Ripunjay Narula (19BCE0470)

JAVA PROGRAMMING LAB

Digital Assignment

File Handling

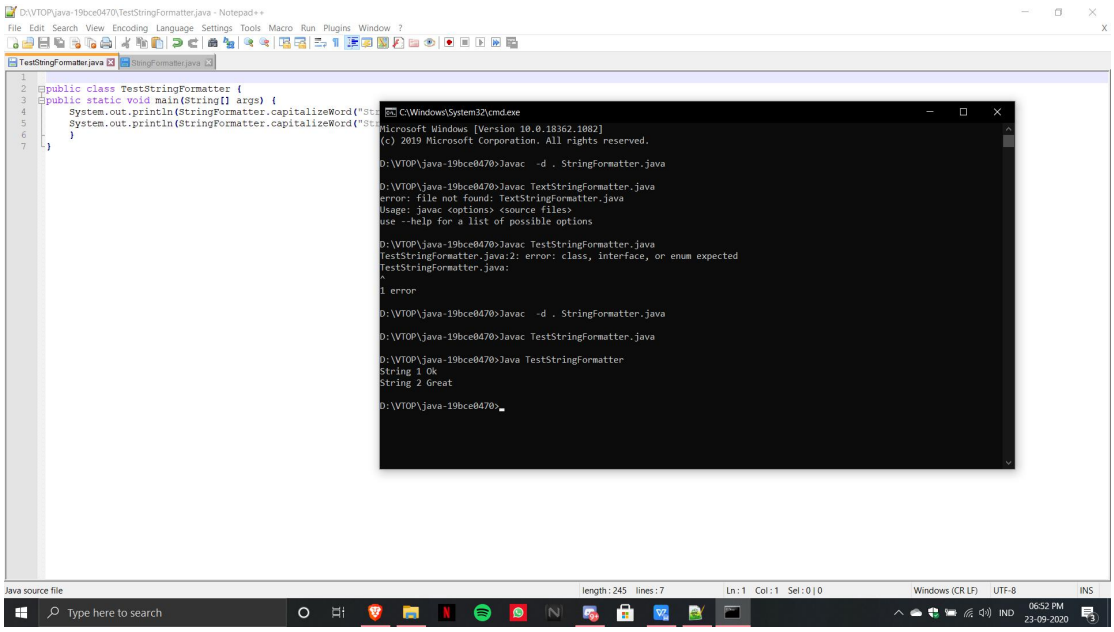
1)

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:

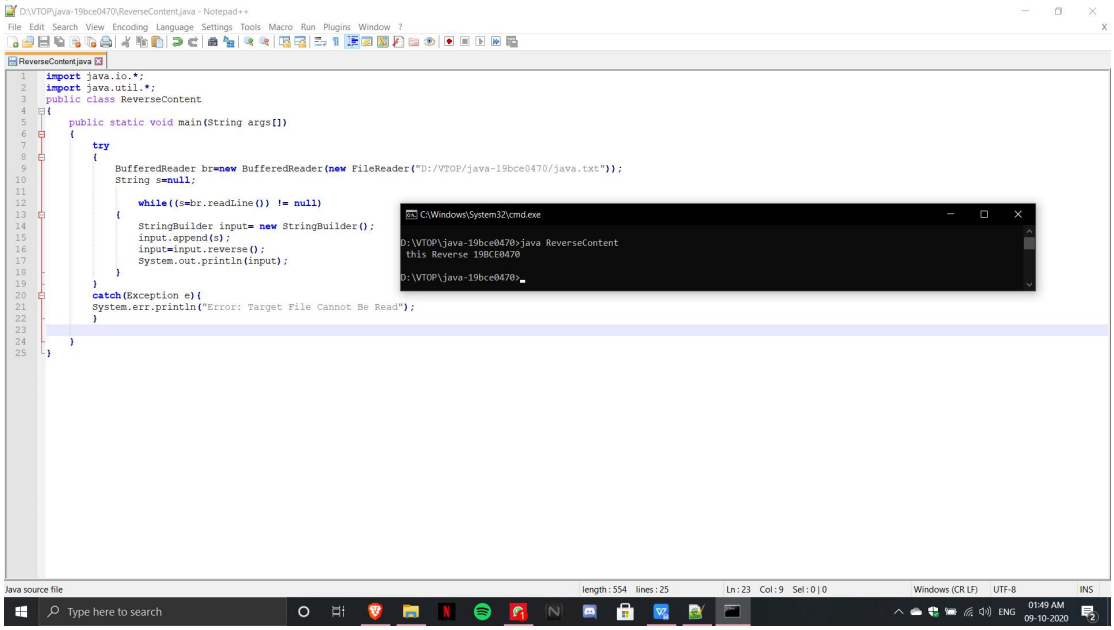
```
public class TestStringFormatter {
public static void main(String[] args) {
	System.out.println(StringFormatter.capitalizeWord("String 1 ok"));
	System.out.println(StringFormatter.capitalizeWord("String 2 great"));
	}
}
```



2)

```
import java.io.*;
```

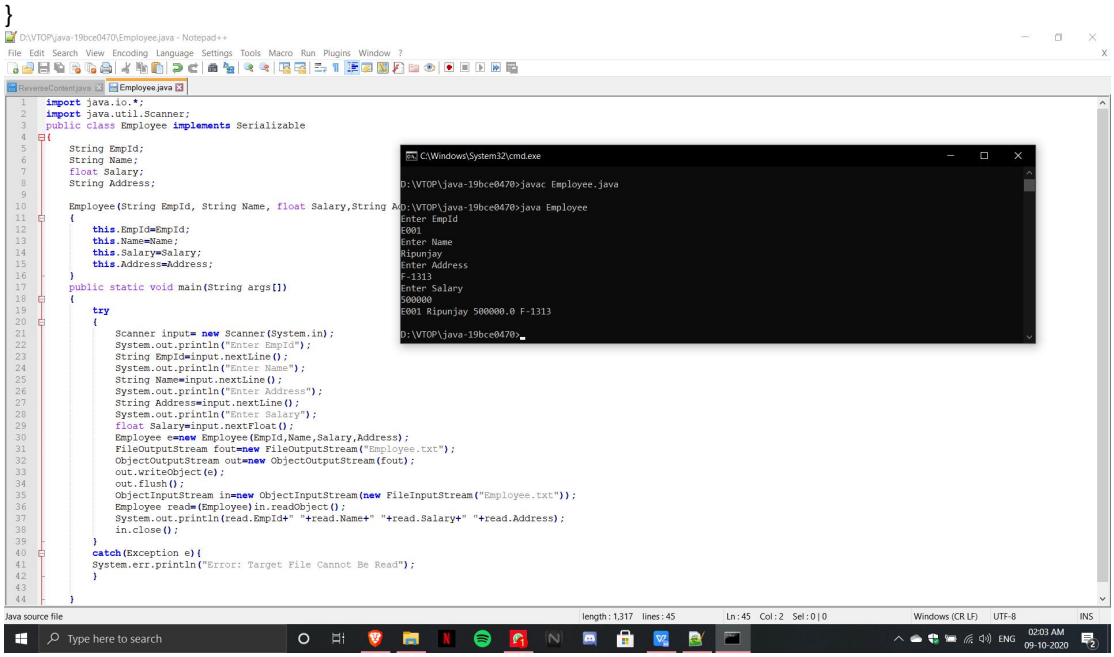
```
import java.util.*;
public class ReverseContent
{
	public static void main(String args[])
	{
		try
		{
			BufferedReader br=new BufferedReader(new
FileReader("D:/VTOP/java-19bce0470/java.txt"));
			String s=null;

				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");
		}

	}
}
```

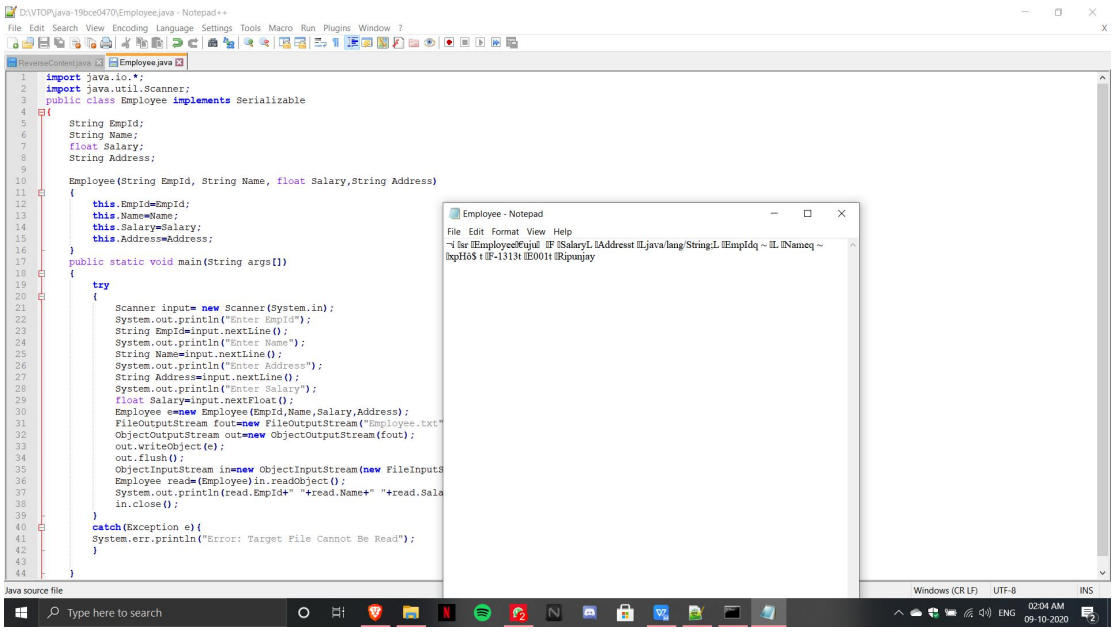


3)

```
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)
```

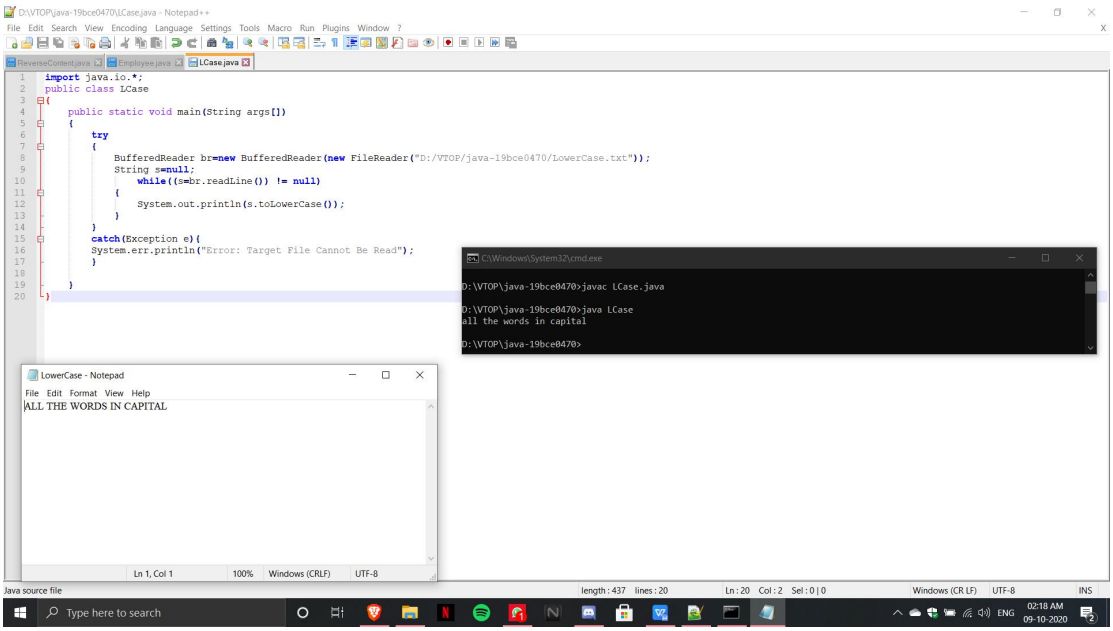
```
{
        this.EmpId=EmpId;
        this.Name=Name;
        this.Salary=Salary;
        this.Address=Address;
}
public static void main(String args[])
{
        try
        {
                Scanner input= new Scanner(System.in);
                System.out.println("Enter EmpId");
                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");
        }

}
}
```





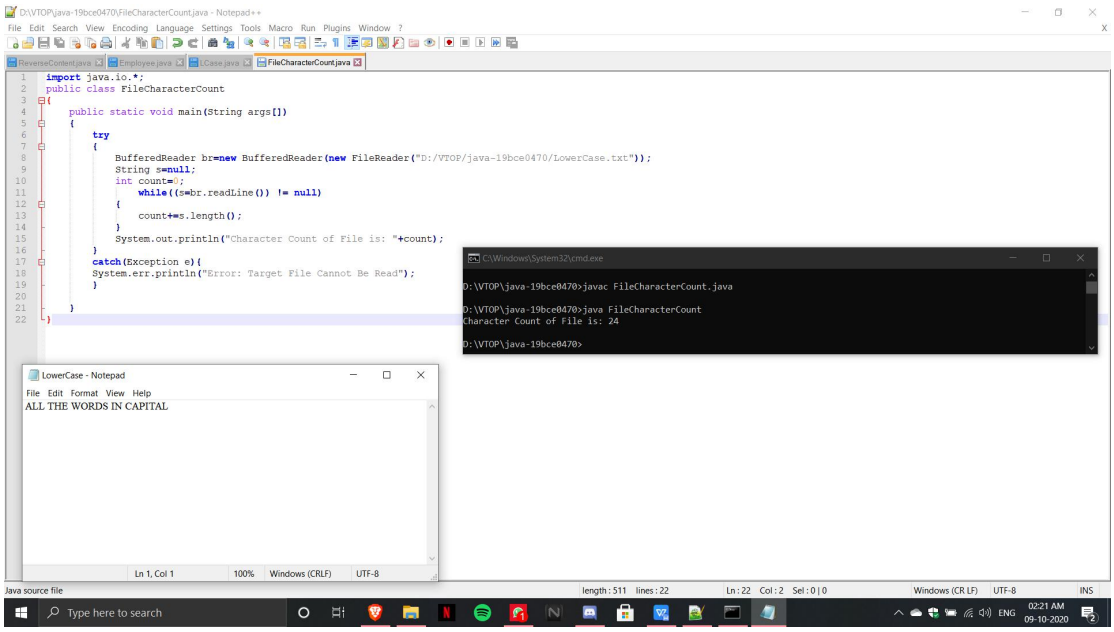
4)

```
import java.io.*;
public class LCase
{
	public static void main(String args[])
	{
		try
		{
			BufferedReader br=new BufferedReader(new
FileReader("D:/VTOP/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");
		}

	}
}
```



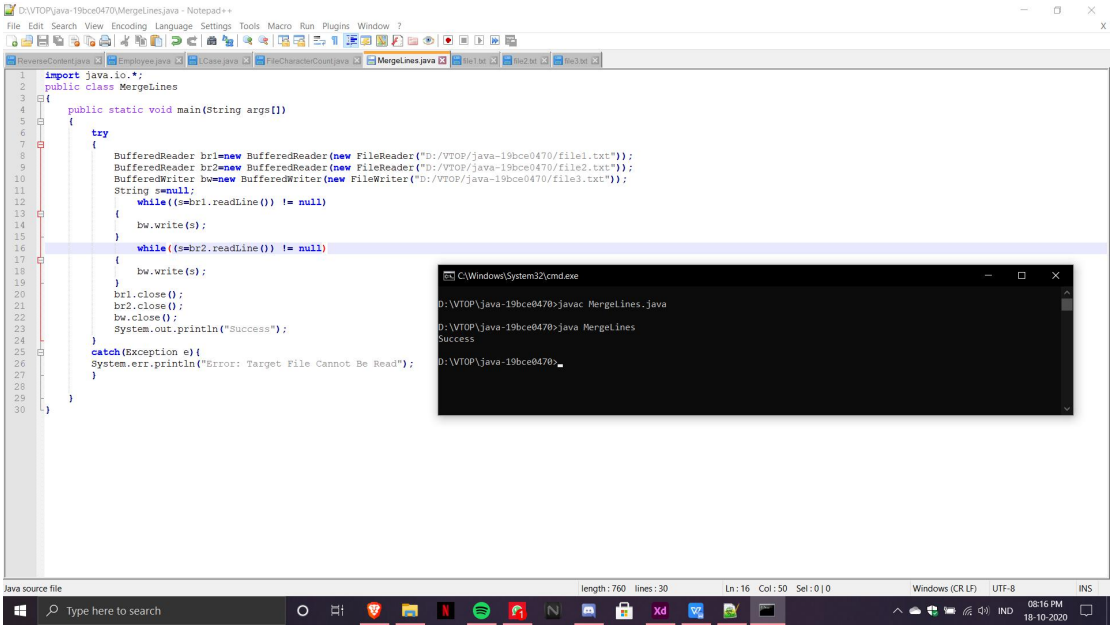
5)

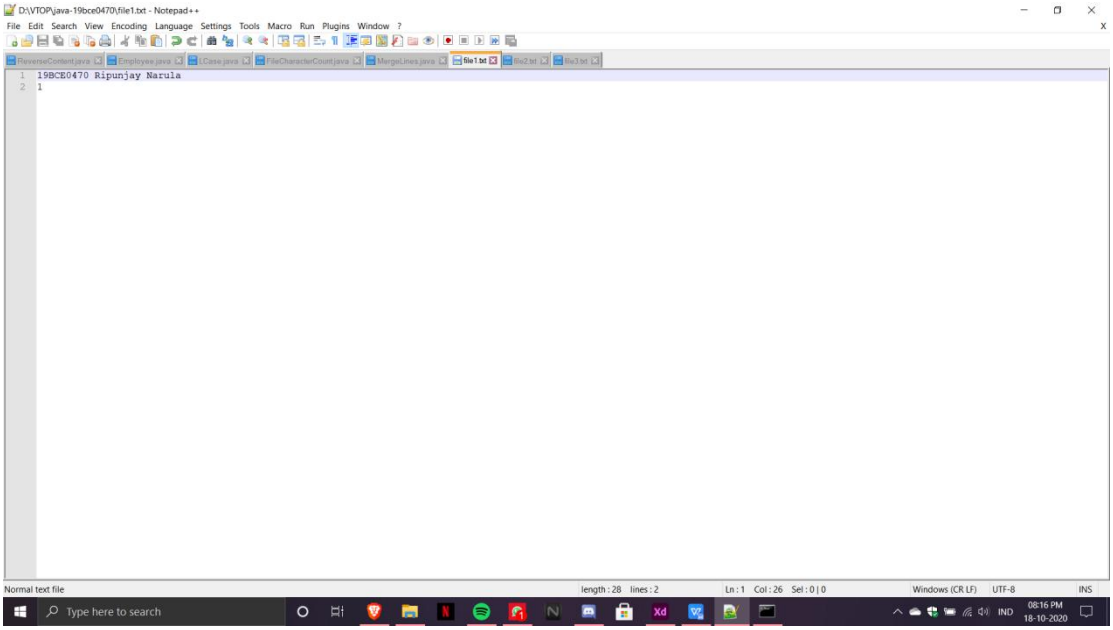
```
import java.io.*;
public class FileCharacterCount
{
	public static void main(String args[])
	{
		try
		{
			BufferedReader br=new BufferedReader(new
FileReader("D:/VTOP/java-19bce0470/LowerCase.txt"));
			String s=null;
			int count=0;
				while((s=br.readLine()) != null)
			{
				count+=s.length();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

	}
}
```

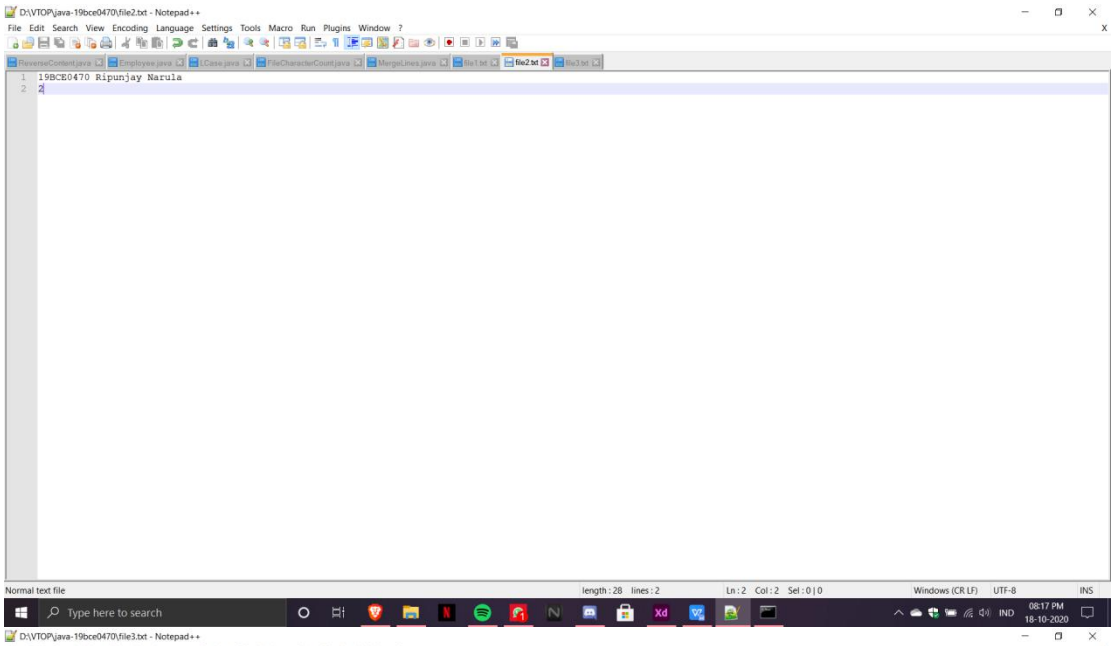


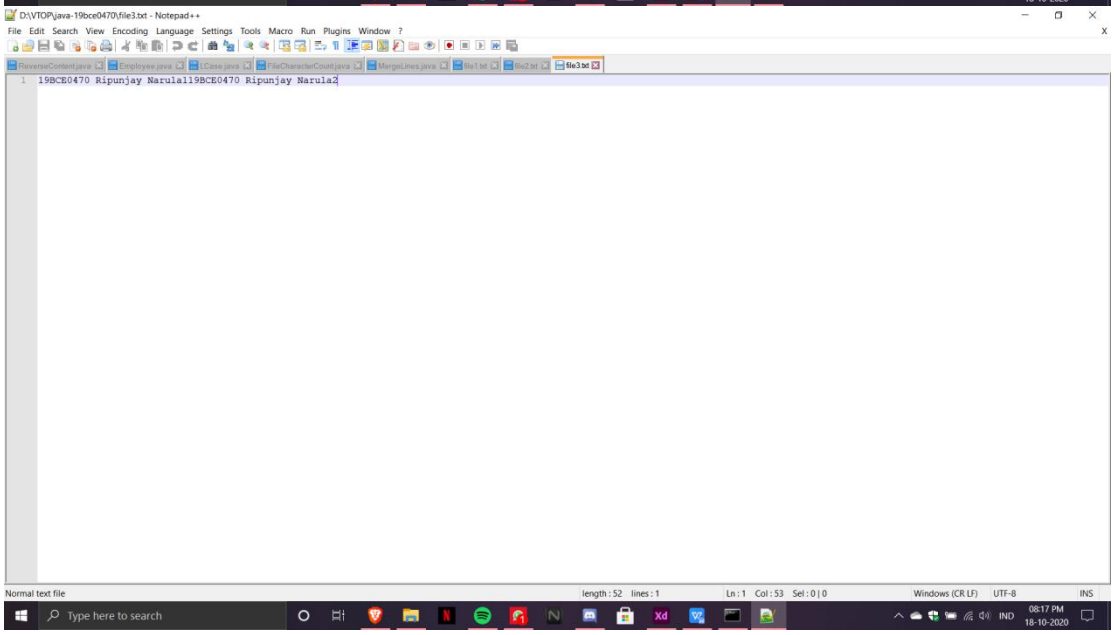
6)

```
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++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
ReverseContent.java Employee.java LCase.java FileCharacterCount.java MergeLines.java file1.txt file2.txt file3.txt
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");
}
}
}
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
Type here to search
IND
08:16 PM
18-10-2020


D:\VTOP\java-19bce0470\file1.txt - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
19BCE0470 Ripunjay Narula
1
Normal text file
length : 28 lines : 2
Ln : 1 Col : 26 Sel : 0 | 0
Windows (CR LF)
UTF-8
INS
Type here to search
IND
08:16 PM
18-10-2020

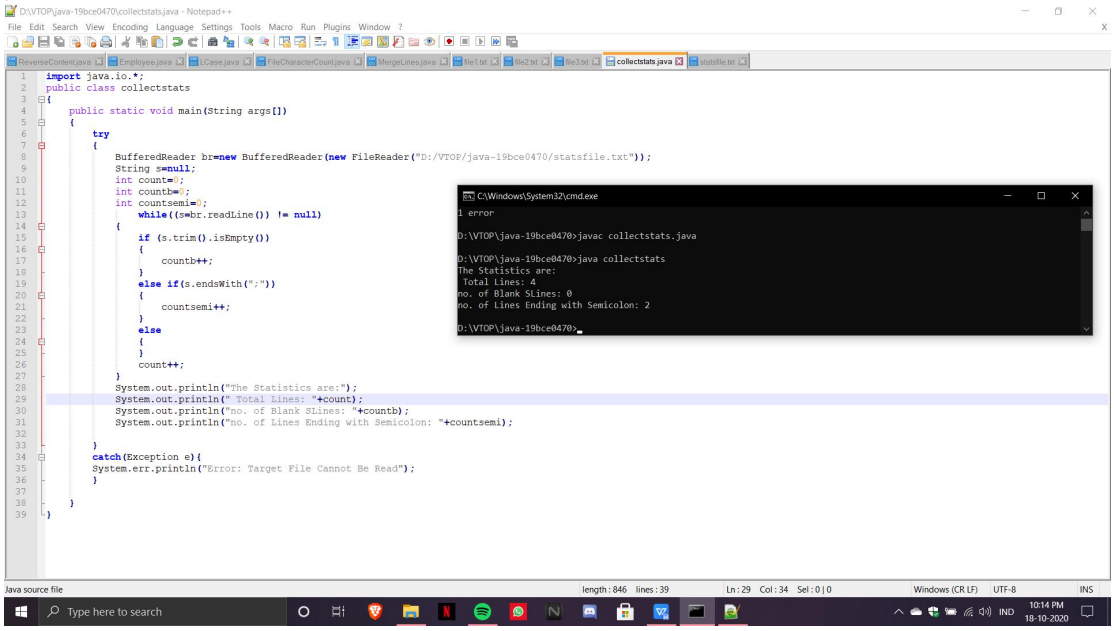


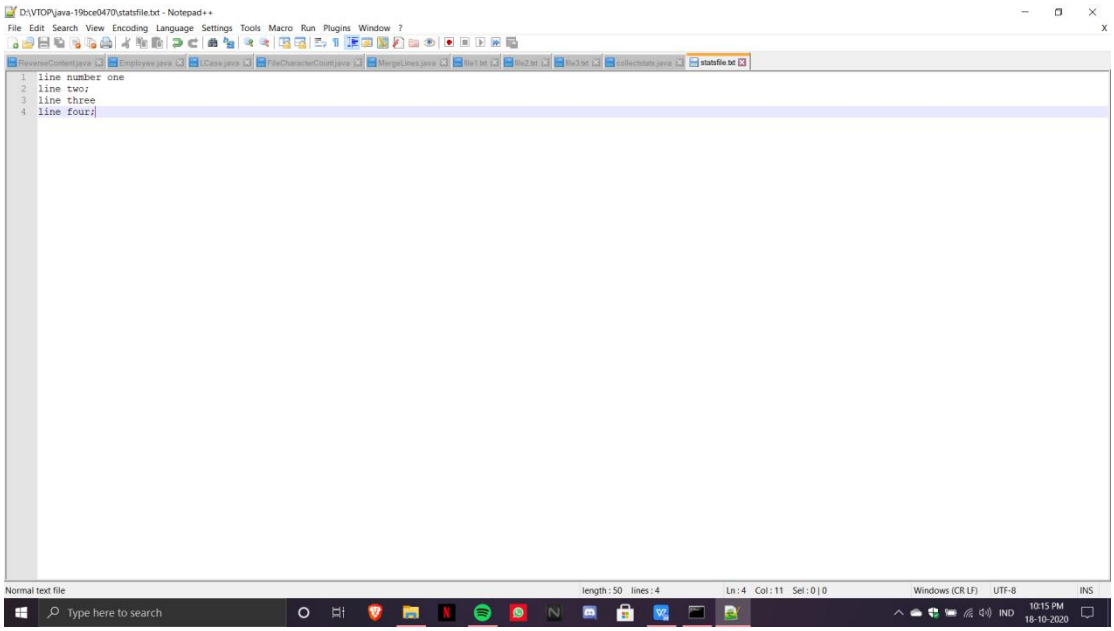


7)

```
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");
        }

    }
}
```



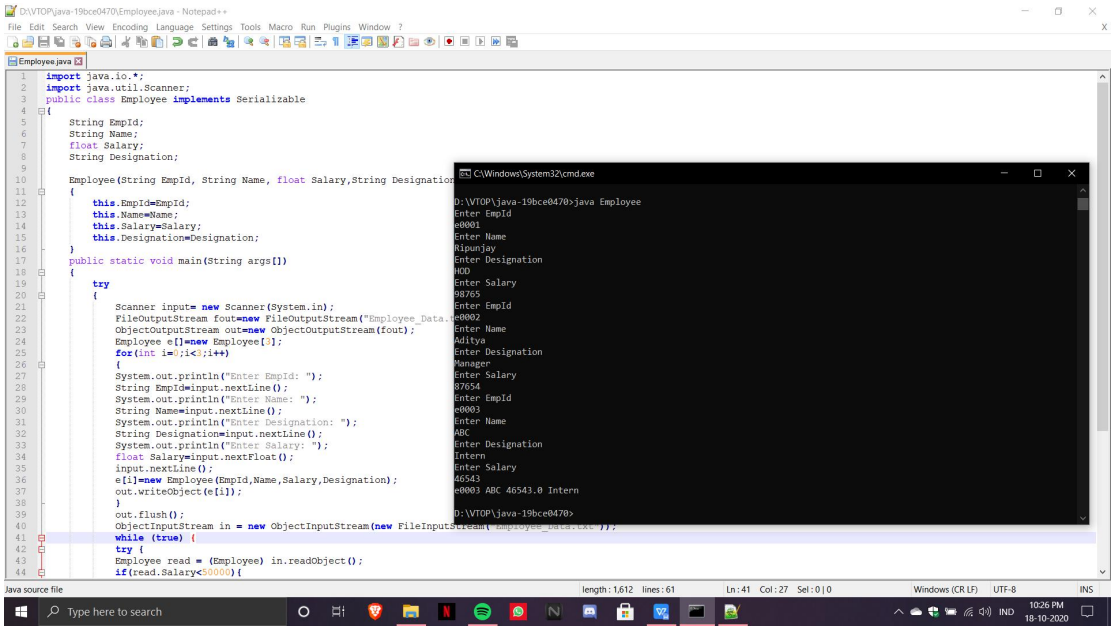

D:\VTOP\java-19bce0470\statsfile.txt - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
ReverseContent.java
Employee.java
LCase.java
FileCharacterCount.java
MergeLines.java
file1.txt
file2.txt
file3.txt
collectstats.java
statsfile.txt
1 line number one
2 line two;
3 line three
4 line four;
Normal text file
length : 50 lines : 4
Ln : 4 Col : 11 Sel : 0 | 0
Windows (CR LF)
UTF-8
INS
Type here to search
IND
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");
            }
        }
}
```

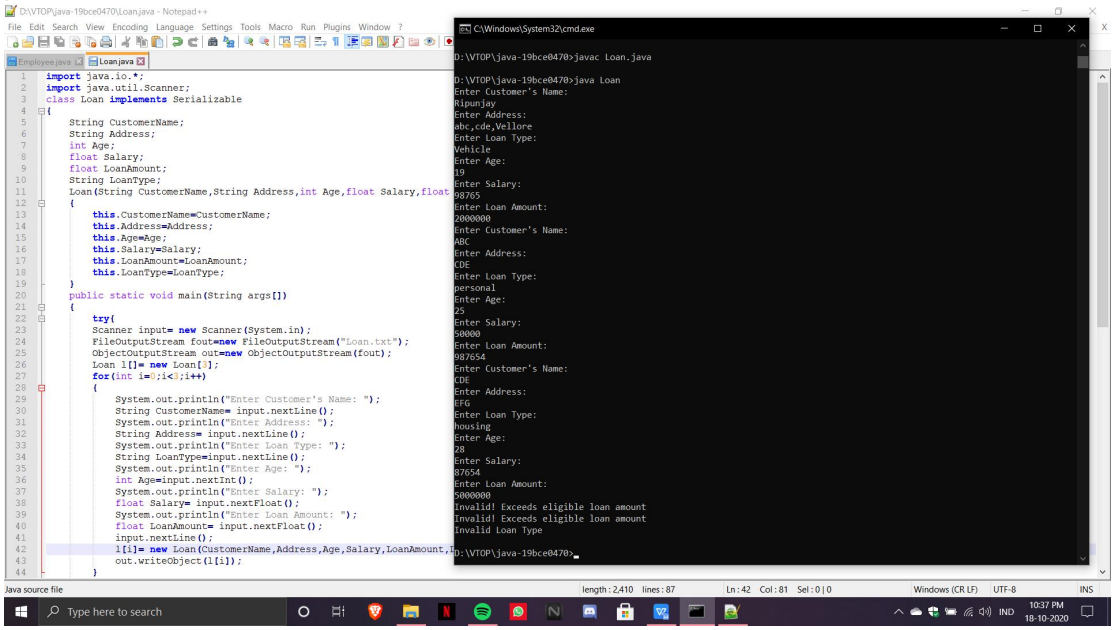


2)

```
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).equal
s("personal")))
	{
		System.out.println("Invalid Loan Type");
	}
	else{
		System.out.println("Valid");
	}
	} catch (EOFException exp) {
	break;
	}
}

in.close();
```

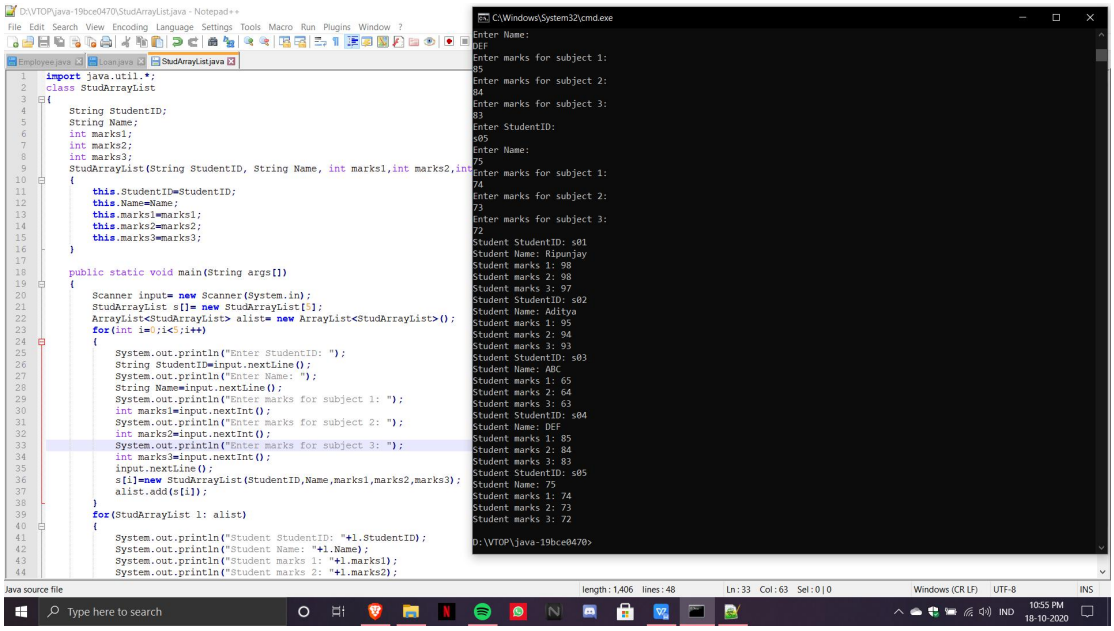
```
            }
            catch(Exception e)
            {
                  e.printStackTrace();
                  System.err.println("Error: Target File Cannot Be Read");
            }

      }
}
```



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);
        }
    }
}
```

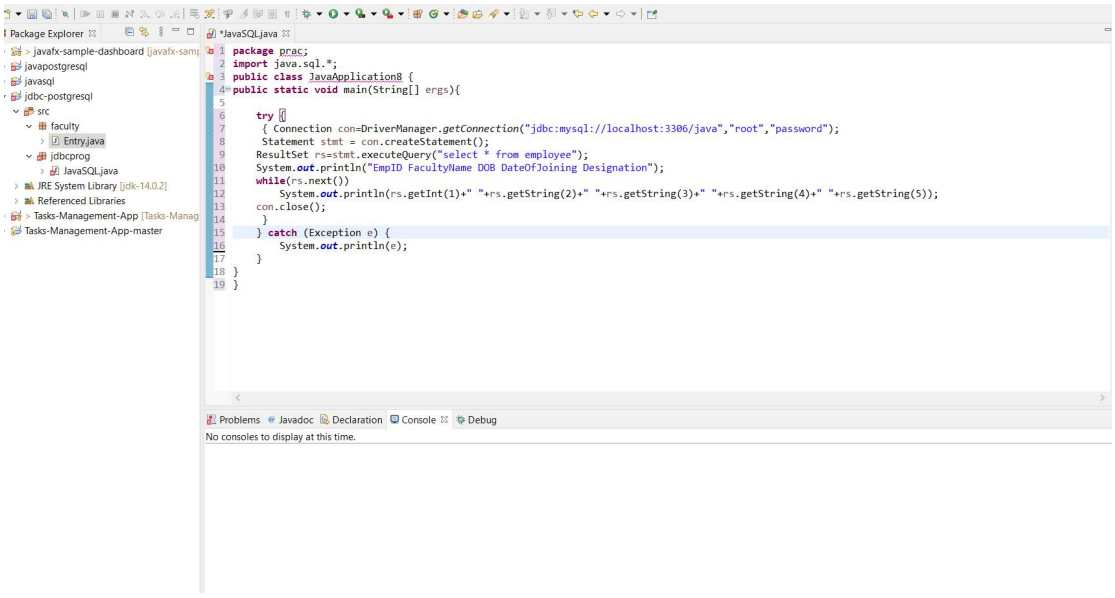


JDBC

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)
```

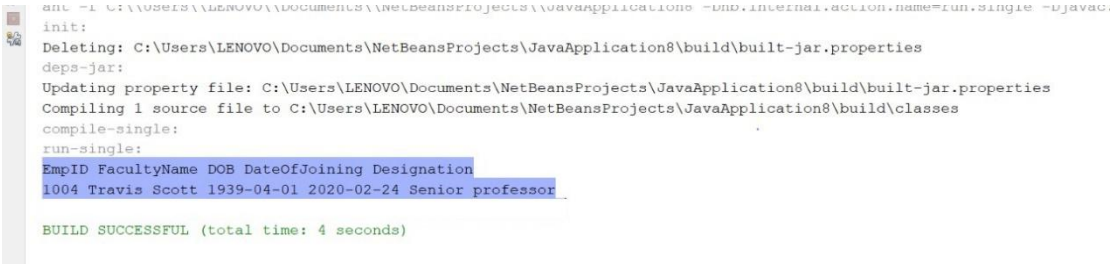



```
deps-jar:
Updating property file: C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\built-jar.properties
Compiling 1 source file to C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\classes
compile-single:
run-single:
EmpID FacultyName DOB DateOfJoining Designation
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
package prac;
import java.sql.*;
//19BCE470
public class JavaApplication8 {
public static void main(String[] ergs){

    try {
     { Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/java","root","password");
     Statement stmt = con.createStatement();
     ResultSet rs=stmt.executeQuery("select * from employee where designation='senior professor'");
     System.out.println("EmpID FacultyName DOB DateOfJoining Designation");
     while(rs.next())
         System.out.println(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3)+" "+rs.getString(4)+" "+rs.getString(5));
     con.close();
     }
    } catch (Exception e) {
       System.out.println(e);
    }
}
}
```

Problems Javadoc Declaration Console Debug

No consoles to display at this time.



```
init:
Deleting: C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\built-jar.properties
deps-jar:
Updating property file: C:\Users\LENOVO\Documents\NetBeansProjects\JavaApplication8\build\built-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();
}

}
```