

Ripunjay Narula (19BCE0470)
JAVA PROGRAMMING LAB
Digital Assignment

File Handling

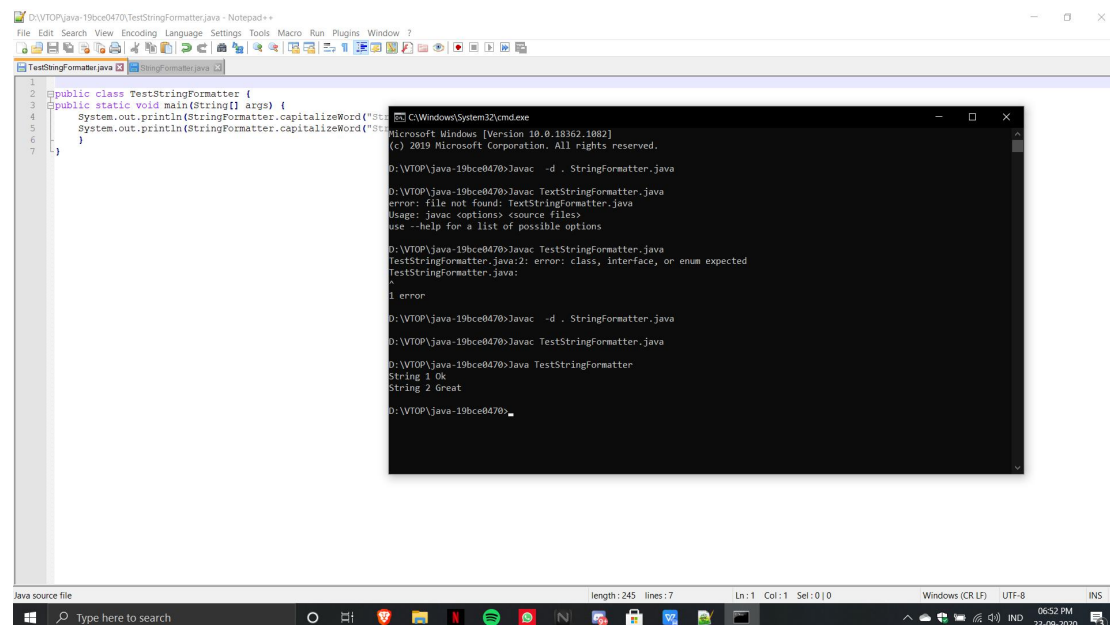
1)

StringFormatter.java:

```
public class StringFormatter {
    public static String capitalizeWord(String str){
        String words[]=str.split("\\s");
        String capitalizeWord="";
        for(String w:words){
            String first=w.substring(0,1);
            String afterfirst=w.substring(1);
            capitalizeWord+=first.toUpperCase()+" "+afterfirst+" ";
        }
        return capitalizeWord.trim();
    }
}
```

TestStringFormatter.java:

```
public class TestStringFormatter {
    public static void main(String[] args) {
        System.out.println(StringFormatter.capitalizeWord("String 1 ok"));
        System.out.println(StringFormatter.capitalizeWord("String 2 great"));
    }
}
```



2)

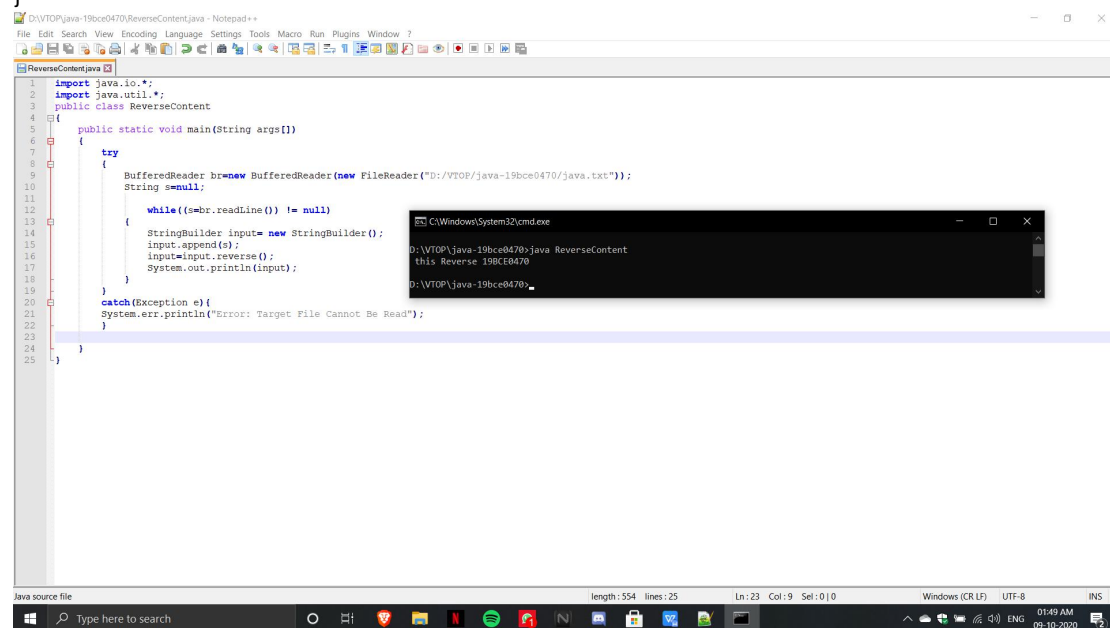
import java.io.*;

```

import java.util.*;
public class ReverseContent
{
    public static void main(String args[])
    {
        try
        {
            BufferedReader br=new BufferedReader(new
            FileReader("D:/VTOP/java-19bce0470/java.txt"));
            String s=null;

            while((s=br.readLine()) != null)
            {
                StringBuilder input= new StringBuilder();
                input.append(s);
                input=input.reverse();
                System.out.println(input);
            }
        }
        catch(Exception e){
            System.err.println("Error: Target File Cannot Be Read");
        }
    }
}

```



3)

```

import java.io.*;
import java.util.Scanner;
public class Employee implements Serializable
{
    String EmpId;
    String Name;
    float Salary;
    String Address;

    Employee(String EmpId, String Name, float Salary,String Address)

```

```

    {
        this.EmpId=EmpId;
        this.Name=Name;
        this.Salary=Salary;
        this.Address=Address;
    }
    public static void main(String args[])
    {
        try
        {
            Scanner input= new Scanner(System.in);
            System.out.println("Enter EmpId");
            String EmpId=input.nextLine();
            System.out.println("Enter Name");
            String Name=input.nextLine();
            System.out.println("Enter Address");
            String Address=input.nextLine();
            System.out.println("Enter Salary");
            float Salary=input.nextFloat();
            Employee e=new Employee(EmpId,Name,Salary,Address);
            FileOutputStream fout=new FileOutputStream("Employee.txt");
            ObjectOutputStream out=new ObjectOutputStream(fout);
            out.writeObject(e);
            out.flush();
            ObjectInputStream in=new ObjectInputStream(new FileInputStream("Employee.txt"));
            Employee read=(Employee)in.readObject();
            System.out.println(read.EmpId+" "+read.Name+" "+read.Salary+" "+read.Address);
            in.close();
        }
        catch(Exception e){
            System.err.println("Error: Target File Cannot Be Read");
        }
    }
}

```

The screenshot shows a Notepad++ window with the following Java code:

```

1 import java.io.*;
2 import java.util.Scanner;
3 public class Employee implements Serializable
4 {
5     String EmpId;
6     String Name;
7     float Salary;
8     String Address;
9
10    Employee(String EmpId, String Name, float Salary, String Address)
11    {
12        this.EmpId=EmpId;
13        this.Name=Name;
14        this.Salary=Salary;
15        this.Address=Address;
16    }
17    public static void main(String args[])
18    {
19        try
20        {
21            Scanner input= new Scanner(System.in);
22            System.out.println("Enter EmpId");
23            String EmpId=input.nextLine();
24            System.out.println("Enter Name");
25            String Name=input.nextLine();
26            System.out.println("Enter Address");
27            String Address=input.nextLine();
28            System.out.println("Enter Salary");
29            float Salary=input.nextFloat();
30            Employee e=new Employee(EmpId,Name,Salary,Address);
31            FileOutputStream fout=new FileOutputStream("Employee.txt");
32            ObjectOutputStream out=new ObjectOutputStream(fout);
33            out.writeObject(e);
34            out.flush();
35            ObjectInputStream in=new ObjectInputStream(new FileInputStream("Employee.txt"));
36            Employee read=(Employee)in.readObject();
37            System.out.println(read.EmpId+" "+read.Name+" "+read.Salary+" "+read.Address);
38            in.close();
39        }
40        catch(Exception e){
41            System.err.println("Error: Target File Cannot Be Read");
42        }
43    }
44 }

```

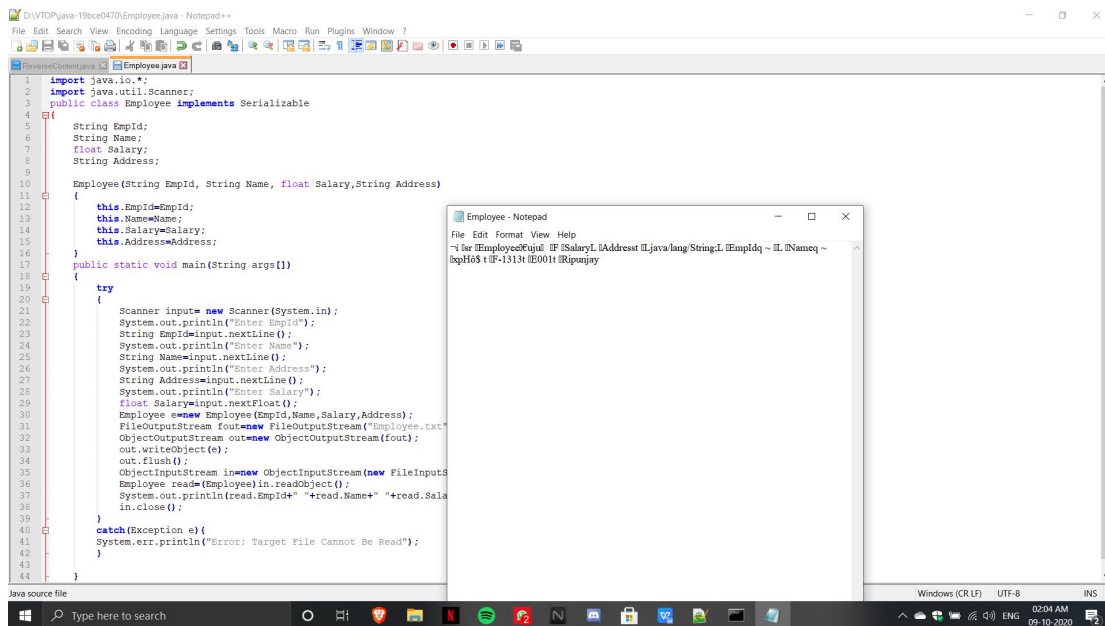
Overlaid on the code is a Windows command prompt window titled "C:\Windows\System32\cmd.exe". It shows the command being executed and its output:

```

D:\VTOP\Java-19bce0470>javac Employee.java
D:\VTOP\Java-19bce0470>java Employee
Enter EmpId
E001
Enter Name
Ripunjay
Enter Address
F-1313
Enter Salary
500000
E001 Ripunjay 500000.0 F-1313
D:\VTOP\Java-19bce0470>

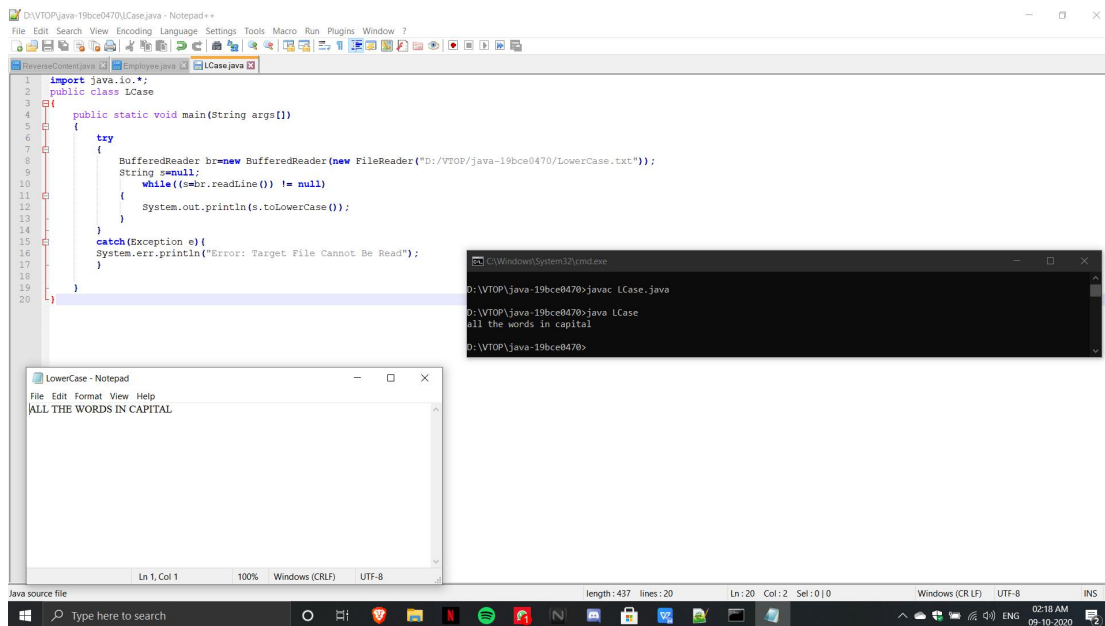
```

The status bar at the bottom of the Notepad++ window indicates the file is "Java source file", has a length of 1317, 45 lines, and is at line 45, column 2.



4)

```
import java.io.*;
public class LCase
{
    public static void main(String args[])
    {
        try
        {
            BufferedReader br=new BufferedReader(new
FileReader("D:/\TOP/java-19bce0470/LowerCase.txt"));
            String s=null;
            while((s=br.readLine()) != null)
            {
                System.out.println(s.toLowerCase());
            }
        }
        catch(Exception e){
            System.err.println("Error: Target File Cannot Be Read");
        }
    }
}
```

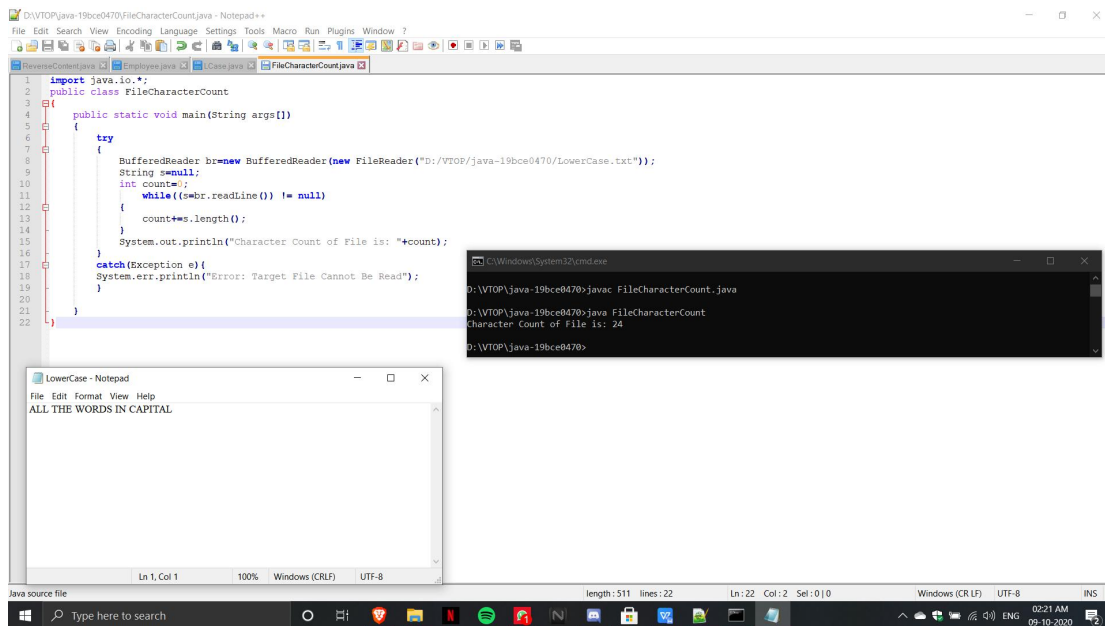


5)

```

import java.io.*;
public class FileCharacterCount
{
    public static void main(String args[])
    {
        try
        {
            BufferedReader br=new BufferedReader(new
            FileReader("D:/VTOP/java-19bce0470/LowerCase.txt"));
            String s=null;
            int count=0;
            while((s=br.readLine()) != null)
            {
                count+=s.length();
            }
            System.out.println("Character Count of File is: "+count);
        }
        catch(Exception e){
            System.err.println("Error: Target File Cannot Be Read");
        }
    }
}

```



6)

```

import java.io.*;
public class MergeLines
{
    public static void main(String args[])
    {
        try
        {
            BufferedReader br1=new BufferedReader(new
            FileReader("D:/VTOP/java-19bce0470/file1.txt"));
            BufferedReader br2=new BufferedReader(new
            FileReader("D:/VTOP/java-19bce0470/file2.txt"));
            BufferedWriter bw=new BufferedWriter(new
            FileWriter("D:/VTOP/java-19bce0470/file3.txt"));
            String s=null;
            while((s=br1.readLine()) != null)
            {
                bw.write(s);
            }
            while((s=br2.readLine()) != null)
            {
                bw.write(s);
            }
            br1.close();
            br2.close();
            bw.close();
            System.out.println("Success");
        }
        catch(Exception e){
            System.err.println("Error: Target File Cannot Be Read");
        }
    }
}

```

D:\VTOP\java-19bce0470\MergeLines.java - Notepad++

```
1 import java.io.*;
2 public class MergeLines
3 {
4     public static void main(String args[])
5     {
6         try
7         {
8             BufferedReader br1=new BufferedReader(new FileReader("D:/VTOP/java-19bce0470/file1.txt"));
9             BufferedReader br2=new BufferedReader(new FileReader("D:/VTOP/java-19bce0470/file2.txt"));
10            BufferedWriter bw=new BufferedWriter(new FileWriter("D:/VTOP/java-19bce0470/file3.txt"));
11            String s=null;
12            while((s=br1.readLine()) != null)
13            {
14                bw.write(s);
15            }
16            while((s=br2.readLine()) != null)
17            {
18                bw.write(s);
19            }
20            br1.close();
21            br2.close();
22            bw.close();
23            System.out.println("Success");
24        }
25        catch(Exception e){
26            System.err.println("Error: Target File Cannot Be Read");
27        }
28    }
29 }
30 }
```

C:\Windows\System32\cmd.exe

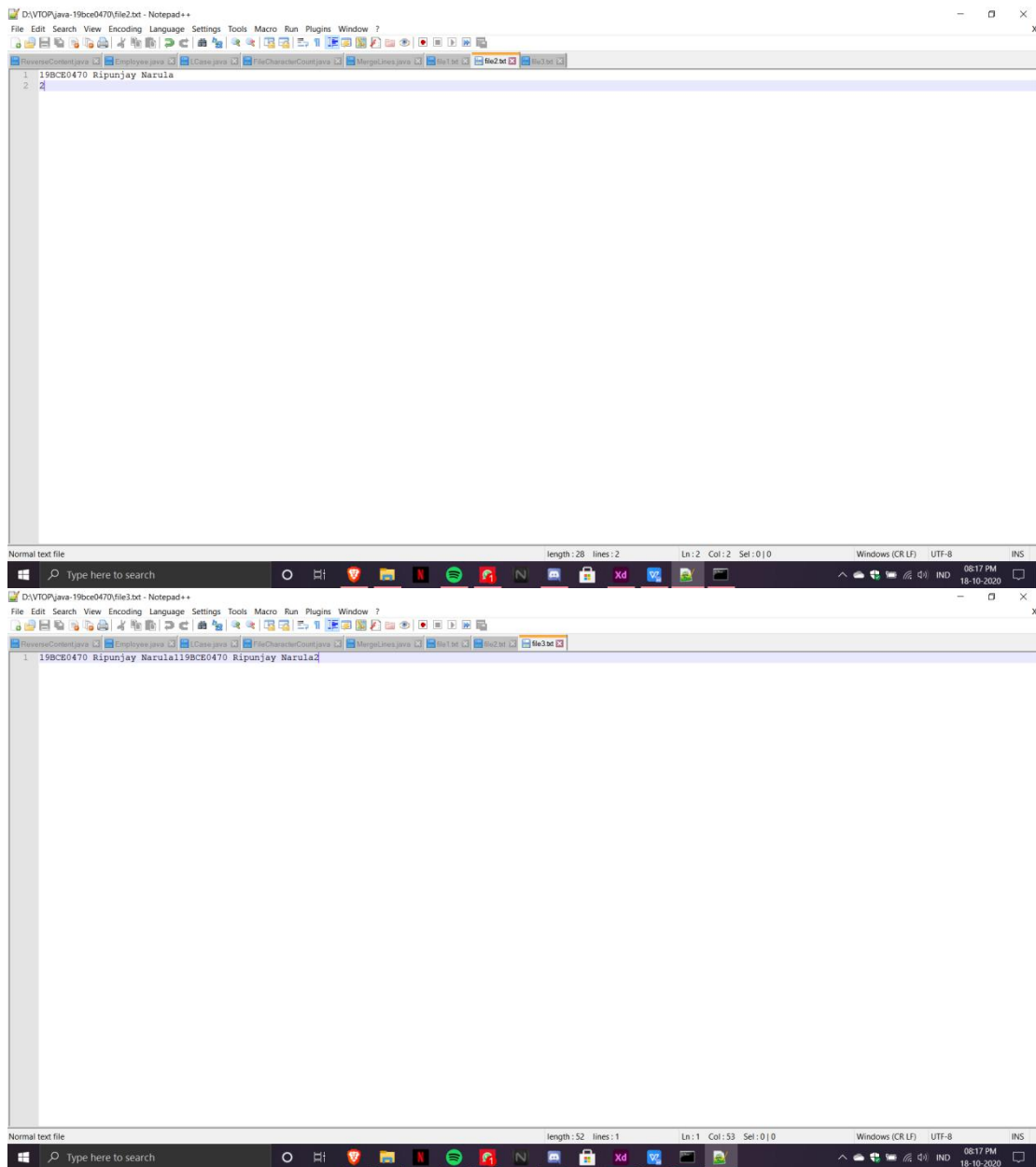
```
D:\VTOP\java-19bce0470>javac MergeLines.java
D:\VTOP\java-19bce0470>java MergeLines
Success
D:\VTOP\java-19bce0470>
```

Java source file length: 760 lines: 30 Ln: 16 Col: 50 Sel: 0|0 Windows (CR LF) UTF-8 INS 08:16 PM 18-10-2020

D:\VTOP\java-19bce0470\file1.txt - Notepad++

```
1 19bCE0470 Ripunjay Narula
2 1
```

Normal text file length: 28 lines: 2 Ln: 1 Col: 26 Sel: 0|0 Windows (CR LF) UTF-8 INS 08:16 PM 18-10-2020



```

7)
import java.io.*;
public class collectstats
{
    public static void main(String args[])
    {
        try
        {
            BufferedReader br=new BufferedReader(new
FileReader("D:/VTOP/java-19bce0470/statsfile.txt"));
            String s=null;
            int count=0;
            int countb=0;
            int countsemi=0;
            while((s=br.readLine()) != null)
            {
                if (s.trim().isEmpty())

```



```

        {
            countb++;
        }
        else if(s.endsWith(";"))
        {
            countsemi++;
        }
        else
        {
        }
        count++;
    }
    System.out.println("The Statistics are:");
    System.out.println(" Total Lines: "+count);
    System.out.println("no. of Blank SLines: "+countb);
    System.out.println("no. of Lines Ending with Semicolon: "+countsemi);

}
catch(Exception e){
System.err.println("Error: Target File Cannot Be Read");
}

}
}

```

The screenshot shows a Notepad++ window with a Java file named `collectstats.java`. The code is as follows:

```

1 import java.io.*;
2 public class collectstats
3 {
4     public static void main(String args[])
5     {
6         try
7         {
8             BufferedReader br=new BufferedReader(new FileReader("D:/VTOP/java-19bce0470/statsfile.txt"));
9             String s=null;
10            int count=0;
11            int countb=0;
12            int countsemi=0;
13            while((s=br.readLine()) != null)
14            {
15                if (s.trim().isEmpty())
16                {
17                    countb++;
18                }
19                else if(s.endsWith(";"))
20                {
21                    countsemi++;
22                }
23                else
24                {
25                    count++;
26                }
27            }
28            System.out.println("The Statistics are:");
29            System.out.println(" Total Lines: "+count);
30            System.out.println("no. of Blank SLines: "+countb);
31            System.out.println("no. of Lines Ending with Semicolon: "+countsemi);
32        }
33        catch(Exception e){
34            System.err.println("Error: Target File Cannot Be Read");
35        }
36    }
37 }
38 }
39 }

```

Overlaid on the code is a Windows Command Prompt window showing the execution of the program:

```

C:\Windows\System32\cmd.exe
1 error
D:\VTOP\java-19bce0470>javac collectstats.java
D:\VTOP\java-19bce0470>java collectstats
The Statistics are:
Total Lines: 4
no. of Blank SLines: 0
no. of Lines Ending with Semicolon: 2
D:\VTOP\java-19bce0470>

```

The status bar at the bottom of the Notepad++ window indicates the file is a Java source file, has a length of 846 bytes, 39 lines, and is currently at line 29, column 34.

D:\VTOP\java-19bce0470\statsfile.txt - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

ReverseContent.java Employee.java Case.java FileCharacterCount.java MergeLines.java File1.txt File2.txt File3.txt collectStats.java statsfile.txt

```
1 line number one
2 line two;
3 line three
4 line four;
```

length: 50 lines: 4 Ln: 4 Col: 11 Sel: 0 | 0 Windows (CR LF) UTF-8 INS

Type here to search 10:15 PM 18-10-2020

Serialisation and Collection

1)

```
import java.io.*;
import java.util.Scanner;
public class Employee implements Serializable
{
    String EmpId;
    String Name;
    float Salary;
    String Designation;

    Employee(String EmpId, String Name, float Salary,String Designation)
    {
        this.EmpId=EmpId;
        this.Name=Name;
        this.Salary=Salary;
        this.Designation=Designation;
    }
    public static void main(String args[])
    {
        try
        {
            Scanner input= new Scanner(System.in);
            FileOutputStream fout=new FileOutputStream("Employee_Data.txt");
            ObjectOutputStream out=new ObjectOutputStream(fout);
            Employee e[]=new Employee[3];
            for(int i=0;i<3;i++)
            {
                System.out.println("Enter EmpId: ");
                String EmpId=input.nextLine();
                System.out.println("Enter Name: ");
                String Name=input.nextLine();
                System.out.println("Enter Designation: ");
                String Designation=input.nextLine();
                System.out.println("Enter Salary: ");
                float Salary=input.nextFloat();
                input.nextLine();
                e[i]=new Employee(EmpId,Name,Salary,Designation);
                out.writeObject(e[i]);
            }
            out.flush();
            ObjectInputStream in = new ObjectInputStream(new
FileInputStream("Employee_Data.txt"));
            while (true) {
                try {
                    Employee read = (Employee) in.readObject();
                    if(read.Salary<50000){
                        System.out.println(read.EmpId+" "+read.Name+" "+read.Salary+"
"+read.Designation);
                    }

                } catch (EOFException exp) {
                    break;
                }
            }
        }
    }
}
```

```

    }

    in.close();

}

catch(Exception e){
    System.err.println("Error: Target File Cannot Be Read");
}

}

}

```

The screenshot shows a Notepad++ editor window with the following Java code:

```

1 import java.io.*;
2 import java.util.Scanner;
3 public class Employee implements Serializable
4 {
5     String EmpId;
6     String Name;
7     float Salary;
8     String Designation;
9
10    Employee(String EmpId, String Name, float Salary, String Designation)
11    {
12        this.EmpId=EmpId;
13        this.Name=Name;
14        this.Salary=Salary;
15        this.Designation=Designation;
16    }
17    public static void main(String args[])
18    {
19        try
20        {
21            Scanner input= new Scanner(System.in);
22            FileOutputStream fout=new FileOutputStream("Employee_Data.ser");
23            ObjectOutputStream out=new ObjectOutputStream(fout);
24            Employee e[]=new Employee[10];
25            for(int i=0;i<10;i++)
26            {
27                System.out.println("Enter EmpId: ");
28                String EmpId=input.nextLine();
29                System.out.println("Enter Name: ");
30                String Name=input.nextLine();
31                System.out.println("Enter Designation: ");
32                String Designation=input.nextLine();
33                System.out.println("Enter Salary: ");
34                float Salary=input.nextFloat();
35                input.nextLine();
36                e[i]=new Employee(EmpId,Name,Salary,Designation);
37                out.writeObject(e[i]);
38            }
39            out.flush();
40            ObjectInputStream in = new ObjectInputStream(new FileInputStream("Employee_Data.ser"));
41            while (true) {
42                try {
43                    Employee read = (Employee) in.readObject();
44                    if(read.Salary<50000){

```

Overlaid on the bottom right is a Windows command prompt window titled "C:\Windows\System32\cmd.exe". It shows the execution of the Java program:

```

D:\VTOP\java-19bce0470>java Employee
Enter EmpId
88002
Enter Name
Ripunjay
Enter Designation
Manager
Enter Salary
87654
Enter EmpId
88003
Enter Name
ABC
Enter Designation
Intern
Enter Salary
46543.0
88003 ABC 46543.0 Intern
D:\VTOP\java-19bce0470>

```

2)

```

import java.io.*;
import java.util.Scanner;
class Loan implements Serializable
{
    String CustomerName;
    String Address;
    int Age;
    float Salary;
    float LoanAmount;
    String LoanType;
    Loan(String CustomerName,String Address,int Age,float Salary,float LoanAmount,String
LoanType)
    {
        this.CustomerName=CustomerName;
        this.Address=Address;
        this.Age=Age;
        this.Salary=Salary;
        this.LoanAmount=LoanAmount;
        this.LoanType=LoanType;
    }
    public static void main(String args[])
    {
        try{
            Scanner input= new Scanner(System.in);

```

```

FileOutputStream fout=new FileOutputStream("Loan.txt");
ObjectOutputStream out=new ObjectOutputStream(fout);
Loan l[]= new Loan[3];
for(int i=0;i<3;i++)
{
    System.out.println("Enter Customer's Name: ");
    String CustomerName= input.nextLine();
    System.out.println("Enter Address: ");
    String Address= input.nextLine();
    System.out.println("Enter Loan Type: ");
    String LoanType=input.nextLine();
    System.out.println("Enter Age: ");
    int Age=input.nextInt();
    System.out.println("Enter Salary: ");
    float Salary= input.nextFloat();
    System.out.println("Enter Loan Amount: ");
    float LoanAmount= input.nextFloat();
    input.nextLine();
    l[i]= new Loan(CustomerName,Address,Age,Salary,LoanAmount,LoanType);
    out.writeObject(l[i]);
}
out.flush();
ObjectInputStream in = new ObjectInputStream(new FileInputStream("Loan.txt"));
while (true) {
    try {
        Loan read = (Loan) in.readObject();
        if(!((read.CustomerName).matches("[A-Za-z]+")))
        {
            System.out.println("Name Invalid");
        }
        else if(read.Age<18)
        {
            System.out.println("Invalid! Minimum Age not met");
        }
        else if(read.Salary<20000)
        {
            System.out.println("Invalid! Minimum Salary not met");
        }
        else if((read.Salary*100000)/2162>=read.LoanAmount)
        {
            System.out.println("Invalid! Exceeds eligible loan amount");
        }
        else
        if(((read.LoanType).equals("housing"))| |((read.LoanType).equals("vehicle"))| |((read.LoanType).equals("personal"))))
        {
            System.out.println("Invalid Loan Type");
        }
        else{
            System.out.println("Valid");
        }
    } catch (EOFException exp) {
        break;
    }
}

in.close();

```

```

    }
    catch(Exception e)
    {
        e.printStackTrace();
        System.err.println("Error: Target File Cannot Be Read");
    }
}
}
}

```

The screenshot shows a Java IDE with a file named `Loan.java` and its execution output in a command prompt window.

Loan.java Code:

```

1 import java.io.*;
2 import java.util.Scanner;
3 class Loan implements Serializable
4 {
5     String CustomerName;
6     String Address;
7     int Age;
8     float Salary;
9     float LoanAmount;
10    String LoanType;
11    Loan(String CustomerName,String Address,int Age,float Salary,float
12    {
13        this.CustomerName=CustomerName;
14        this.Address=Address;
15        this.Age=Age;
16        this.Salary=Salary;
17        this.LoanAmount=LoanAmount;
18        this.LoanType=LoanType;
19    }
20    public static void main(String args[])
21    {
22        try{
23            Scanner input= new Scanner(System.in);
24            FileOutputStream fout=new FileOutputStream("Loan.txt");
25            ObjectOutputStream out=new ObjectOutputStream(fout);
26            Loan l[]= new Loan[5];
27            for(int i=0;i<5;i++)
28            {
29                System.out.println("Enter Customer's Name: ");
30                String CustomerName= input.nextLine();
31                System.out.println("Enter Address: ");
32                String Address= input.nextLine();
33                System.out.println("Enter Loan Type: ");
34                String LoanType=input.nextLine();
35                System.out.println("Enter Age: ");
36                int Age=input.nextInt();
37                System.out.println("Enter Salary: ");
38                float Salary= input.nextFloat();
39                System.out.println("Enter Loan Amount: ");
40                float LoanAmount= input.nextFloat();
41                input.nextLine();
42                l[i]= new Loan(CustomerName,Address,Age,Salary,LoanAmount);
43                out.writeObject(l[i]);
44            }
45        }
46        catch(Exception e)
47        {
48            e.printStackTrace();
49            System.err.println("Error: Target File Cannot Be Read");
50        }
51    }
52 }

```

Command Prompt Output:

```

D:\VTOP\Java-19bce0470>javac Loan.java
D:\VTOP\Java-19bce0470>java Loan
Enter Customer's Name:
Ripunjay
Enter Address:
abc,cde,Vellore
Enter Loan Type:
Vehicle
Enter Age:
19
Enter Salary:
98765
Enter Loan Amount:
2000000
Enter Customer's Name:
abc
Enter Address:
CDE
Enter Loan Type:
personal
Enter Age:
25
Enter Salary:
50000
Enter Loan Amount:
987654
Enter Customer's Name:
CDE
Enter Address:
RFG
Enter Loan Type:
Housing
Enter Age:
28
Enter Salary:
87654
Enter Loan Amount:
2000000
Invalid! Exceeds eligible loan amount
Invalid! Exceeds eligible loan amount
Invalid Loan Type
D:\VTOP\Java-19bce0470>

```

```

3)
import java.util.*;
class StudArrayList
{
    String StudentID;
    String Name;
    int marks1;
    int marks2;
    int marks3;
    StudArrayList(String StudentID, String Name, int marks1,int marks2,int marks3)
    {
        this.StudentID=StudentID;
        this.Name=Name;
        this.marks1=marks1;
        this.marks2=marks2;
        this.marks3=marks3;
    }

    public static void main(String args[])
    {
        Scanner input= new Scanner(System.in);
        StudArrayList s[]= new StudArrayList[5];
        ArrayList<StudArrayList> alist= new ArrayList<StudArrayList>();
        for(int i=0;i<5;i++)
        {
            System.out.println("Enter StudentID: ");
            String StudentID=input.nextLine();
            System.out.println("Enter Name: ");

```

```

        String Name=input.nextLine();
        System.out.println("Enter marks for subject 1: ");
        int marks1=input.nextInt();
        System.out.println("Enter marks for subject 2: ");
        int marks2=input.nextInt();
        System.out.println("Enter marks for subject 3: ");
        int marks3=input.nextInt();
        input.nextLine();
        s[i]=new StudArrayList(StudentID,Name,marks1,marks2,marks3);
        alist.add(s[i]);
    }
    for(StudArrayList l: alist)
    {
        System.out.println("Student StudentID: "+l.StudentID);
        System.out.println("Student Name: "+l.Name);
        System.out.println("Student marks 1: "+l.marks1);
        System.out.println("Student marks 2: "+l.marks2);
        System.out.println("Student marks 3: "+l.marks3);
    }
}
}

```

The screenshot displays a Java IDE with two windows. The left window, titled 'D:\VTOP\java-19bce0470\StudArrayList.java - Notepad++', shows the source code of the 'StudArrayList' class. The code includes a constructor that initializes student details and a 'main' method that uses a 'Scanner' to collect input for three students, storing them in an 'ArrayList' of 'StudArrayList' objects. The right window, titled 'C:\Windows\System32\cmd.exe', shows the program's execution. It prompts the user to enter student details (Name, StudentID, and marks for three subjects) for three different students. The output shows the collected data for each student, such as 'Student StudentID: s01', 'Student Name: Ripunjay', and 'Student marks 1: 90'.

```

D:\VTOP\java-19bce0470\StudArrayList.java - Notepad++
1  import java.util.*;
2  class StudArrayList
3  {
4      String StudentID;
5      String Name;
6      int marks1;
7      int marks2;
8      int marks3;
9      StudArrayList(String StudentID, String Name, int marks1,int marks2,int
10     {
11         this.StudentID=StudentID;
12         this.Name=Name;
13         this.marks1=marks1;
14         this.marks2=marks2;
15         this.marks3=marks3;
16     }
17
18     public static void main(String args[])
19     {
20         Scanner input= new Scanner(System.in);
21         StudArrayList s[]= new StudArrayList[3];
22         ArrayList<StudArrayList> alist= new ArrayList<StudArrayList>();
23         for(int i=0;i<3;i++)
24         {
25             System.out.println("Enter StudentID: ");
26             String StudentID=input.nextLine();
27             System.out.println("Enter Name: ");
28             String Name=input.nextLine();
29             System.out.println("Enter marks for subject 1: ");
30             int marks1=input.nextInt();
31             System.out.println("Enter marks for subject 2: ");
32             int marks2=input.nextInt();
33             System.out.println("Enter marks for subject 3: ");
34             int marks3=input.nextInt();
35             input.nextLine();
36             s[i]=new StudArrayList(StudentID,Name,marks1,marks2,marks3);
37             alist.add(s[i]);
38         }
39         for(StudArrayList l: alist)
40         {
41             System.out.println("Student StudentID: "+l.StudentID);
42             System.out.println("Student Name: "+l.Name);
43             System.out.println("Student marks 1: "+l.marks1);
44             System.out.println("Student marks 2: "+l.marks2);
45         }
46     }
47 }

```

```

C:\Windows\System32\cmd.exe
Enter Name:
DEF
Enter marks for subject 1:
85
Enter marks for subject 2:
84
Enter marks for subject 3:
83
Enter StudentID:
s01
Enter Name:
Ripunjay
Enter marks for subject 1:
90
Enter marks for subject 2:
98
Enter marks for subject 3:
97
Student StudentID: s01
Student Name: Ripunjay
Student marks 1: 90
Student marks 2: 98
Student marks 3: 97
Student StudentID: s02
Student Name: Aditya
Student marks 1: 95
Student marks 2: 94
Student marks 3: 93
Student StudentID: s03
Student Name: ABC
Student marks 1: 65
Student marks 2: 64
Student marks 3: 63
Student StudentID: s04
Student Name: DEF
Student marks 1: 85
Student marks 2: 84
Student marks 3: 83
Student StudentID: s05
Student Name: 75
Student marks 1: 74
Student marks 2: 73
Student marks 3: 72
D:\VTOP\java-19bce0470>

```

JDBC

1.1. Write a Java program to perform the following (i) Create a FACULTY table with the following fields empid, facultyname, DOB, Dateofjoining and designation. (ii) Insert 5 faculty details on to the table (iii) Retrieve all the faculty details. (iv) Retrieve the faculty information whose designation is Senior professor

```
mysql> describe employee;
```

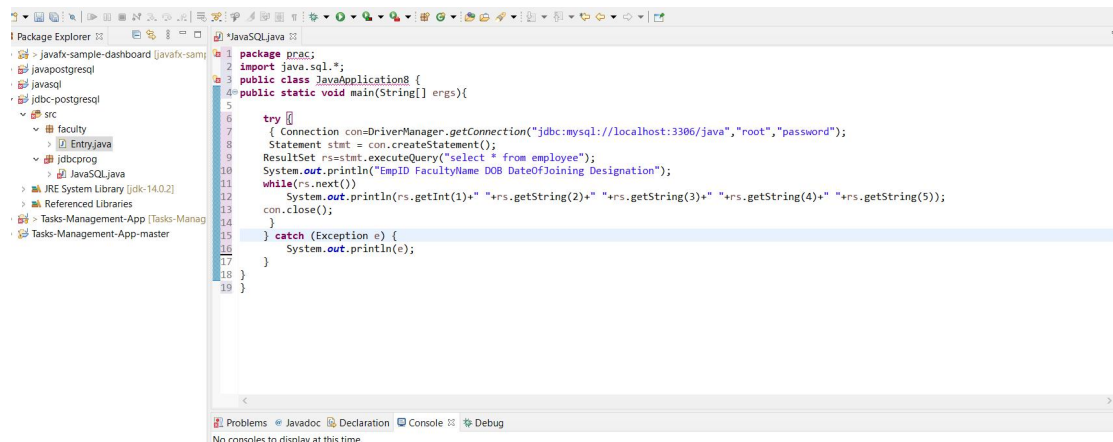
Field	Type	Null	Key	Default	Extra
empid	varchar(10)	YES		NULL	
facultyname	varchar(20)	YES		NULL	
dob	date	YES		NULL	
joinDate	date	YES		NULL	
designation	varchar(30)	YES		NULL	

```
5 rows in set (0.02 sec)
```

```
mysql> select*from employee;
```

empid	facultyname	dob	joinDate	designation
1004	Travis Scott	1939-04-01	2020-02-24	Senior professor
1005	Justin Biber	1995-05-14	2020-05-01	Junior professor
1003	Eminem	1987-03-01	2020-04-19	Junior professor
1001	Halsey	1990-05-19	2020-06-21	Associate professor
1006	Ripunjay Narula	2000-06-25	2018-04-12	Senior professor

```
5 rows in set (0.01 sec)
```



```
deps-jar:
Updating property file: C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\build-jar.properties
Compiling 1 source file to C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\classes
compile-single:
run-single:
EmpID FacultyName DOB DateOfJoining Designation
1004 Travis Scott 1939-04-01 2020-02-24 Senior professor
1005 Justin Biber 1995-05-14 2020-05-01 Junior professor
1003 Eminem 1987-03-01 2020-04-19 Junior professor
1001 Halsey 1990-05-19 2020-06-21 Associate professor

BUILD SUCCESSFUL (total time: 4 seconds)
```



```
*JavaSQL.java
1 package prac;
2 import java.sql.*;
3 //198CE470
4 public class JavaApplication8 {
5     public static void main(String[] args){
6
7         try {
8             Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/java","root","password");
9             Statement stmt = con.createStatement();
10            ResultSet rs=stmt.executeQuery("select * from employee where designation='senior professor'");
11            System.out.println("EmpID FacultyName DOB DateOfJoining Designation");
12            while(rs.next())
13            {
14                System.out.println(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3)+" "+rs.getString(4)+" "+rs.getString(5));
15            }
16        } catch (Exception e) {
17            System.out.println(e);
18        }
19    }
20 }
```

```
ant -f C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build-internal\action.name=run.single -pjavac
init:
Deleting: C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\build-jar.properties
Compiling 1 source file to C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\classes
compile-single:
run-single:
EmpID FacultyName DOB DateOfJoining Designation
1004 Travis Scott 1939-04-01 2020-02-24 Senior professor
BUILD SUCCESSFUL (total time: 4 seconds)
```

Servlet Code

1. Design a Signup form using HTML and write a servlet code to print welcome message only if the password and confirm password are same otherwise print Password mismatch message in the same HTML form.

HTML FILE:

```
<html>
<body>
<form action="servlet/Register" method="post">
Name:<input type="text" name="userName"/><br/><br/>
Email Id:<input type="text" name="userEmail"/><br/><br/>
Password:<input type="password" name="userPass"/><br/><br/>
Confirm_Password:<input type="password" name="userPass_c"/><br/><br/>

<input type="submit" value="register"/>

</form>
</body>
</html>
```

JAVA FILE:

```
import java.io.*;
import java.sql.*;
import javax.servlet.ServletException;
import javax.servlet.http.*;
```

```
public class Register extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
```

```
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
```

```
        String n=request.getParameter("userName");
        String e=request.getParameter("userEmail");
        String p=request.getParameter("userPass");
        String p=request.getParameter("userPass_c");
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            Connection con=DriverManager.getConnection(
                "jdbc:oracle:thin:@localhost:1521:xe","system","oracle");
```

```
            PreparedStatement ps=con.prepareStatement(
                "insert into registeruser values(?,?,?,?)");
```

```
            ps.setString(1,n);
            ps.setString(2,p);
            ps.setString(3,e);
            ps.setString(4,c);
```

```
            int i=ps.executeUpdate();
            if(i>0)
                out.print("Welcome");
```

```
        }catch (Exception e2) {System.out.println("Password mismatch");}
```

```
        out.close();
    }
}
```

```

1  import java.io.*;
2  import java.sql.*;
3  import javax.servlet.ServletException;
4  import javax.servlet.http.*;
5
6  public class Register extends HttpServlet {
7      public void doPost(HttpServletRequest request, HttpServletResponse response)
8          throws ServletException, IOException {
9
10         response.setContentType("text/html");
11         PrintWriter out = response.getWriter();
12
13         String n=request.getParameter("userName");
14         String e=request.getParameter("userEmail");
15         String p=request.getParameter("userPass");
16         String p=request.getParameter("userPass_c");
17         try{
18             Class.forName("oracle.jdbc.driver.OracleDriver");
19             Connection con=DriverManager.getConnection(
20                 "jdbc:oracle:thin:@localhost:1521:xe","system","oracle");
21
22             PreparedStatement ps=con.prepareStatement(
23                 "insert into registeruser values(?,?,?,?)");
24
25             ps.setString(1,n);
26             ps.setString(2,p);
27             ps.setString(3,e);
28             ps.setString(4,c);
29
30             int i=ps.executeUpdate();
31             if(i>0)
32                 out.print("Welcome");
33
34         }catch (Exception e2) {System.out.println("Password mismatch");}
35
36         out.close();
37     }
38 }
39
40

```