
 - public static void main(String[] args) - точка входу.
 - Метод, що реалізує задане завдання. Метод має перевіряти аргументи та у разі їх помилковості аварійно закінчувати свою роботу шляхом викидання стандартного виключення IllegalArgumentException або NullPointerException. В разі неможливості виконання операції, метод повинен викидати IOException або FileNotFoundException. В жодному разі цей метод не повинен напряму взаємодіяти з користувачем через консоль або інший UI (ніколи не змішуйте бізнес-логіку та користувацький інтерфейс).
- Клас може містити інші допоміжні методи.
- При виконанні завдань слід звернути увагу на ефективність з точки зору швидкодії. При виконанні завдань 1-6 слід використовувати клас BufferedReader та BufferedWriter, а при 7-11 – ні в якому разі не намагались обробляти усі байти по одному, а використовувати методи read(byte[] b) та write(byte[] b), які працюють з масивами.

Завдання 1(1)

Знайти у текстовому файлі рядок найбільшої довжини. Якщо є кілька рядків, довжина яких дорівнює максимальній, повернути перший з них.

В цьому завданні я використав `BufferedReader` з розміром буферу 512. Далі створюємо цикл, який працює поки рядок вказаного файлу не буде дорівнювати `null`. Потім порівнюється довжина теперішнього рядка і попереднього. Метод повертає найбільший рядок.

```
static String firstLongestString(String fileName) throws IOException{
    String longestString = "";
    String presentString;
    try (BufferedReader bufferedReader = new BufferedReader(new FileReader(fileName), 512);) {
        while ((presentString = bufferedReader.readLine()) != null) {
            if(longestString.length() < presentString.length()){
                longestString = presentString;
            }
        }
    } catch (FileNotFoundException ex){
        longestString = "File Not Found";
    } finally{
        return longestString;
    }
}
```

Результат

```
TaskONE
"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:A:\IntelliJ IDEA Ultimate\
Найдовший рядок файлу: London is a capital of United Kingdom

Process finished with exit code 0
```

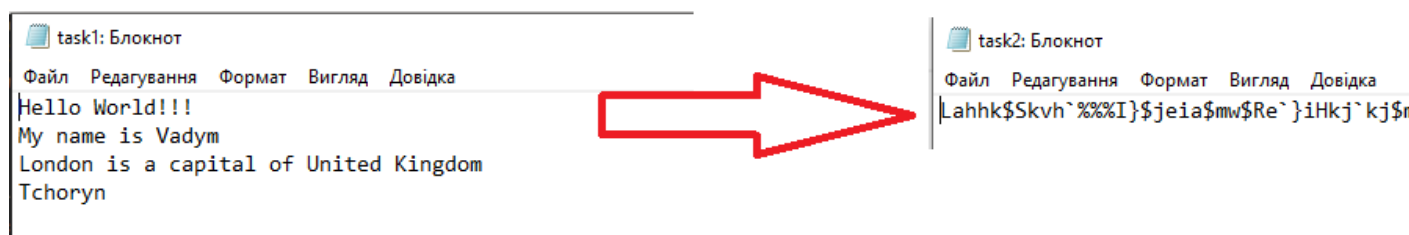
Завдання 2(8)

Створити зашифровану копію файлу, замінивши кожен з байтів сумою за модулем 2 (XOR) його значення та ключа <key>.

В цьому завданні використовуємо `BufferedReader` та `BufferWriter`. `BufferedReader` зчитує рядок у `inputFile` а потім перетворюємо зчитаний рядок у байти. Потім цей масив байтів і значення `key` сумуємо за модулем 2 (XOR). Далі цей масив перетворюємо назад у `String` і за допомогою `BufferWriter` записуємо у `outputFile`. Цей цикл працює поки у `inputFile` не Прочитаємо всі рядки.

```
static void cypher(String source, String destination, byte key) throws IOException {
    String line = "";
    try(BufferedReader bufferedReader = new BufferedReader(new FileReader(source), 512);
        BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(destination), 512)){
        while ((line = bufferedReader.readLine()) != null){
            byte[] byteText = line.getBytes( "UTF-8");
            for(int i = 0; i<byteText.length; i++){
                byteText[i] = (byte)(byteText[i]^key);
            }
            String newLine = new String(byteText, "UTF-8");
            bufferedWriter.write(newLine);
        }
    }catch (FileNotFoundException ex){
        System.out.println("File not found");
    }
}
```

Результат



Висновок: ми ознайомилися з потоками вводу-виводу та навчилися користуватися ними.