**Computer Language 2022**

**Assignment #8**

**Due: 30/May 23:59:59**

**Student ID: 21102052 Name: Lee Jeong-Yun**

**1. Read an arbitrary file and print it. Use Scanner with FileReader.**

**Output) (example of reading c:\windows\system.ini)**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Your code:**

import java.io.File;  
import java.io.FileReader;  
import java.io.IOException;  
import java.util.Scanner;  
  
public class ScannerFileReader {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("Printing ");  
 File file = new File(scanner.nextLine());  
 FileReader fileReader = null;  
  
 try {  
 fileReader = new FileReader(file);  
 int c;  
 while ((c=fileReader.read()) != -1) {  
 System.*out*.print((char) c);  
 }  
 fileReader.close();  
 }  
 catch (IOException e) {  
 System.*out*.println("IO Error!");  
 }  
  
 }  
}

**Your result (screenshot):**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**Your explanation on the code:**

In the main method, Scanner is instantiated and get the string type value using the scanner. Then, I made Filereader to read the file. FileReader.read() method would return the int type value, so I needed to cast the value type from int to char. The loop would print out the character one by one with the read() method. Also, to handle the exception, I added try – catch statement.

**Q2. Read an arbitrary file and print it with line numbers.**

**Output) (example of reading c:\windows\system.ini)**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Your code:**

import java.io.File;  
import java.io.IOException;  
import java.util.Scanner;  
  
public class lineNumebrs {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 try {  
 System.*out*.print("Reading ");  
 String fileName = scanner.next();  
 Scanner fileScanner = new Scanner(new File(fileName));  
 int number = 1;  
 while (fileScanner.hasNext()) {  
 System.*out*.printf(number + ":\t");  
 String line = fileScanner.nextLine();  
 System.*out*.println(line);  
 number++;  
 }  
 }  
 catch (IOException e) {  
 System.*out*.println("IO Error!");  
 }  
 }  
}

**Your result (screenshot):**

**텍스트, 명판, 스크린샷이(가) 표시된 사진

자동 생성된 설명**

**Your explanation on the code:**

I used Scanner to read the txt file line by line. In the try statement, I got the file name or path using scanner and the string get by scanner is put into another Scanner, fileScanner. Using while loop statement, if the fileScanner has next, then print out the line number and each line’s contents.

**Q3. Write a program that takes two file names from the user and creates a new file which is made by appending the second file to the first file.**

**Output)**

텍스트이(가) 표시된 사진

자동 생성된 설명

(elvis1.txt)

텍스트이(가) 표시된 사진

자동 생성된 설명

(elvis2.txt)

텍스트이(가) 표시된 사진

자동 생성된 설명

(appended.txt)

텍스트이(가) 표시된 사진

자동 생성된 설명

**Your code:**

import java.io.\*;  
import java.util.Scanner;  
  
public class appendingFile {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("Input your first name >> ");  
 String oneName = scanner.next();  
 System.*out*.print("Input your second name >> ");  
 String twoName = scanner.next();  
  
 File one = new File(oneName);  
 File two = new File(twoName);  
 File three = new File("three.txt");  
  
 int o1, o2;  
  
 try {  
 FileInputStream oneFileReader = new FileInputStream(one);  
 FileInputStream twoFileReader = new FileInputStream(two);  
 FileOutputStream threeFileWriter = new FileOutputStream(three);  
  
 while ((o1=oneFileReader.read()) != -1) {  
 threeFileWriter.write((char) o1);  
 }  
  
 threeFileWriter.write('\n');  
  
 while ((o2=twoFileReader.read()) != -1) {  
 threeFileWriter.write((char) o2);  
 }  
 System.*out*.println("Complete.");  
  
 }  
 catch (IOException e) {  
 System.*out*.println("IO Error!");  
 }  
  
 }  
}

**Your result (screenshot):**

텍스트이(가) 표시된 사진

자동 생성된 설명

letItBe.txt



bohemianRhapsody.txt

텍스트이(가) 표시된 사진

자동 생성된 설명

three.txt

텍스트이(가) 표시된 사진

자동 생성된 설명

**Your explanation on the code:**

**I used two Scanners to get first and second file names for file merging. And make File objects for existing two files and the a merged file. Int type value o1 and o2 to write the contents into file three.txt. In the try statement, FileInputStream and FileOutStream were instantiated for each file. First, the file one is read and written into file three.txt and then file two is read and written into the three.txt. After the writing, the complete message is printed out.**

**Q4. Write a program that finds and prints the biggest file in C:\. Use File class.**

**Output)**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Your code:**

import java.io.File;  
  
public class maxSize {  
 public static void main(String[] args) {  
 File cDrive = new File("C:\\");  
 File[] files = cDrive.listFiles();  
 File maxFile = new File("");  
 for (File f : files) {  
 if (maxFile.length() < f.length()) {  
 maxFile = f;  
 }  
 }  
 System.*out*.println("The biggest file is "+maxFile.getPath());  
 System.*out*.println("Size: "+maxFile.length()+" bytes");  
  
 }  
}

**Your result (screenshot):**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**Your explanation on the code:**

**cDrive File object is declared and File array files is consisted of the cDrive’s file list using the listFile() method. Then, advanced for loop is used to get each file in the array and the each file’s size is compared to the maxSize’s size by if statement. If the f.length() is bigger than maxSize,length(), then the f is maxSize until the bigger one appears in the comparison. After the loop, biggest file’s name and its size is showed in console.**

**Congratulations!**

**You’ve reached the last line of this semester 😊**

**The time and efforts to solve the assignments will not betray you!**

**Java will be used as a fundamental language in the following modules: e.g., Data Structure and Mobile Programming (actually, Kotlin which is 100% Java compliant modern language).**

**Let’s see again next year @ Mobile Programming course!**

**😍**