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

```
import java.io.FileInputStream;
import java.io.IOException;

public class Main{
	public static void main(String[] args){
	try{
	FileInputStream fos = new
FileInputStream("/home/sumedhk/SEM_4/AdvanceJava/AOOC/EXPERIMENT_7/sample.txt");
	int i;

	while((i = fos.read()) != -1)
		System.out.print((char)i);

	fos.close();
	}
	catch(IOException e){
	e.printStackTrace();
	}
	}
}
```

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 Student{
	public static void main(String[] args){
	Scanner scanner = new
Scanner(System.in);
	String filename = "student_info.dat";
		System.out.println("Enter Student
Details");

		System.out.print("Student name: ");
		String name = scanner.nextLine();

		System.out.print("Age: ");
		int age = scanner.nextInt();

		System.out.print("Weight (in kg): ");
```

```
        double weight =
scanner.nextDouble();

        System.out.print("height (in cm): ");
        int height = scanner.nextInt();
        scanner.nextLine();

        System.out.print("City: ");
        String city = scanner.nextLine();

        System.out.print("phone number: ");
        String ph_no = scanner.nextLine();

        try{
        DataOutputStream dos = new
DataOutputStream(new
FileOutputStream(filename));
        dos.writeUTF(name);
        dos.writeInt(age);
        dos.writeDouble(weight);
        dos.writeInt(height);
        dos.writeUTF(city);
        dos.writeUTF(ph_no);

        System.out.println("Data Written
Sucessfully to " + filename);
        dos.close();
        }catch(IOException e){
        System.out.println("Caught
Exception: " + e.getMessage());
        }
        try{
        DataInputStream ios = new
DataInputStream(new
FileInputStream(filename));

        System.out.println("Name of the
Student: " + ios.readUTF());
        System.out.println("Age: " +
ios.readInt());
        System.out.println("Weight: " +
ios.readDouble());
        System.out.println("Height: " +
ios.readInt());
        System.out.println("City: " +
ios.readUTF());
        System.out.println("ph_no: " +
ios.readUTF());
        ios.close();

        }catch(IOException e){
        System.out.println("Caught
Exception: " + e.getMessage());
        }
        finally{
        scanner.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 Main{
        public static void main(String[] args){
        String file_name = "demo.txt";
        int vowel_cnt = 0;
        int word_cnt = 0;
        int a_cnt = 0;
```

```
        try{
        BufferedReader br = new
BufferedReader(new
FileReader(file_name));
        String line;

        while((line = br.readLine()) != null){
        for(char ch:
line.toLowerCase().toCharArray()){
                if("aeiou".indexOf(ch) != -1)
                vowel_cnt++;

                if(ch == 'a')
                a_cnt++;
        }

        String[] words =
line.trim().split("\\s+");
        if(!line.trim().isEmpty())
                word_cnt += words.length;
        }
        br.close();
        System.out.println("Total vowels: " +
vowel_cnt);
        System.out.println("Total Words: " +
word_cnt);
        System.out.println("Occurance of a:
" + a_cnt);
        }catch(IOException e){
        System.out.println("Caught
exception: " + 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 ofthe 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 Main{
        public static void main(String[] args){
        if(args.length == 0){
        System.out.println("Please Provide
the File Name");
        return;
        }
        String file_name = args[0];
        File file = new File(file_name);
        Scanner scanner = new
Scanner(System.in);

        try{
        if(file.exists()){
                System.out.println("File
Exists. Displaying Contents:\n");
```

```
                BufferedReader br = new
BufferedReader(new FileReader(file));
                String line;
                while((line = br.readLine()) !=
null)
                System.out.println(line);
                br.close();

                System.out.print("\n Do you
want to add data to end of the file: ?
(yes/no): ");
                String response =
scanner.nextLine();


if(response.equalsIgnoreCase("yes")){
                BufferedWriter bw = new
BufferedWriter(new FileWriter(file, true));
                System.out.println("Enter
data for append (exit to finish)");

                while(true){
                String inp =
scanner.nextLine();

if(inp.equalsIgnoreCase("exit"))
                        break;
                bw.write(inp);
                bw.newLine();
                }
                bw.close();
                System.out.println("Data
appended Successfully");
                }
        }
        else{
                System.out.println("File
doesn't exits. Creating new file: " +
file_name);
                BufferedWriter bw = new
BufferedWriter(new FileWriter(file));
                System.out.println("Enter
data to write (type 'exit' to finish)");
                while(true){
                String inp =
scanner.nextLine();

if(inp.equalsIgnoreCase("exit"))
                break;
                bw.write(inp);
                bw.newLine();
                }
                bw.close();
        }
        }catch(IOException e){
        System.out.println("Caught
Exception: " + e.getMessage());
        }
        finally{
        scanner.close();
        }
        }
}
```