Assignment 7 :-

1. **Write a Java program that reads data from a sample.txt file located outside the program's directory.**

import java.io.\*;

public class ReadExternalFile {

public static void main(String[] args) {

try {

File file = new File("sample.txt"); // same folder as your .java file

BufferedReader br = new BufferedReader(new FileReader(file));

String line;

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

System.out.println(line);

}

br.close();

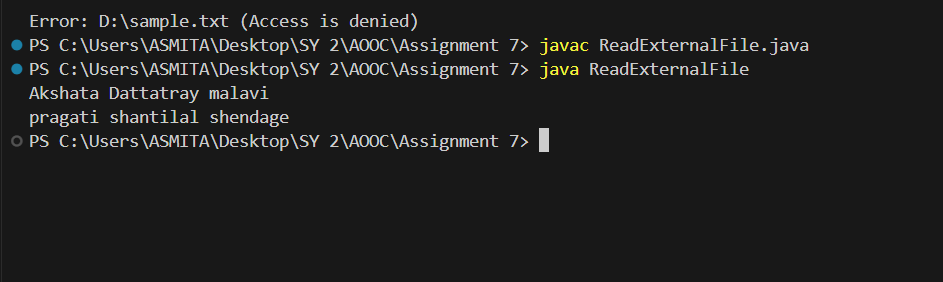
} catch (IOException e) {

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

}

}

}



**2) Develop a Java program that performs the following operations:**

**Accept student information such as name, age, weight, height, city, and phone number from the user.**

**Store this information in a file using DataOutputStream along with FileOutputStream.**

**Retrieve and display the data using DataInputStream along with FileInputStream.**

import java.io.\*;

import java.util.Scanner;

public class StudentDataIO {

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

Scanner sc = new Scanner(System.in);

DataOutputStream dos = new DataOutputStream(new FileOutputStream("student.dat"));

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

dos.writeUTF(sc.nextLine());

System.out.print("Age: ");

dos.writeInt(sc.nextInt());

System.out.print("Weight: ");

dos.writeDouble(sc.nextDouble());

System.out.print("Height: ");

dos.writeDouble(sc.nextDouble());

sc.nextLine(); // clear buffer

System.out.print("City: ");

dos.writeUTF(sc.nextLine());

System.out.print("Phone Number: ");

dos.writeUTF(sc.nextLine());

dos.close();

DataInputStream dis = new DataInputStream(new FileInputStream("student.dat"));

System.out.println("\n--- Retrieved Student Data ---");

System.out.println("Name: " + dis.readUTF());

System.out.println("Age: " + dis.readInt());

System.out.println("Weight: " + dis.readDouble());

System.out.println("Height: " + dis.readDouble());

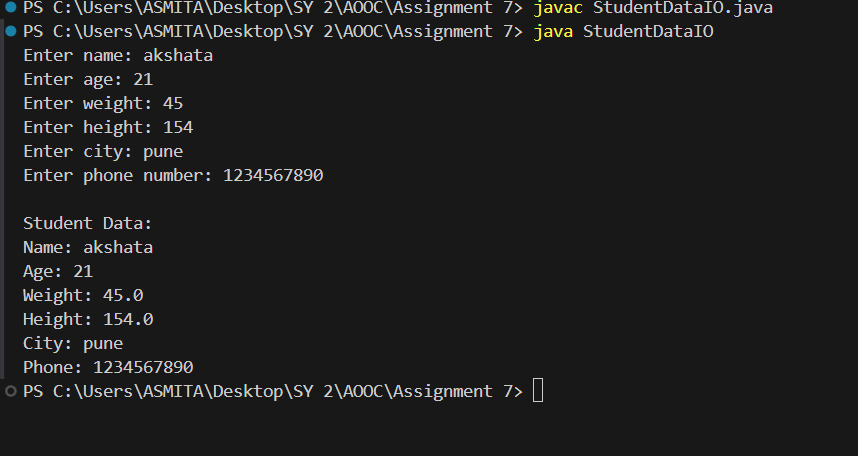
System.out.println("City: " + dis.readUTF());

System.out.println("Phone: " + dis.readUTF());

dis.close();

}

}



**3. Write a Java program to read a text file and compute the following:**

**The total number of vowels in the file.**

**The total number of words in the file,**

**The number of times the character 'a' appears in the file.**

import java.io.\*;

public class FileAnalyzer {

public static void main(String[] args) {

int vowels = 0, words = 0, aCount = 0;

try {

BufferedReader reader = new BufferedReader(new FileReader("sample.txt"));

String line;

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

words += line.split("\\s+").length;

for (char ch : line.toLowerCase().toCharArray()) {

if ("aeiou".indexOf(ch) != -1) vowels++;

if (ch == 'a') aCount++;

}

}

reader.close();

System.out.println("Total vowels: " + vowels);

System.out.println("Total words: " + words);

System.out.println("Occurrences of 'a': " + aCount);

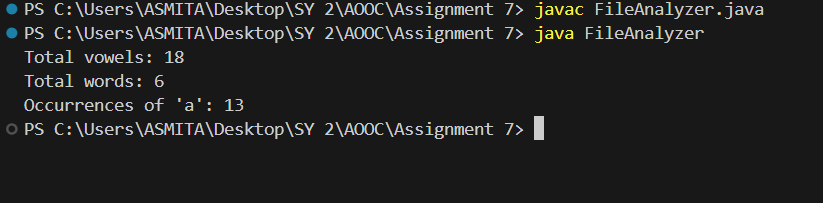
} catch (IOException e) {

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

}

}

}



**4. Write a program that takes a file name as input through the command line.**

**If the file exists, open it and display its contents.**

**After displaying the contents, ask the user: "Do you want to add data to the end of the file?"**

**If the user's response is "Yes", accept data from the user and append it to the file.**

**If the file does not exist, create a new file and allow the user to input data to store in it.**

**The user should type "exit" on a new line to stop entering data. Implement this program using character stream classes.**

import java.io.\*;

import java.util.Scanner;

public class FileEditor {

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

Scanner sc = new Scanner(System.in);

if (args.length < 1) {

System.out.println("Please provide a file name as command line argument.");

return;

}

String fileName = args[0];

File file = new File(fileName);

if (file.exists()) {

System.out.println("File exists. Contents:");

BufferedReader br = new BufferedReader(new FileReader(file));

String line;

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

System.out.println(line);

}

br.close();

System.out.print("\nDo you want to add data to the end of the file? (Yes/No): ");

String response = sc.nextLine();

if (response.equalsIgnoreCase("Yes")) {

FileWriter fw = new FileWriter(file, true);

System.out.println("Enter data to append (type 'exit' to stop):");

while (true) {

String data = sc.nextLine();

if (data.equalsIgnoreCase("exit")) break;

fw.write(data + "\n");

}

fw.close();

}

} else {

System.out.println("File does not exist. Creating new file.");

FileWriter fw = new FileWriter(file);

System.out.println("Enter data to store (type 'exit' to stop):");

while (true) {

String data = sc.nextLine();

if (data.equalsIgnoreCase("exit")) break;

fw.write(data + "\n");

}

fw.close();

}

}

}

