

**(Established under the Presidency University Act, 2013 of the Karnataka Act 41 of 2013)**

## CSA2022 – Advanced JAVA Programming LAB SHEET - 6

**Module 2 – Input Output Operations in Java**

**Operations with Text files**

**Q1**. Write a Java program to perform the following operations with text files.

1. Create a folder, student name as folder name in c drive by passing the folder name at run time using Scanner.
2. Create a file to write an essay “shortnnoteson multithreading”, file name must be “multithread.txt”.
3. Read the content of the file to output console.
4. Copy the content of above file to another file studentname\_multithread\_copy.txt.

**Aim :** The aim of this program is to create a folder with a name provided at runtime using the Scanner class.

**Algorithm:**

1. Import the necessary libraries, including **java.io.\*** and **java.util.Scanner**.
2. Create a Scanner object to take user input.
3. Prompt the user to enter the folder name.
4. Read the folder name from the user using **sc.next()**.
5. Create a File object with the given folder name.
6. Check if the folder already exists and is a directory using **exists()** and **isDirectory()** methods.
7. If the folder does not exist, create the folder using the **mkdir()** method.
8. Print a success message if the folder is successfully created.

**Program Explanation:** This p art of the program allows the user to create a folder with a name entered at runtime. It uses the **Scanner** class to take user input for the folder name. The program then checks if the folder already exists and if it is a directory. If the folder does not exist, it is created, and a success message is displayed.

**Program:**

a) Creating folder

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

**import** java.util.Scanner;

**public** **class** CreateFolder

{

**public** **static** **void** main(String arr[])

{

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter folder name");

String foldername=sc.next();

File f=**new** File(foldername);

**if**(f.exists()&&f.isDirectory())

System.***out***.println("already exist");

**else** **if**(f.mkdir()==**true**)

System.***out***.println("successfully created");

**else**

System.***out***.println("cannot be created");

}

}

**Output:**

Enter the folder name

5BCA2

Successfully created

**Aim:** The aim of this program is to create a text file named "multithread.txt" and write an essay to it.

**Algorithm:**

1. Import the necessary libraries, including **java.io.FileWriter** and **java.util.Scanner**.
2. Define the content of the essay as a string.
3. Create a Scanner object to take user input.
4. Prompt the user to enter the file name and folder name.
5. Read the file name and folder name from the user using **sc.next()**.
6. Create a **FileWriter** object with the folder name and file name.
7. Write the essay content to the file using the **write** method.
8. Close the FileWriter to save the changes to the file.

**Program Explanation:** This part of the program allows the user to create a text file named "multithread.txt" within a specified folder and write an essay to it. The user is prompted to enter the file name and folder name, and the program creates the file, writes the essay content, and saves it.

**Program:**

b) Creating file

Output:

File created

File written

**Aim:** The aim of this program is to read the content of the file "multithread.txt" and display it on the console.

**Algorithm:**

1. Import the necessary libraries, including **java.io.BufferedReader** and **java.io.FileReader**.
2. Create a File object for the file "multithread.txt."
3. Create a **FileReader** object to read the file.
4. Create a **BufferedReader** object to read the file line by line.
5. Use a while loop to read each line from the file until the end of the file is reached.
6. Display each line on the console.
7. Close the BufferedReader to release resources.

**Program Explanation:** This part of the program reads the content of the file "multithread.txt" and displays it on the console. It uses a **FileReader** and a **BufferedReader** to read the file line by line and prints each line to the console until the end of the file is reached.

**Program:**

c) Reading File

**import** java.io.BufferedReader;

**import** java.io.File;

**import** java.io.FileReader;

**public** **class** Main {

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

File fileob=**new** File("5BCA2\\multithread.txt");

FileReader fr = **new** FileReader(fileob);

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

String line;

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

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

}

br.close();

}}

**Output:**

Multithreading is Concurrent execution of threads.

Enhances performance and responsiveness.

Utilizes multiple CPU cores.

**Aim:** The aim of this program is to copy the content of "multithread.txt" to another file named "studentname\_multithread\_copy.txt."

**Algorithm:**

1. Import the necessary libraries, including **java.io.\***.
2. Create a File object for the source file "multithread.txt" and the target file "studentname\_multithread\_copy.txt."
3. Create a **FileReader** and a **FileWriter** for the source and target files, respectively.
4. Create a **BufferedReader** to read from the source file and a **PrintWriter** to write to the target file.
5. Use a while loop to read each line from the source file, write it to the target file, and append a newline character.
6. Close the BufferedReader, PrintWriter, FileReader, and FileWriter to save and release resources.

**Program Explnation:** This part of the program copies the content of the file "multithread.txt" to another file named "studentname\_multithread\_copy.txt." It uses **FileReader**, **FileWriter**, **BufferedReader**, and **PrintWriter** to read and write the file's content, ensuring that each line is copied to the target file.

Top of Form

**Program:**

d) Copying File

//either PrintWriter or BufferWriter

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

**public** **class** Main {

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

File source=**new** File("5BCA2\\multithread.txt");

File target=**new** File("5BCA2\\multithread\_copy.txt");

FileReader fr = **new** FileReader(source);

FileWriter fw = **new** FileWriter(target);

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

//BufferedWriter bw = new BufferedWriter(fw);

PrintWriter pw = **new** PrintWriter(fw, **false**);

String line;

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

pw.write(line); //bw.write(line);

pw.println(); //bw.newLine();

}

System.***out***.println("1 file copied");

br.close();

//bw.close();

pw.close();

}

}

**Output:**

1 file copied