# JAVA LAB ASSESSEMENT 4

RegNo: 19BCE2099

Name:Akansha Saji

**File Handling**

**Q1. Java Program to Replace First Letter of Every Word with Capital Letter.**

**Program:**

import java.io.\*;

public class FirstLetterCaptial

{

public static void main(String args[])

{

try

{

BufferedReader br=new BufferedReader(new FileReader("D:\\CapLetter.txt"));

String s=null;

while((s=br.readLine()) != null)

{

String cap = s.substring(0, 1).toUpperCase() + s.substring(1);

System.out.println(cap);

}

}

catch(Exception e){

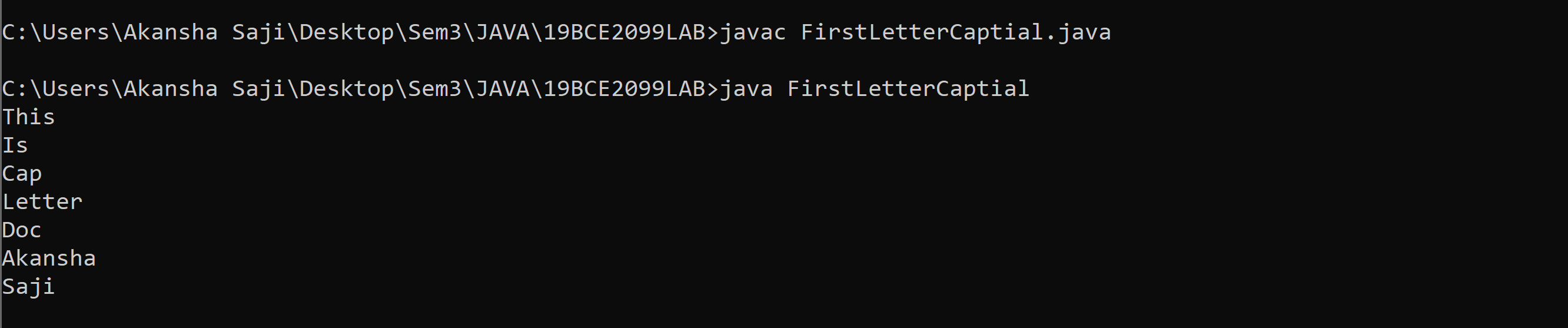
System.err.println("Error: Target File Cannot Be Read");

}

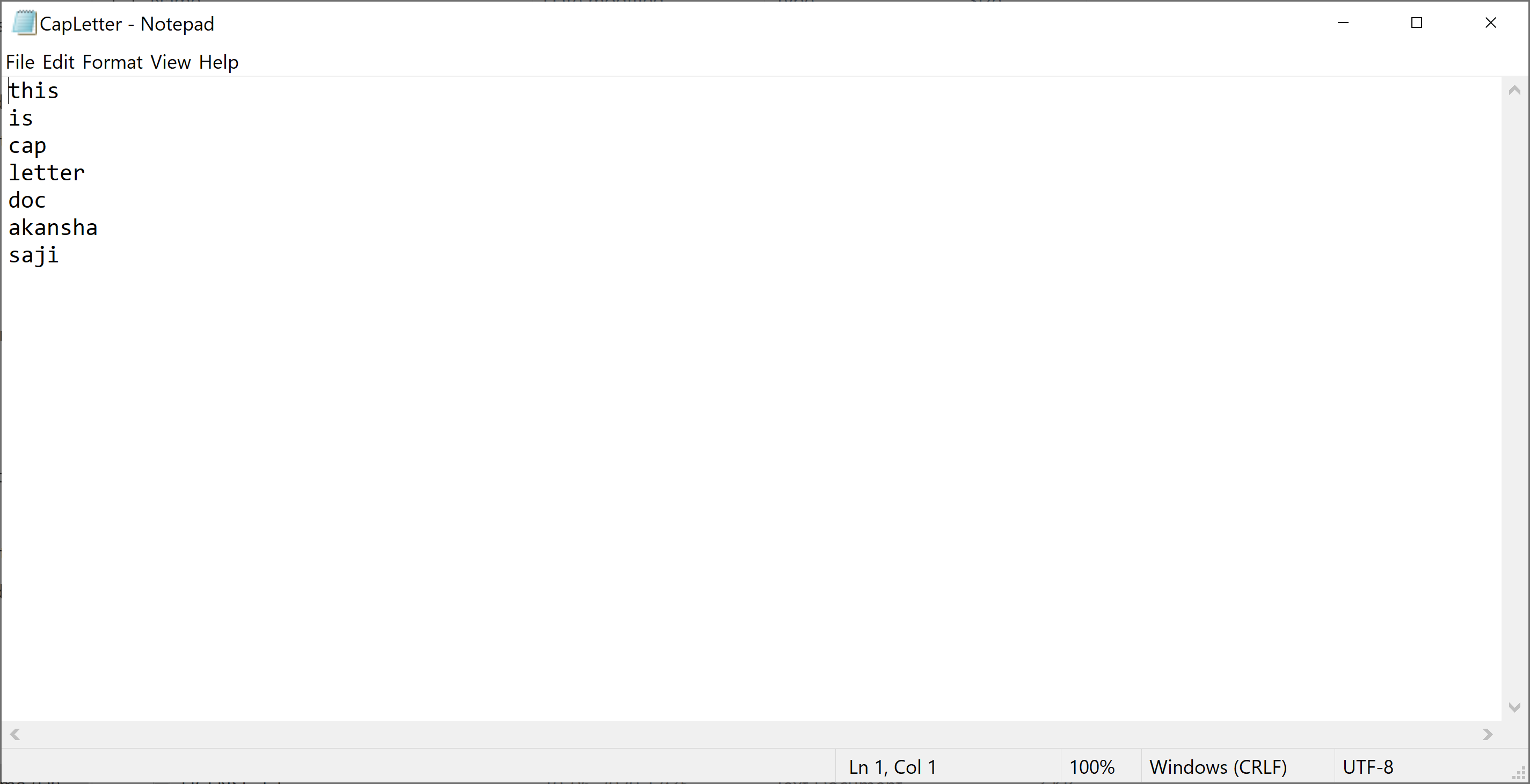
}

}

**Output**:



**CapLetter.txt:**



**Q2. Java Program to Reverse the Contents of a File and Print it.**

**Program:**

import java.io.\*;

import java.util.\*;

public class ReverseContent

{

public static void main(String args[])

{

try

{

BufferedReader br=new BufferedReader(new FileReader("D:\\RevContent.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){

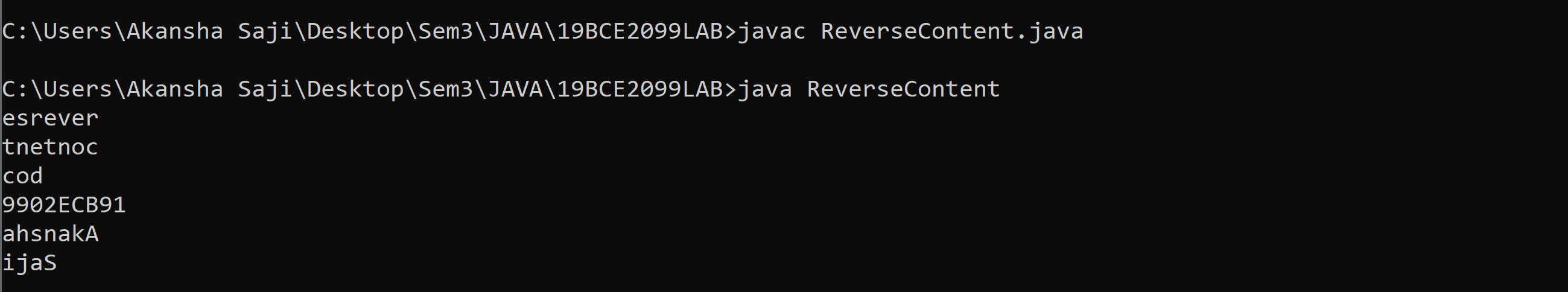
System.err.println("Error: Target File Cannot Be Read");

}

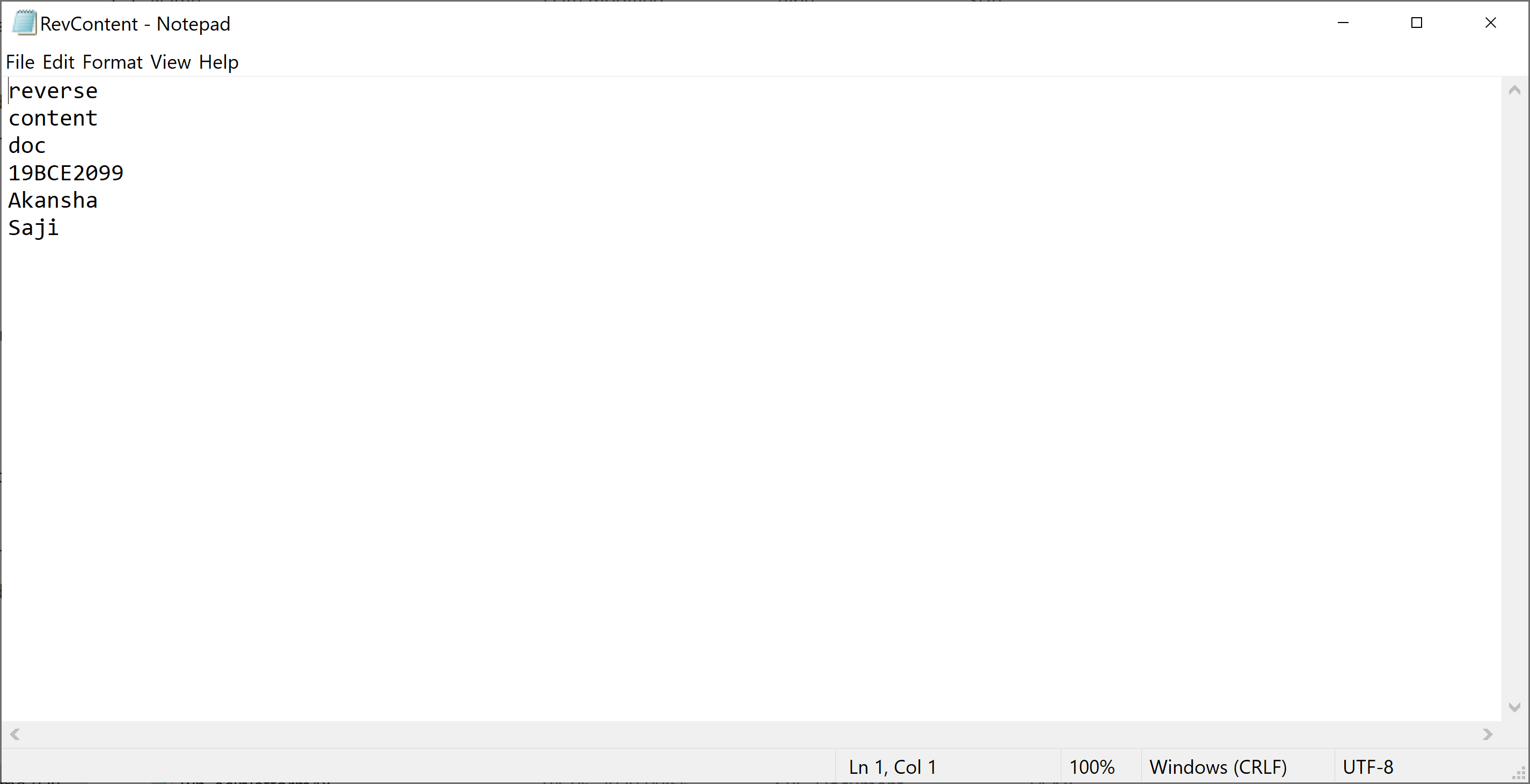
}

}

**Output:**



**RevContent.txt:**



**Q3. Java Program to Update Details of Employee Using Files.**

**Program**:

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){

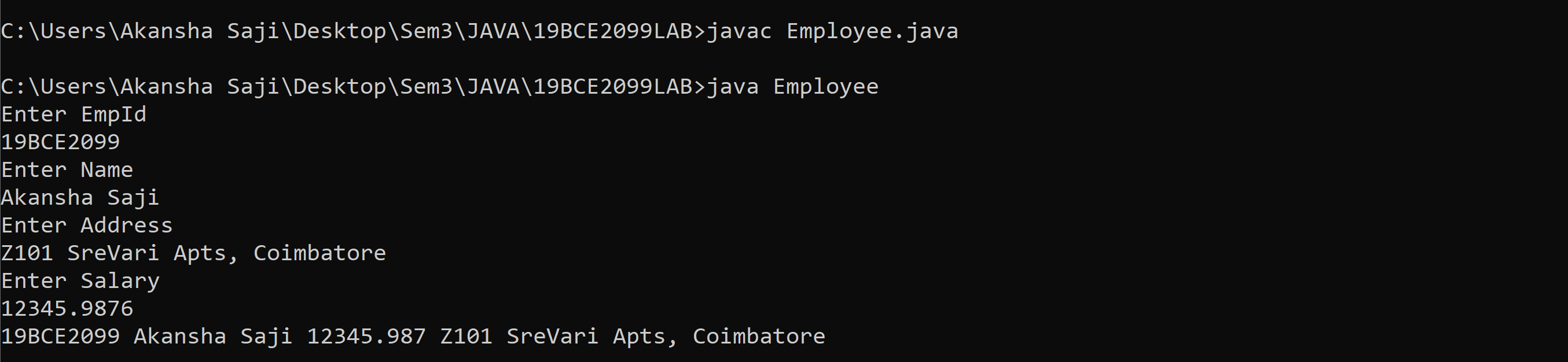
System.err.println("Error: Target File Cannot Be Read");

}

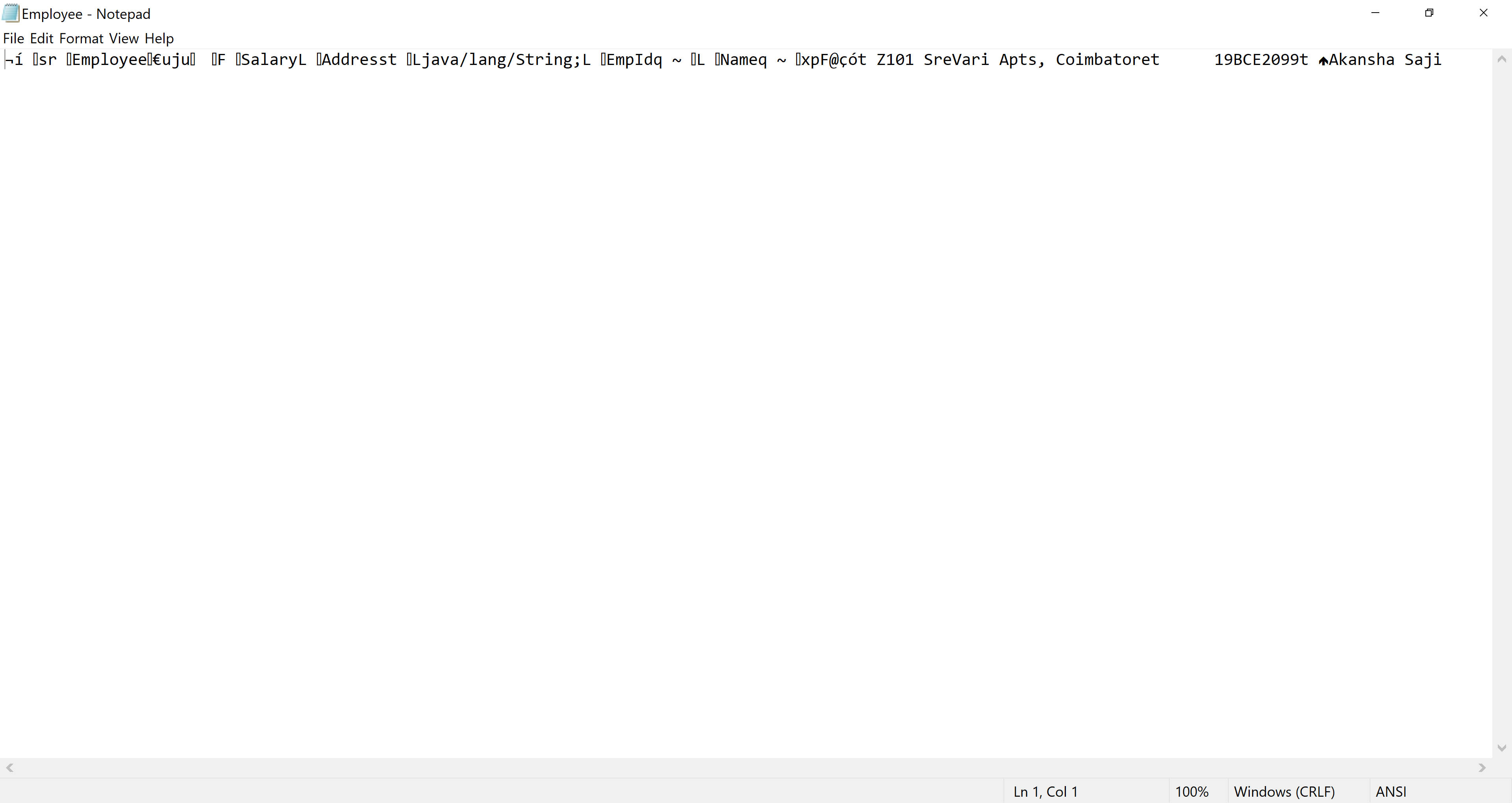
}

}

**Output**:



**Employee.txt:**



**Q4. Java Program to Convert the Content of File to LowerCase.**

**Program:**

import java.io.\*;

public class LowerCaseContent

{

public static void main(String args[])

{

try

{

BufferedReader br=new BufferedReader(new FileReader("D:\\LowerCase.txt"));

String s=null;

while((s=br.readLine()) != null)

{

System.out.println(s.toLowerCase());

}

}

catch(Exception e){

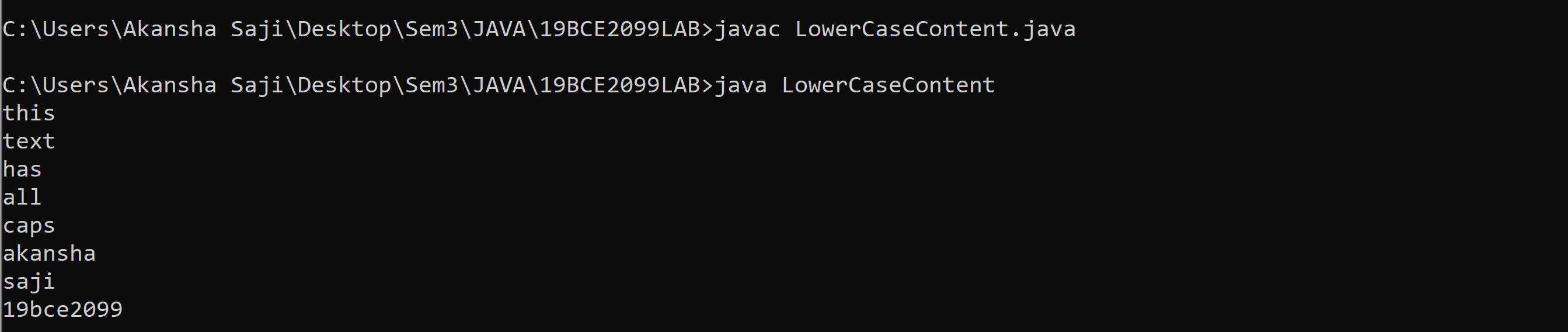
System.err.println("Error: Target File Cannot Be Read");

}

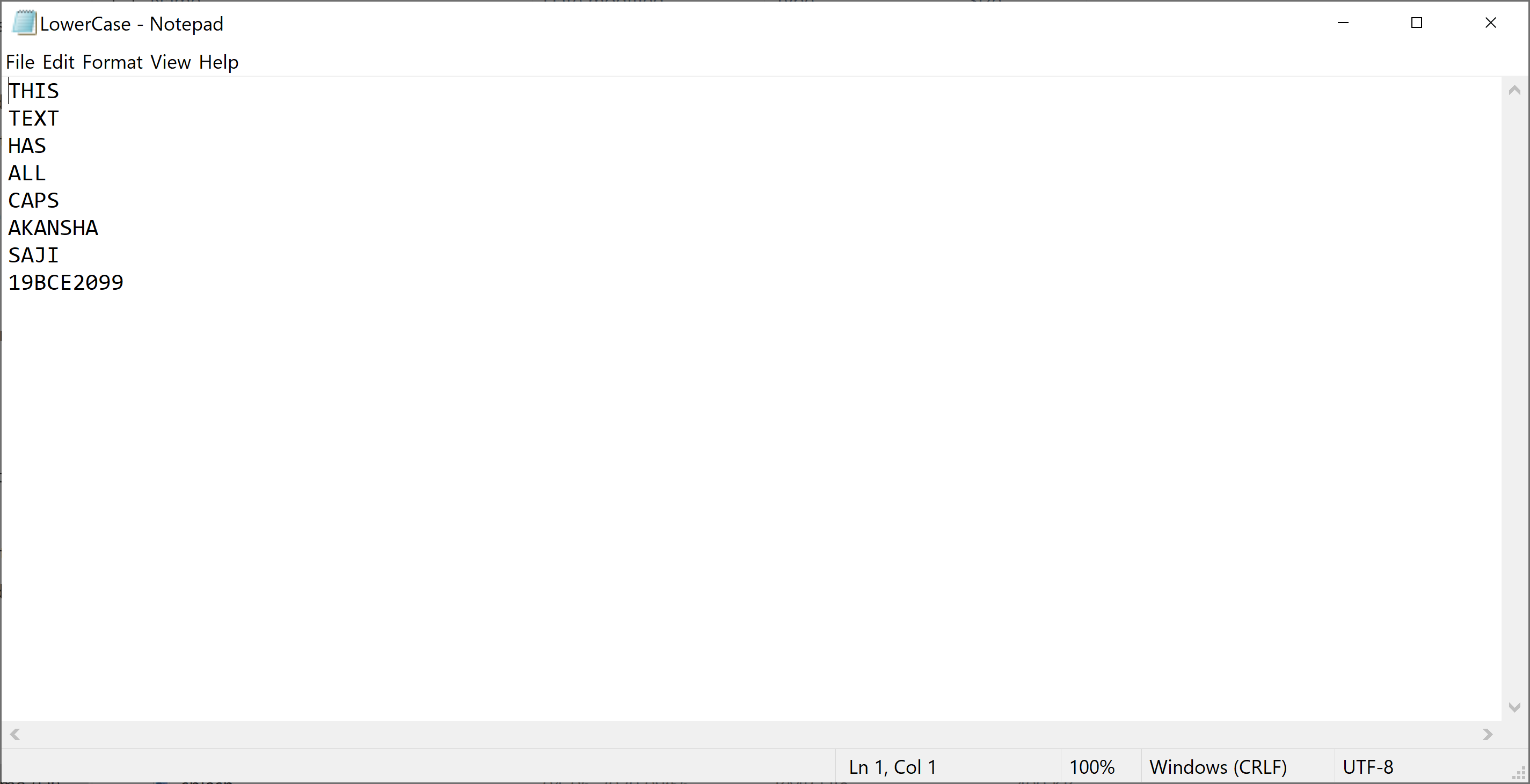
}

}

**Output:**



LowerCase.txt:



**Q5. Java Program to Create and Count Number of Characters in a File.**

**Program:**

import java.io.\*;

public class FileCharacterCount

{

public static void main(String args[])

{

try

{

BufferedReader br=new BufferedReader(new FileReader("D:\\CharacterCount.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){

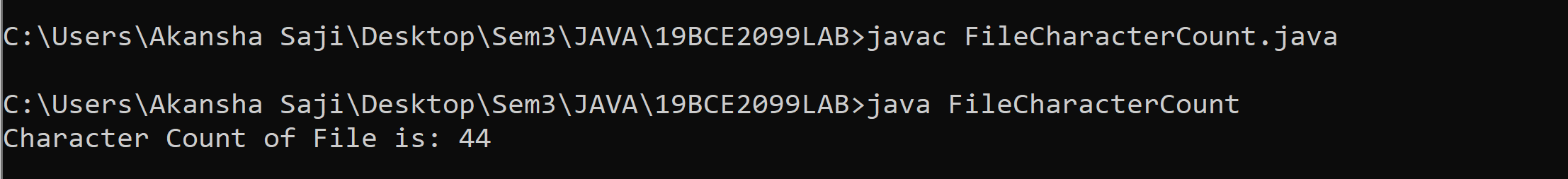
System.err.println("Error: Target File Cannot Be Read");

}

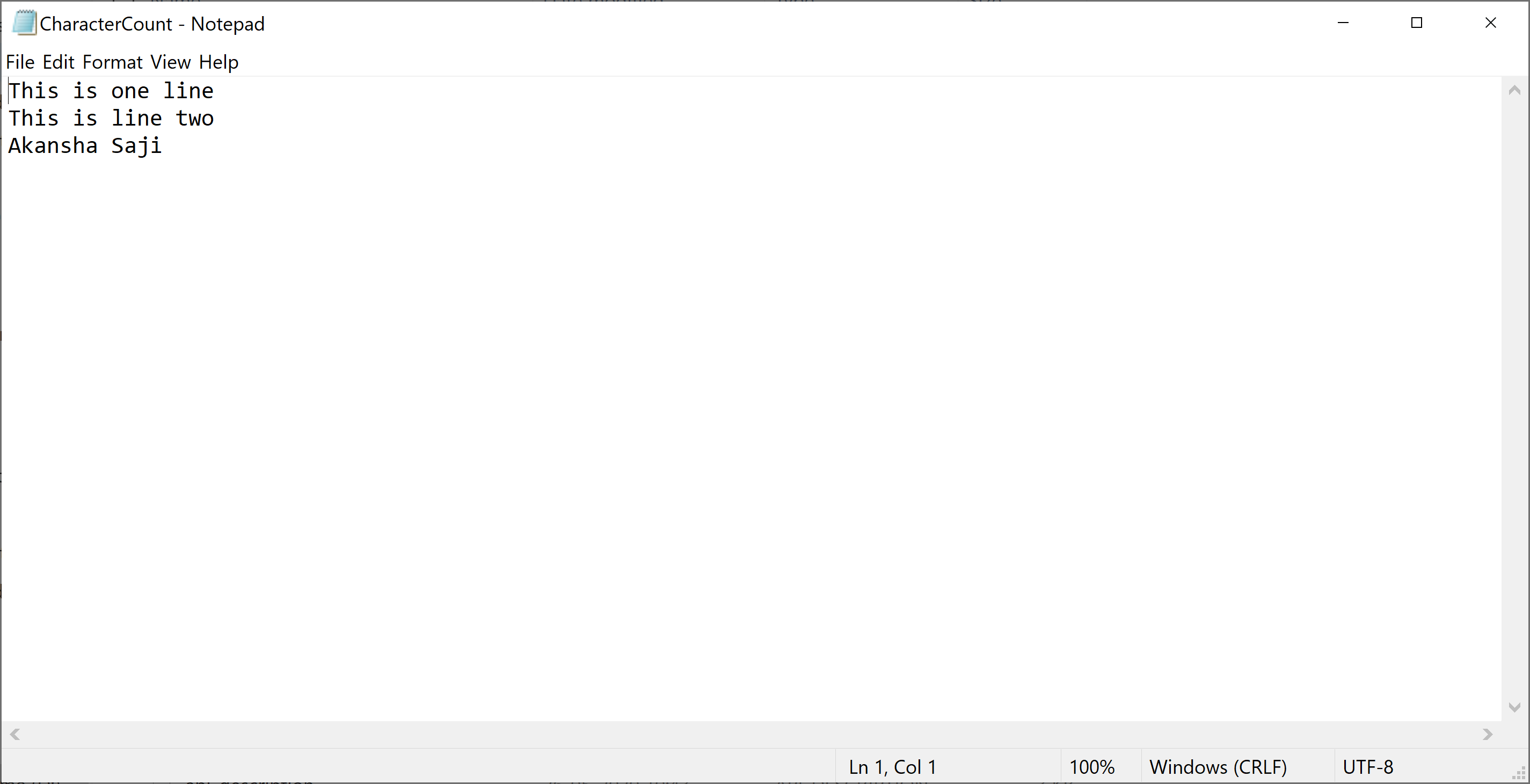
}

}

**Output:**



**CharacterCount.txt**



**Q6. Java Program to Join Lines of Two given Files and Store them in a New file**

**Program**:

import java.io.\*;

public class MergeLines

{

public static void main(String args[])

{

try

{

BufferedReader br1=new BufferedReader(new FileReader("D:\\File1.txt"));

BufferedReader br2=new BufferedReader(new FileReader("D:\\File2.txt"));

BufferedWriter bw=new BufferedWriter(new FileWriter("D:\\JoinedFile.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){

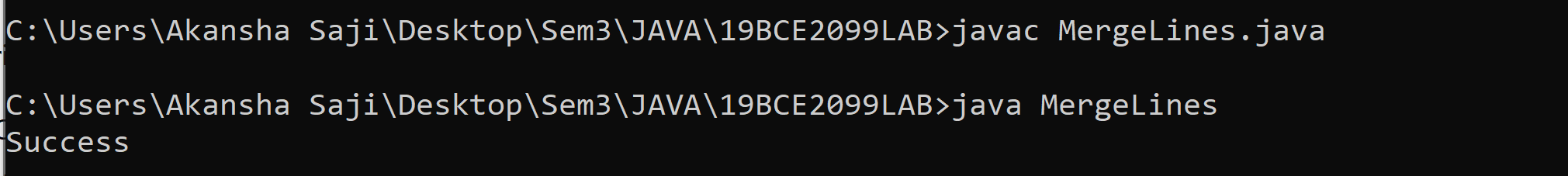
System.err.println("Error: Target File Cannot Be Read");

}

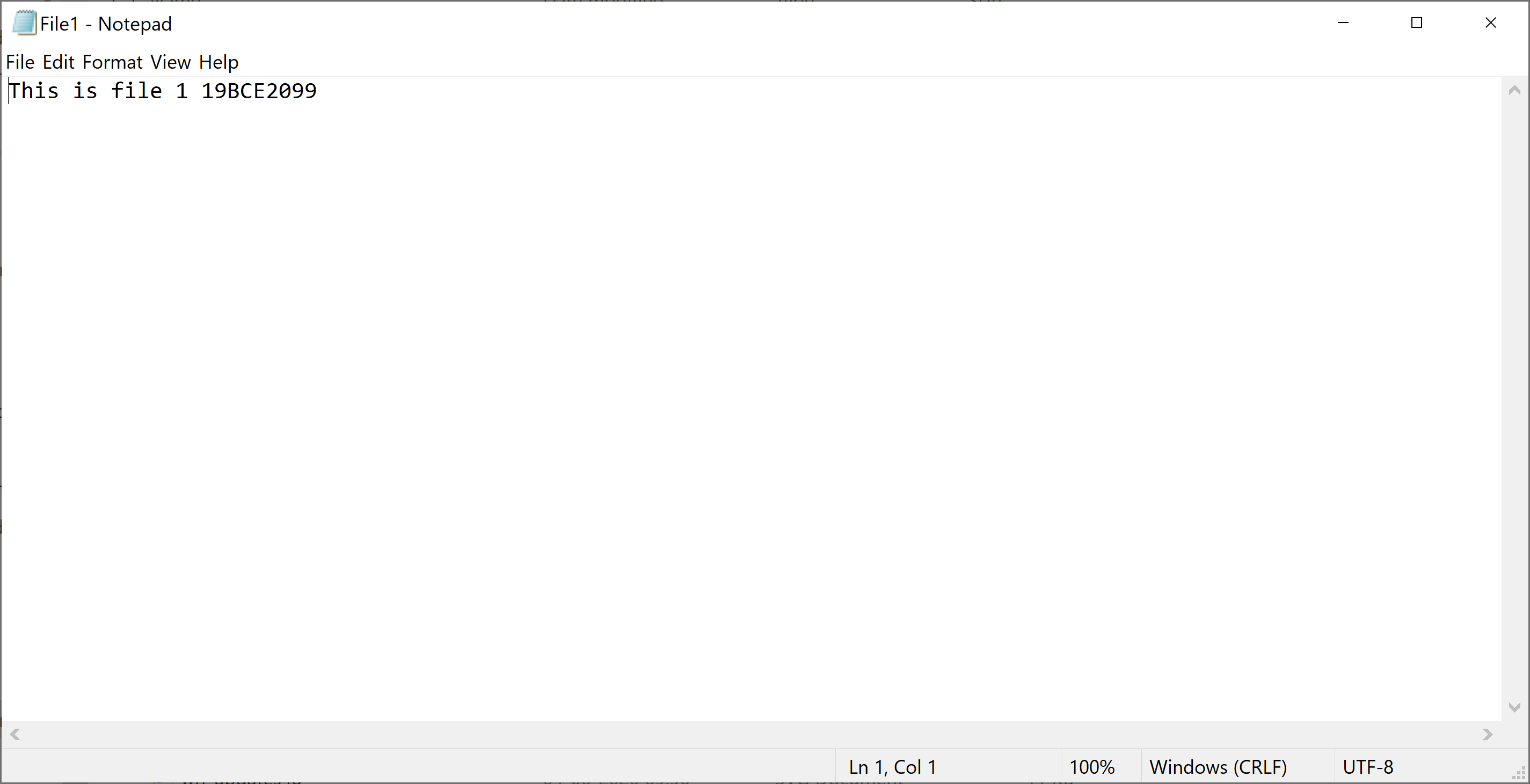
}

}

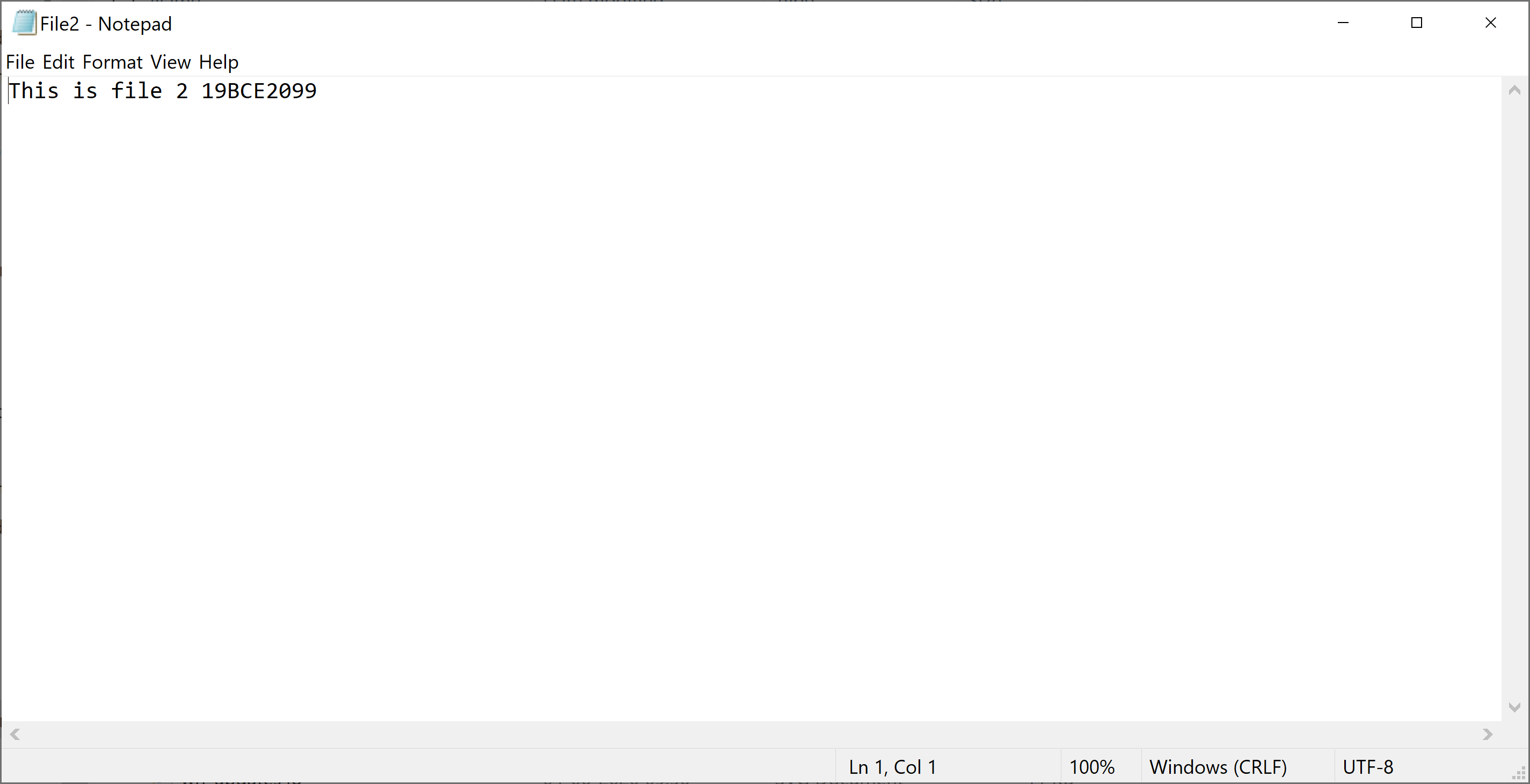
**Output**:



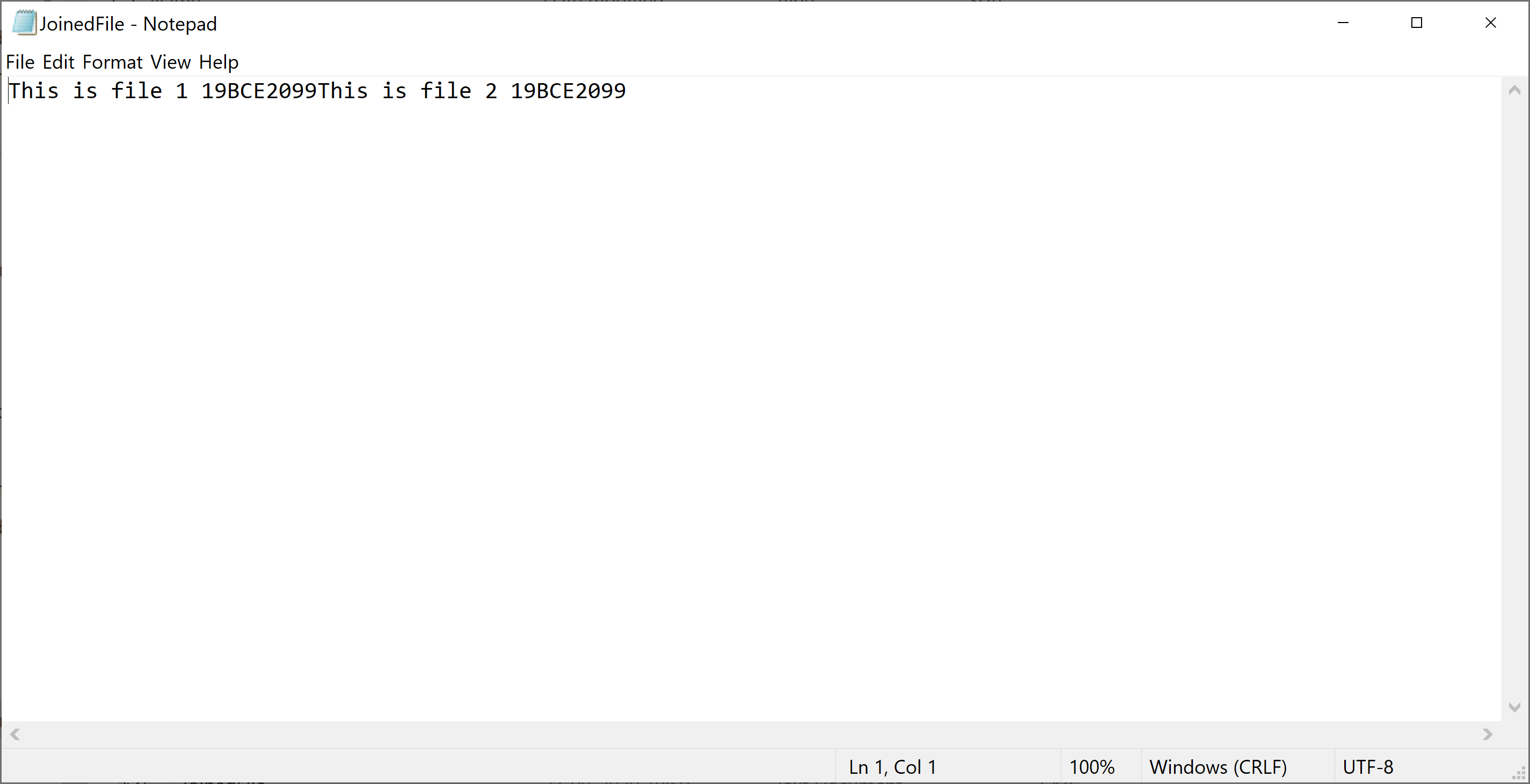
**File1.txt**



**File2.txt**



**JoinedFile.txt**



**Q7. Java Program to Collect Statistics of a Source File like Total Lines, Total no. of Blank Lines, Total no. of Lines Ending with Semicolon.**

**Program:**

import java.io.\*;

public class CollectStatisticsFile

{

public static void main(String args[])

{

try

{

BufferedReader br=new BufferedReader(new FileReader("D:\\StatisticsFile.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(" Total no. of BlankLines: "+countb);

System.out.println(" Total no. of Lines Ending with Semicolon: "+countsemi);

}

catch(Exception e){

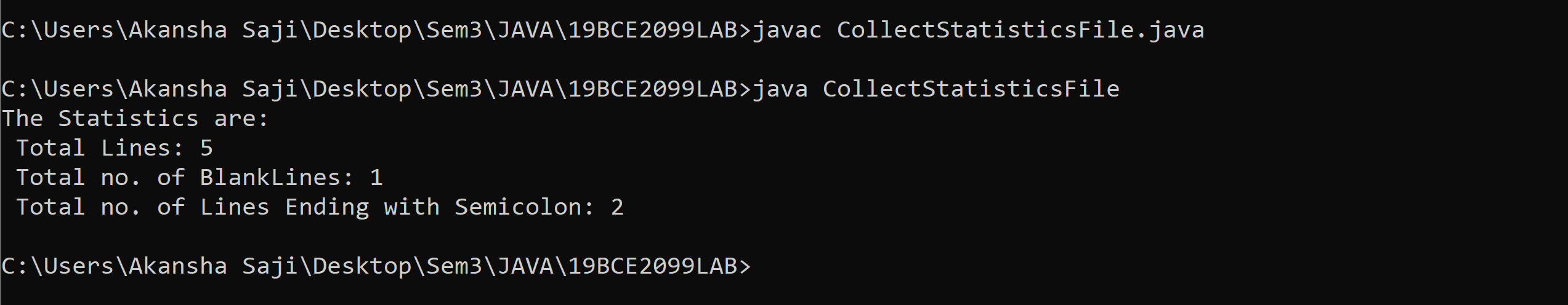
System.err.println("Error: Target File Cannot Be Read");

}

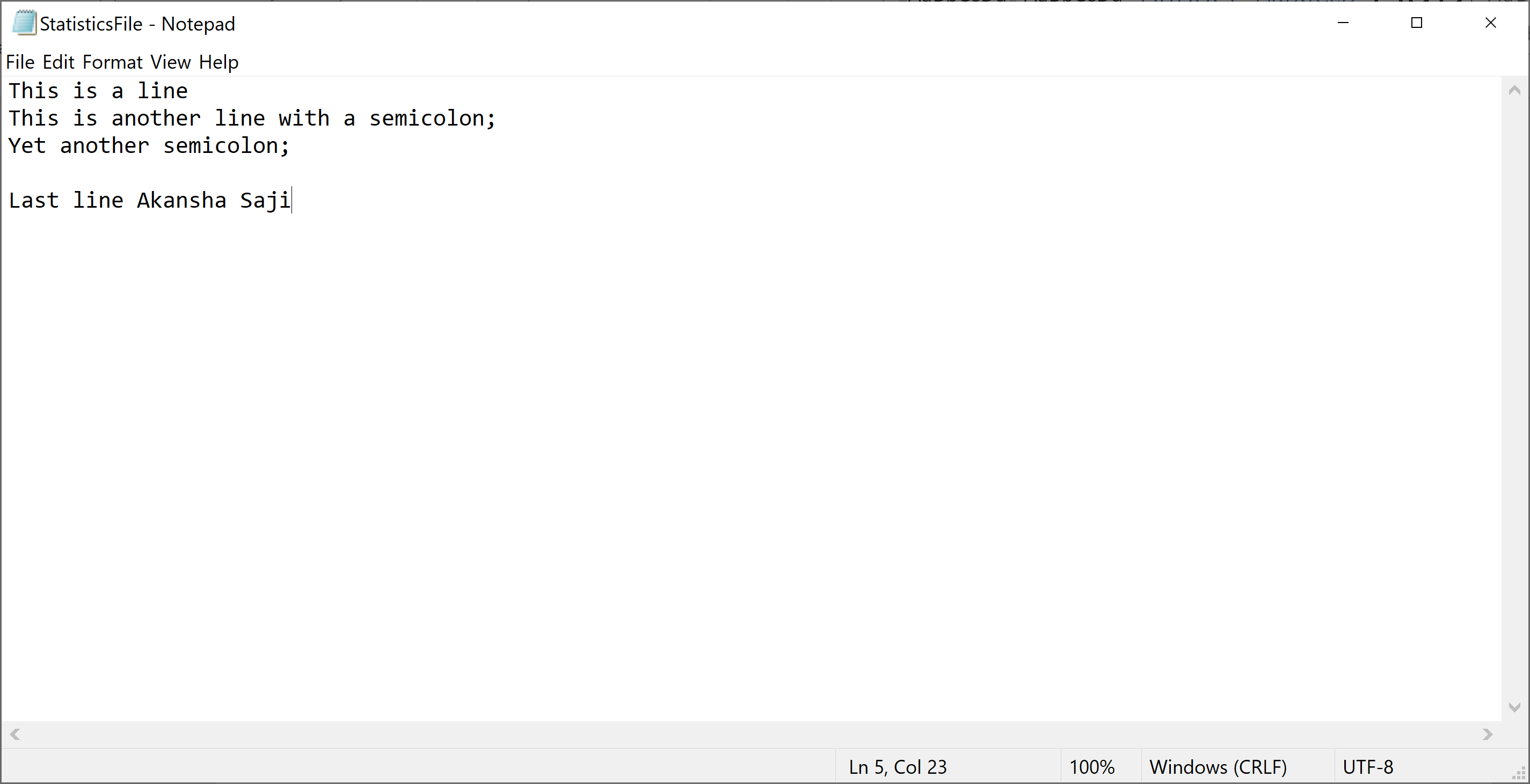
}

}

**Output:**



**StatisticsFile.txt:**



**Serialization & Collection**

**Q1. Write a Java program to store the objects of the class Employee (Emp\_id, Name, Designation and salary) in a file in sequential order. De-serialize the same file to display the details of the employees whose salary is lesser than 50000.**

**Program:**

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("EmployeeData.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("EmployeeData.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){

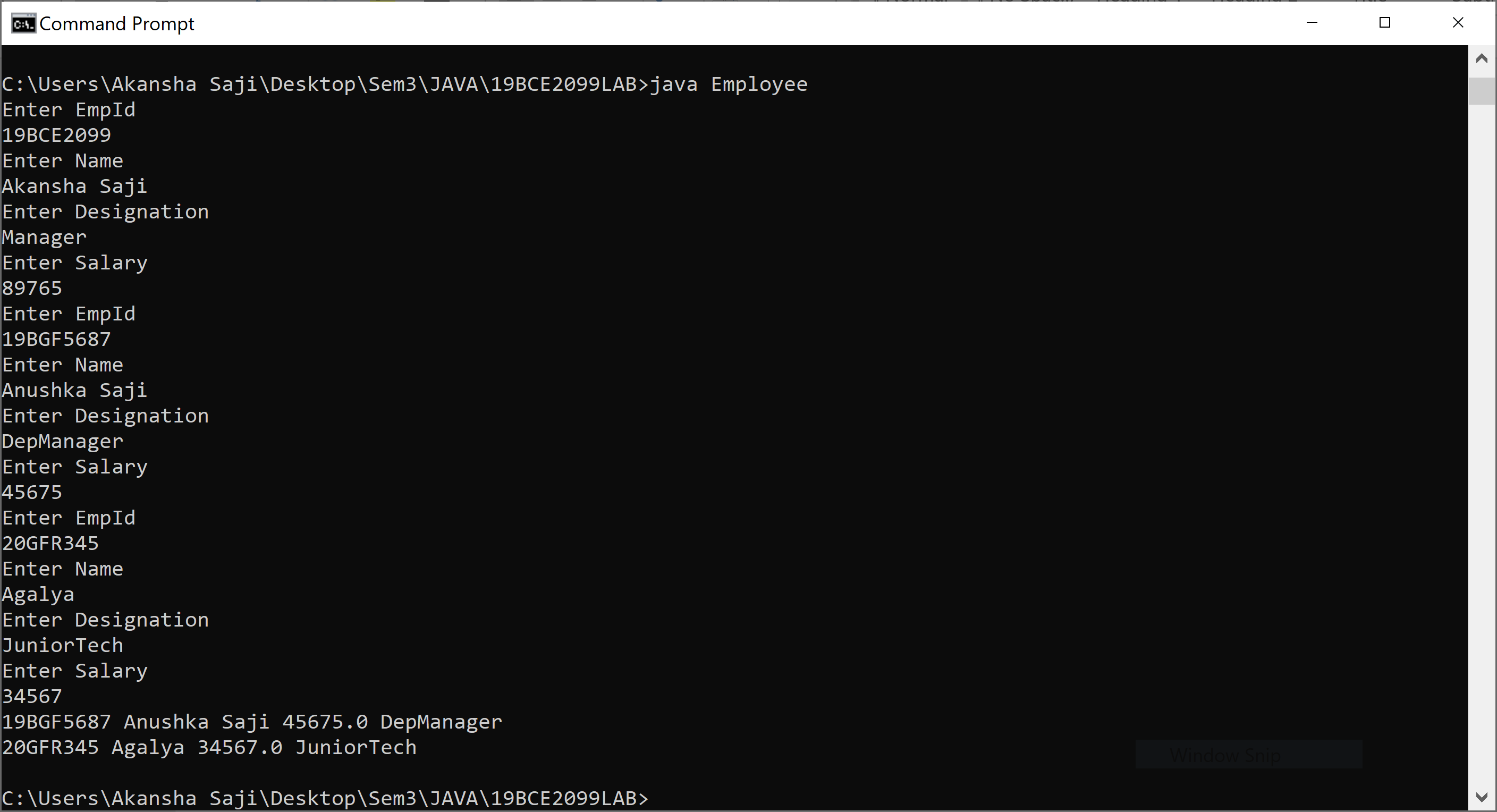
System.err.println("Error: Target File Cannot Be Read");

}

}

}

**Output:**



**Q2. Online bank loan sanction facility is launched to facilitate the customer. Write a java program to create class Loan with data members customer name, address, age, salary, loan amount, loan type(housing, vehicle, personal) . Take the necessary inputs and help the customer to write the object into the file when they submit the application. and Once the application is submitted bank manager will fetch the loan details from the file and verify the details for approval.**

**Program**:

import java.io.\*;

import java.util.Scanner;

class Loan implements Serializable

{

//customer name, address, age, salary, loan amount, loan type(housing, vehicle, personal)

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 CustomerName");

String CustomerName= input.nextLine();

System.out.println("Enter Address");

String Address= input.nextLine();

System.out.println("Enter LoanType");

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 LoanAmount");

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("Invalid Name");

}

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();

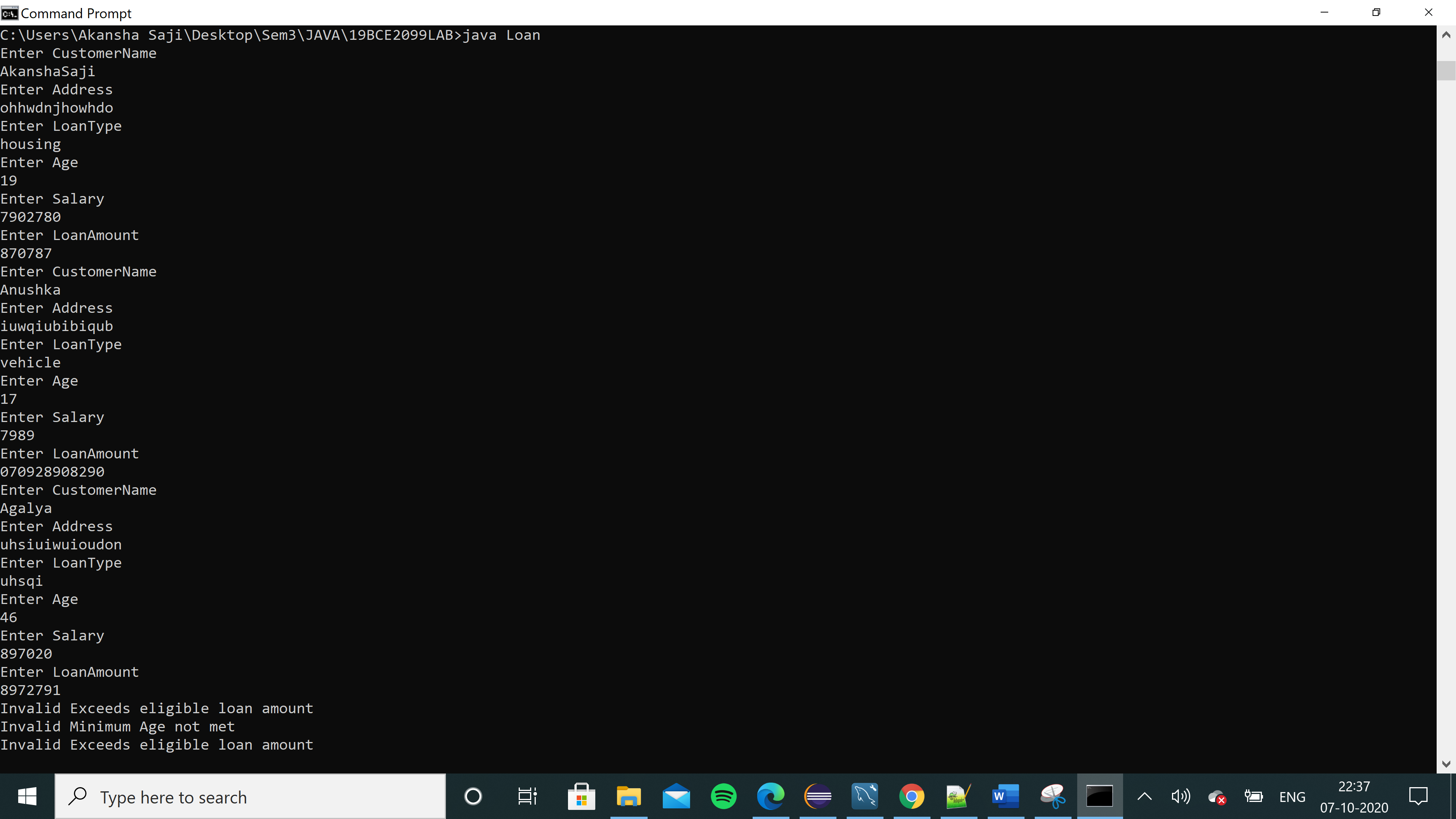
System.err.println("Error: Target File Cannot Be Read");

}

}

}

**Output**:



**Q3. Write a Java program to read student id, name and marks of three subjects and add student to list and print the student details using ArrayList and Iterator.**

**Program:**

import java.util.\*;

class StudentArrayList

{

String StudentID;

String Name;

int marks1;

int marks2;

int marks3;

StudentArrayList(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);

StudentArrayList s[]= new StudentArrayList[5];

ArrayList<StudentArrayList> alist= new ArrayList<StudentArrayList>();

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 StudentArrayList(StudentID,Name,marks1,marks2,marks3);

alist.add(s[i]);

}

for(StudentArrayList 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);

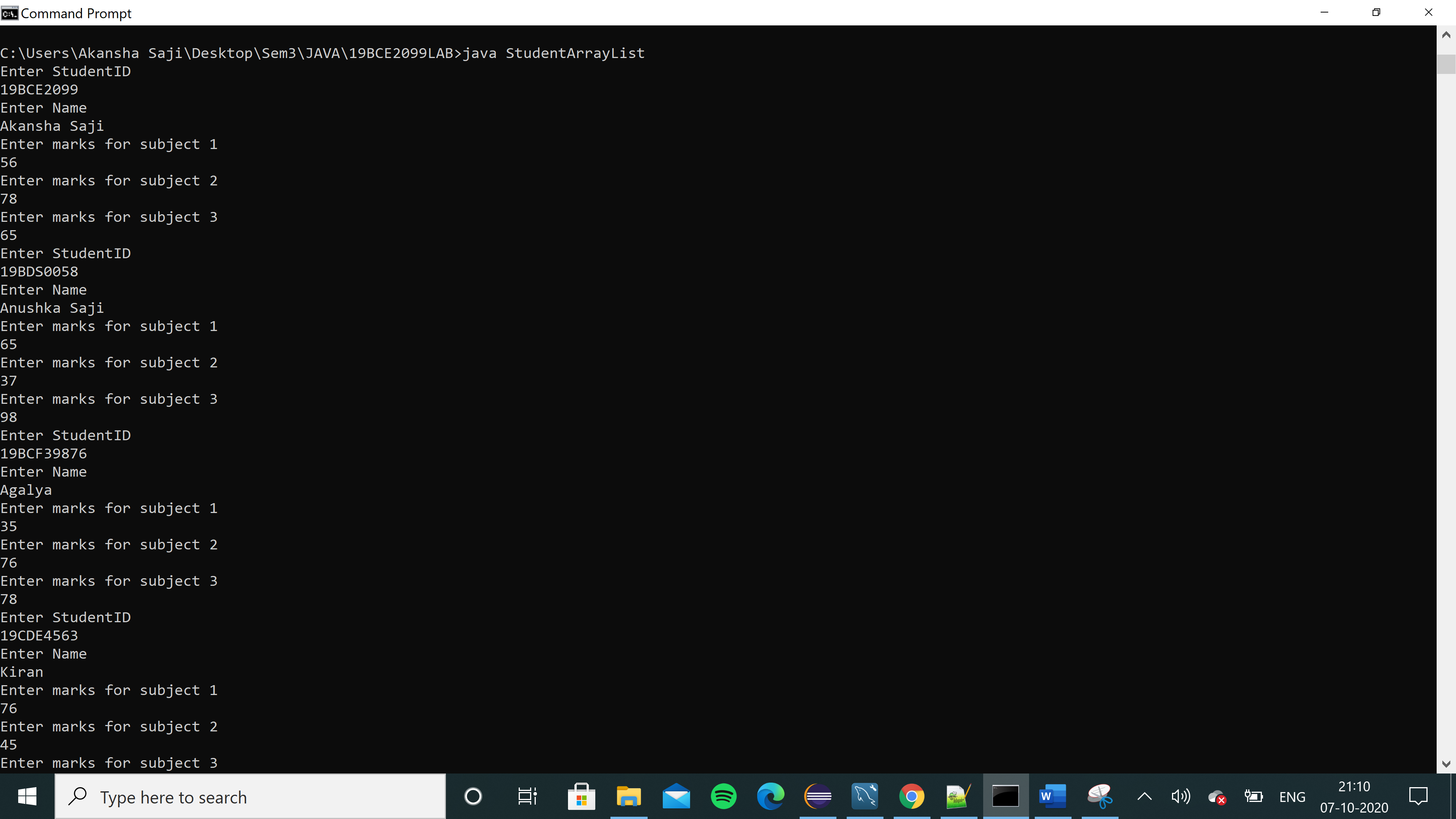
System.out.println("Student marks 3:"+l.marks3);

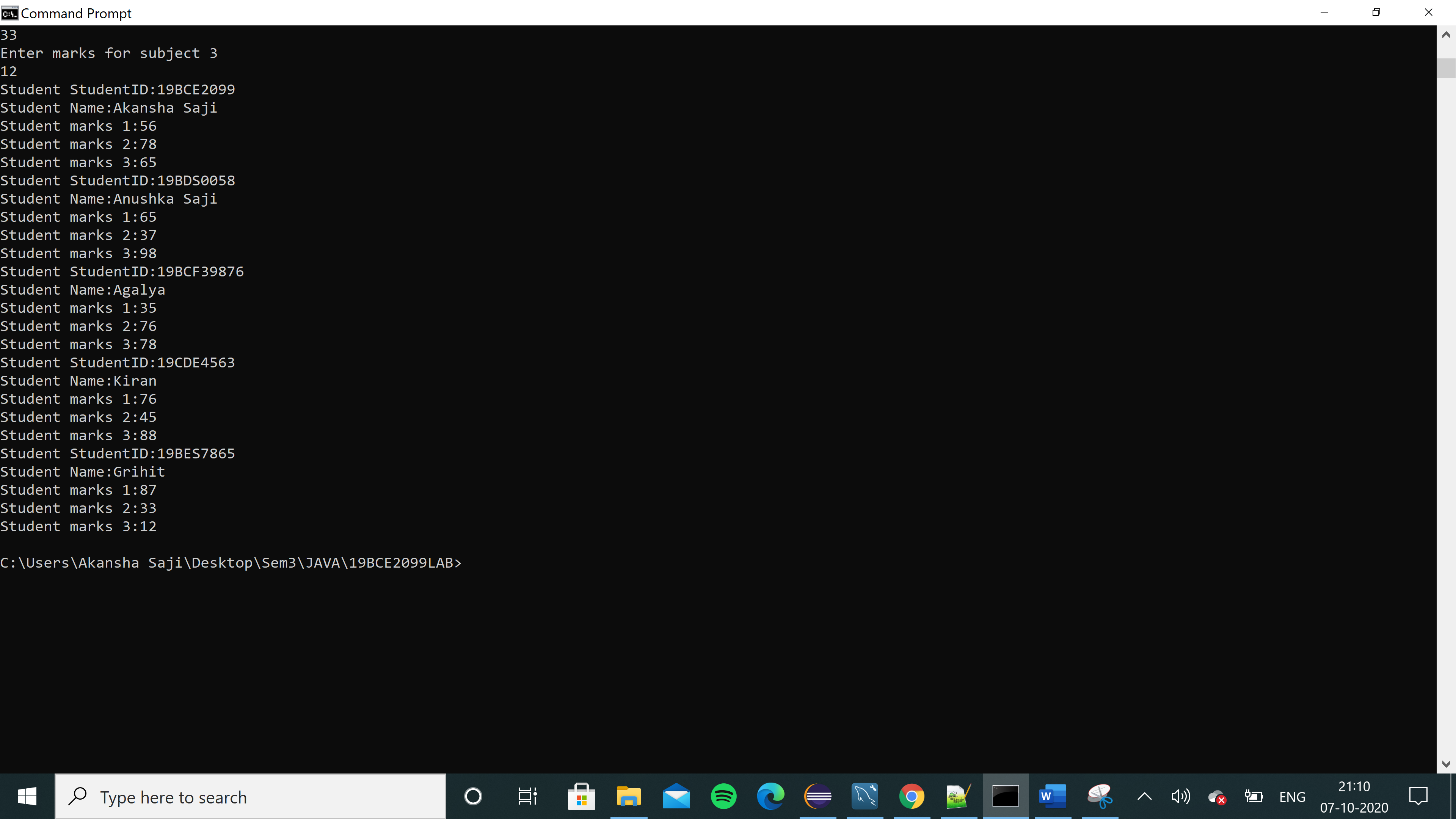
}

}

}

**Output:**





**JDBC**

**Q1. 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**

**Program**:

**package** facultymysql;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**import** java.util.Scanner;

**public** **class** Faculty {

**public** **static** **void** insertRecords(PreparedStatement p) **throws** SQLException

{

Scanner input=**new** Scanner(System.***in***);

System.***out***.println("Enter employee id");

String id=input.nextLine();

System.***out***.println("Enter faculty name");

String name=input.nextLine();

System.***out***.println("Enter DOB");

String dob=input.nextLine();

System.***out***.println("Enter date of joining");

String joindate=input.nextLine();

System.***out***.println("Enter designation");

String des=input.nextLine();

p.setString(1, id);

p.setString(2, name);

p.setString(3, dob);

p.setString(4, joindate);

p.setString(5, des);

p.executeUpdate();

}

**private** **static** **void** displayRecord(ResultSet rs) **throws** SQLException {

String id=rs.getString(1);

System.***out***.println("Employee id: "+id);

String name=rs.getString(2);

System.***out***.println("Faculty name: "+name);

String dob=rs.getString(3);

System.***out***.println("DOB: "+dob);

String joindate=rs.getString(4);

System.***out***.println("Date of joining: "+joindate);

String des=rs.getString(5);

System.***out***.println("Designation: "+des);

}

**public** **static** **void** main(String[] args) **throws** ClassNotFoundException, SQLException {

// **TODO** Auto-generated method stub

Class.*forName*("com.mysql.cj.jdbc.Driver");

Connection con = DriverManager.*getConnection*("jdbc:mysql://localhost/facultydb","root","root");

Statement s1=con.createStatement();

Statement s2=con.createStatement();

Statement s3=con.createStatement();

String sql1="create table Faculty(empid varchar(20), facultyname varchar(20), DOB date ,Dateofjoining date,designation varchar(20))";

s1.executeUpdate(sql1);

System.***out***.println("Faculty table created");

String sql2="insert into Faculty "+"(empid,facultyname,DOB,Dateofjoining,designation) values(?,?,?,?,?)";

PreparedStatement pstmt1 = con.prepareStatement(sql2);

ResultSet rs1= s2.executeQuery("select \* from Faculty");

ResultSet rs2=s3.executeQuery("select \* from Faculty where designation="+" 'Senior professor'");

**for**(**int** i=0;i<5;i++)

{

*insertRecords*(pstmt1);

}

System.***out***.println("19BCE2099 Retrieve all the faculty details. ");

**while**(rs1.next())

{

*displayRecord*(rs1);

}

System.***out***.println("19BCE2099 Retrieve the faculty information whose designation is Senior professor. ");

**while**(rs2.next())

{

*displayRecord*(rs2);

}

s1.close();

s2.close();

s3.close();

rs1.close();

rs2.close();

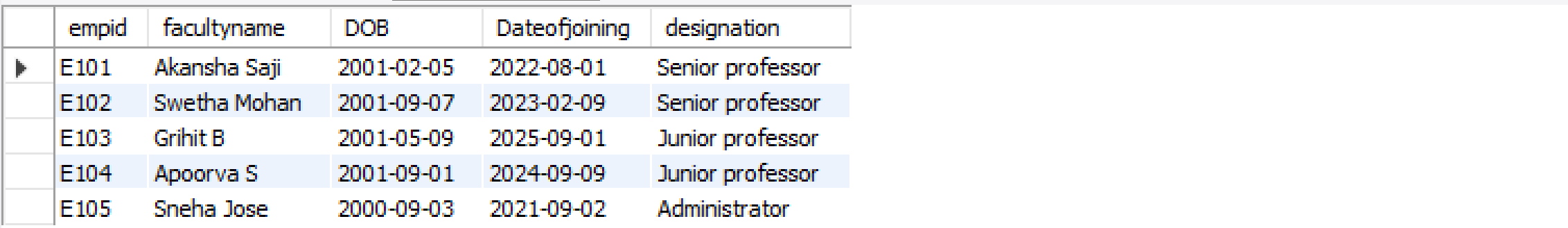
con.close();

}

}

**Ouput**:

**Mysql Table:**



**Eclipse Console:**

19BCE2099 Retrieve all the faculty details.

Employee id: E101

Faculty name: Akansha Saji

DOB: 2001-02-05

Date of joining: 2022-08-01

Designation: Senior professor

Employee id: E102

Faculty name: Swetha Mohan

DOB: 2001-09-07

Date of joining: 2023-02-09

Designation: Senior professor

Employee id: E103

Faculty name: Grihit B

DOB: 2001-05-09

Date of joining: 2025-09-01

Designation: Junior professor

Employee id: E104

Faculty name: Apoorva S

DOB: 2001-09-01

Date of joining: 2024-09-09

Designation: Junior professor

Employee id: E105

Faculty name: Sneha Jose

DOB: 2000-09-03

Date of joining: 2021-09-02

Designation: Administrator

19BCE2099 Retrieve the faculty information whose designation is Senior professor.

Employee id: E101

Faculty name: Akansha Saji

DOB: 2001-02-05

Date of joining: 2022-08-01

Designation: Senior professor

Employee id: E102

Faculty name: Swetha Mohan

DOB: 2001-09-07

Date of joining: 2023-02-09

Designation: Senior professor

**Eclipse IDE:**

