Assignment No - 08

SET A

1. Write a java program that displays the number of characters, lines and words of a file.

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class FileMetrics {

public static void main(String[] args) {

String filePath = "input.txt";

int charCount = 0;

int wordCount = 0;

int lineCount = 0;

try (BufferedReader br = new BufferedReader(new FileReader(filePath))) {

String line;

while ((line = br.readLine()) != null) {

lineCount++;

charCount += line.length();

String[] words = line.split("\\s+");

wordCount += words.length;

}

System.out.println("Number of Characters: " + charCount);

System.out.println("Number of Words: " + wordCount);

System.out.println("Number of Lines: " + lineCount);

} catch (IOException e) {

System.out.println("An error occurred while reading the file.");

e.printStackTrace();

}

}

}

Output

Number of Characters: 54

Number of Words: 9

Number of Lines: 3

1. Write a java program to display name and size of the given files,

import java.io.File;

public class FileInfoDisplay {

public static void main(String[] args) {

String[] filePaths = {

"file1.txt",

"file2.jpg",

"document.pdf"

};

for (String filePath : filePaths) {

File file = new File(filePath);

if (file.exists() && file.isFile()) {

System.out.println("File Name: " + file.getName());

System.out.println("File Size: " + file.length() + " bytes");

} else {

System.out.println("File not found or not a valid file: " + filePath);

}

System.out.println();

}

}

}

Output

File Name: file1.txt

File Size: 150 bytes

File Name: file2.jpg

File Size: 1024 bytes

File Name: document.pdf

File Size: 2048 byte

3. Write a java program to copy the data from one file into another file, while copying change the case of characters in target file and replaces all digits by ‘\*’ symbol.

**import java.io.\*;**

**class FileCopier**

**{**

**public static void main(String args[])throws IOException**

**{**

**int c;**

**try**

**{**

**FileReader fr=new FileReader("a.txt");**

**FileWriter fw=new FileWriter("b.txt");**

**while((c=fr.read())!=-1)**

**{**

**if(c>=65&&c<=90)**

**{**

**c=c+32;**

**fw.write(c);**

**}**

**else if(c>=97&&c<=122)**

**{**

**c=c-32;**

**fw.write(c);**

**}**

**else if(c>=48&&c<=57)**

**{**

**fw.write('\*');**

**}**

**else**

**{**

**fw.write(c);**

**}**

**}**

**System.out.println("Copy Successfully");**

**fr.close();**

**fw.close();**

**}catch(Exception e)**

**{**

**System.out.println(e);**

**}**

**}**

**}**

4. Write a java program to display content s of a file in reverse order.

**import java.io.\*;**

**class ReverseFileContent**

**{**

**public static void main(String arg[])throws IOException**

**{**

**FileReader fr=new FileReader("input.txt");**

**FileWriter fw=new FileWriter("output.txt");**

**BufferedReader b=new BufferedReader(fr);**

**String data;**

**String reverse;**

**while((data=b.readLine())!=null)**

**{**

**String words[]=data.split(" ");**

**for(String a:words)**

**{**

**StringBuilder builder=new StringBuilder(a);**

**System.out.println(builder.reverse().toString());**

**}**

**}**

**}**

**}**

**input.txt save in bin directory**

**hello world**

**output.txt save in bin directory**

**output:**

**C:\Users\BBA11>cd C:\Program Files\Java\jdk1.8.0\_144\bin**

**C:\Program Files\Java\jdk1.8.0\_144\bin>javac slip21.java**

**C:\Program Files\Java\jdk1.8.0\_144\bin>java slip21**

**olleh**

**dlrow**

* 1. import java.io.FileInputStream;
  2. import java.io.File;

**SET B Find the Error:**

1)

import java.io.IOException;

public class Example

{

public static void main(String[] args)

{

try

{

FileInputStream fis = new FileInputStream("example.txt");

int data = fis.read();

while (data != -1)

{

System.out.print((char) data);

data = fis.read();

}

// No fis.close() here

}

catch (IOException e)

{

System.out.println("An error occurred: " + e.getMessage());

}

}

}

**Error Solve Program**

import java.io.IOException;

import java.io.FileInputStream;

public class Example {

public static void main(String[] args) {

try (FileInputStream fis = new FileInputStream("example.txt")) {

int data = fis.read();

while (data != -1) {

System.out.print((char) data);

data = fis.read();

}

} catch (IOException e) {

System.out.println("An error occurred: " + e.getMessage());

}

}

}

2)

* 1. import java.io.File;

import java.util.Scanner;

public class Example6

{

public static void main(String[] args)

{

File file = new File("nonexistent.txt");

Scanner reader = new Scanner(file);

while (reader.hasNextLine())

{

System.out.println(reader.nextLine());

}

reader.close();

}

}

**Error Solve Program**

import java.io.File;

import java.util.Scanner;

public class Example6 {

public static void main(String[] args) {

File file = new File("nonexistent.txt");

if (file.exists()) {

try (Scanner reader = new Scanner(file)) {

while (reader.hasNextLine()) {

System.out.println(reader.nextLine());

}

}

} else {

System.out.println("File not found.");

}

}

}

3)

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Example7

{

public static void main(String[] args)

{

try {

File file = new File("invalid/path/to/file.txt");

Scanner reader = new Scanner(file);

while (reader.hasNextLine())

{

System.out.println(reader.nextLine());

}

reader.close();

}

catch (FileNotFoundException e)

{

System.out.println("File not found: " + e.getMessage());

}

}

}

**Error Solve Program**

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

import java.io.IOException;

public class Example7 {

public static void main(String[] args) {

try (Scanner reader = new Scanner(new File("invalid/path/to/file.txt"))) {

while (reader.hasNextLine()) {

System.out.println(reader.nextLine());

}

} catch (FileNotFoundException e) {

System.out.println("File not found: " + e.getMessage());

} catch (IOException e) {

System.out.println("Error reading file: " + e.getMessage());

}

}

}

SET C

1. Write a java program to accept list of file names through command line. Delete the files having extension .txt. Display name, location and size of remaining files.

**import java.io.\*;**

**class FileDeleter {**

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

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

**File file=new File(args[i]);**

**if(file.isFile()) {**

**String name = file.getName();**

**if(name.endsWith(".txt")) {**

**file.delete();**

**System.out.println("file is deleted " + file);**

**}else{**

**System.out.println("File Name : " + name + "\nFile**

**Location : " +file.getAbsolutePath()+"\nFile Size :**

**"+file.length()+" bytes");**

**}**

**}**

**else{**

**System.out.println(args[i]+ "is not a file");**

**}**

**}**

**}**

**}**

1. Write a java program to display the files having extension .txt from a given directory.

**import java.io.File;**

**class TextFileFinder {**

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

**File file = new**

**File("C:\\Users\\Saurabh\_Sapkal\\Desktop\\ln\\java\\Slips");**

**String[] fileList = file.list();**

**for(String str : fileList) {**

**if(str.endsWith(".txt")){**

**System.out.println(str);**

**}**

**}**

**}**

**}**

1. Write a java program to count number of lines, words and characters from a given file.

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class FileMetricsCounter {

public static void main(String[] args) {

String filePath = "sample.txt";

int lineCount = 0;

int wordCount = 0;

int charCount = 0;

try (BufferedReader br = new BufferedReader(new FileReader(filePath))) {

String line;

while ((line = br.readLine()) != null) {

lineCount++;

charCount += line.length();

String[] words = line.split("\\s+");

wordCount += words.length;

}

System.out.println("Number of Lines: " + lineCount);

System.out.println("Number of Words: " + wordCount);

System.out.println("Number of Characters: " + charCount);

} catch (IOException e) {

System.out.println("An error occurred while reading the file.");

e.printStackTrace();

}

}

}

Output

Number of Lines: 3

Number of Words: 10

Number of Characters: 56

1. Write a java program to accept details of n Employees (e\_id, cname, address, mobile\_no) from user and store it in a file (Use DataOutputStream class). Display the details of customers by reading it from file.(Use DataInputStream class)

import java.io.\*;

import java.util.Scanner;

public class EmployeeDetails {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String filePath = "employees.dat";

System.out.print("Enter the number of employees: ");

int n = scanner.nextInt();

scanner.nextLine();

try (DataOutputStream dos = new DataOutputStream(new FileOutputStream(filePath))) {

for (int i = 1; i <= n; i++) {

System.out.println("Enter details for Employee " + i + ":");

System.out.print("Employee ID: ");

int e\_id = scanner.nextInt();

scanner.nextLine();

System.out.print("Employee Name: ");

String cname = scanner.nextLine();

System.out.print("Employee Address: ");

String address = scanner.nextLine();

System.out.print("Employee Mobile Number: ");

String mobile\_no = scanner.nextLine();

dos.writeInt(e\_id);

dos.writeUTF(cname);

dos.writeUTF(address);

dos.writeUTF(mobile\_no);

}

} catch (IOException e) {

System.out.println("An error occurred while writing to the file.");

e.printStackTrace();

}

System.out.println("\nEmployee Details from File:");

try (DataInputStream dis = new DataInputStream(new FileInputStream(filePath))) {

while (dis.available() > 0) {

int e\_id = dis.readInt();

String cname = dis.readUTF();

String address = dis.readUTF();

String mobile\_no = dis.readUTF();

System.out.println("\nEmployee ID: " + e\_id);

System.out.println("Employee Name: " + cname);

System.out.println("Employee Address: " + address);

System.out.println("Employee Mobile Number: " + mobile\_no);

}

} catch (IOException e) {

System.out.println("An error occurred while reading from the file.");

e.printStackTrace();

}

scanner.close();

}

}

Output

Enter the number of employees: 2

Enter details for Employee 1:

Employee ID: 101

Employee Name: John Doe

Employee Address: 123 Elm Street

Employee Mobile Number: 555-1234

Enter details for Employee 2:

Employee ID: 102

Employee Name: Jane Smith

Employee Address: 456 Oak Avenue

Employee Mobile Number: 555-567

Output

Employee Details from File:

Employee ID: 101

Employee Name: John Doe

Employee Address: 123 Elm Street

Employee Mobile Number: 555-1234

Employee ID: 102

Employee Name: Jane Smith

Employee Address: 456 Oak Avenue

Employee Mobile Number: 555-5678