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

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class File {
    public static void main(String[] args) {
        // Replace this with the actual path of your file
        String filePath = "D:\\DYP\\Second Year\\Sem4\\AOOC\\Experiments\\Experiment 7\\Q1\\sample.txt";

        try (BufferedReader br = new BufferedReader(new FileReader(filePath))) {
            String line;
            System.out.println("Contents of the file:");
            while ((line = br.readLine()) != null) {
                System.out.println(line);
            }
        } catch (IOException e) {
            System.out.println("An error occurred while reading the file:");
            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 StudentDataIO {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Accept student information
        System.out.print("Enter Name: ");
        String name = scanner.nextLine();

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

        System.out.print("Enter Weight (kg): ");
        float weight = scanner.nextFloat();

        System.out.print("Enter Height (cm): ");
        float height = scanner.nextFloat();
        scanner.nextLine();  // consume newline

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

        System.out.print("Enter Phone Number: ");
        String phone = scanner.nextLine();

        // File to store the data
        String fileName = "studentdata.dat";

        // Writing to file using DataOutputStream
        try (DataOutputStream dos = new DataOutputStream(new FileOutputStream(fileName))) {
            dos.writeUTF(name);
            dos.writeInt(age);
            dos.writeFloat(weight);
            dos.writeFloat(height);
            dos.writeUTF(city);
            dos.writeUTF(phone);
            System.out.println("\nData written to file successfully.\n");
        } catch (IOException e) {
            System.out.println("Error writing to file: " + e.getMessage());
        }

        // Reading from file using DataInputStream
        try (DataInputStream dis = new DataInputStream(new FileInputStream(fileName))) {
            System.out.println("Reading data from file...");
            String readName = dis.readUTF();
            int readAge = dis.readInt();
            float readWeight = dis.readFloat();
            float readHeight = dis.readFloat();
            String readCity = dis.readUTF();
            String readPhone = dis.readUTF();

            // Displaying the data
            System.out.println("\n--- Student Information ---");
            System.out.println("Name   : " + readName);
            System.out.println("Age    : " + readAge);
            System.out.println("Weight : " + readWeight + " kg");
            System.out.println("Height : " + readHeight + " cm");
            System.out.println("City   : " + readCity);
            System.out.println("Phone  : " + readPhone);
        } catch (IOException e) {
            System.out.println("Error reading from file: " + e.getMessage());
        }
    }
}
```

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 FileAnalysis {
    public static void main(String[] args) {
        // Change this path to the actual file location
        String filePath = "D:\\DYP\\Second Year\\Sem4\\AOOC\\Experiments\\Experiment 7\\Q3\\sample.txt";

        int vowelCount = 0;
        int wordCount = 0;
        int aCount = 0;

        try (BufferedReader reader = new BufferedReader(new FileReader(filePath))) {
            String line;

            while ((line = reader.readLine()) != null) {
                // Count vowels and 'a'
                for (char ch : line.toLowerCase().toCharArray()) {
                    if ("aeiou".indexOf(ch) != -1) {
                        vowelCount++;
                    }
                    if (ch == 'a') {
                        aCount++;
                    }
                }

                // Count words
                String[] words = line.trim().split("\\s+");
                if (!line.trim().isEmpty()) {
                    wordCount += words.length;
                }
            }

            // Output the results
            System.out.println("Total number of vowels: " + vowelCount);
            System.out.println("Total number of words : " + wordCount);
            System.out.println("Occurrences of 'a'     : " + aCount);

        } catch (IOException e) {
            System.out.println("Error reading the file: " + e.getMessage());
        }
    }
}
```

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 Implement this program using character stream classes.

```
import java.io.*;
import java.util.Scanner;

public class FileEditor {
    public static void main(String[] args) {
        // Check if filename is provided via command line
        if (args.length == 0) {
            System.out.println("Usage: java FileEditor <filename>");
            return;
        }

        String filename = args[0];
        File file = new File(filename);
        Scanner scanner = new Scanner(System.in);

        // If file exists, display contents
        if (file.exists()) {
            System.out.println("\n--- File Contents ---");
            try (BufferedReader reader = new BufferedReader(new FileReader(file))) {
                String line;
                while ((line = reader.readLine()) != null) {
                    System.out.println(line);
                }
            } catch (IOException e) {
                System.out.println("Error reading file: " + e.getMessage());
            }

            // Ask user if they want to append
            System.out.print("\nDo you want to add data to the end of the file? (Yes/No): ");
            String response = scanner.nextLine();

            if (response.equalsIgnoreCase("Yes")) {
                appendToFile(file, scanner);
            } else {
                System.out.println("No changes made.");
            }

        } else {
            System.out.println("File does not exist. Creating new file...");
            appendToFile(file, scanner);
        }
    }

    // Method to accept user input and write to file
    private static void appendToFile(File file, Scanner scanner) {
        System.out.println("Enter data (type 'exit' on a new line to stop):");
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(file, true))) {
            while (true) {
                String input = scanner.nextLine();
                if (input.equalsIgnoreCase("exit")) {
                    break;
                }
                writer.write(input);
                writer.newLine();
            }
            System.out.println("Data saved to file: " + file.getName());
        } catch (IOException e) {
            System.out.println("Error writing to file: " + e.getMessage());
        }
    }
}
```