1. Write a program to write to a file, then read from the file and display the contents on the console

import java.util.\*;

import java.io.\*;

public class EmployeeFileIO {

public static void main(String[] args) {

try {

Scanner obj = new Scanner(System.in);

FileOutputStream fout = new FileOutputStream("employee2.txt");

int empno;

String empname;

int salary;

System.out.println("Enter the number of employees:");

int limit = obj.nextInt();

obj.nextLine(); // Consume the newline left by nextInt()

for (int j = 0; j < limit; j++) {

try {

System.out.println("Enter the employee number:");

empno = obj.nextInt();

obj.nextLine(); // Consume the newline left by nextInt()

String a = String.valueOf(empno);

byte a1[] = a.getBytes();

fout.write(a1);

fout.flush();

fout.write("\r\n".getBytes());

System.out.println("Enter the employee name:");

empname = obj.nextLine();

byte b[] = empname.getBytes();

fout.write(b);

fout.flush();

fout.write("\r\n".getBytes());

System.out.println("Enter the employee salary:");

salary = obj.nextInt();

obj.nextLine(); // Consume the newline left by nextInt()

String c = String.valueOf(salary);

byte c1[] = c.getBytes();

fout.write(c1);

fout.flush();

fout.write("\r\n".getBytes());

} catch (InputMismatchException e) {

System.out.println("Invalid input. Please enter a valid integer.");

obj.nextLine(); // Consume the invalid input

j--; // Decrement the loop counter to retry entering the employee's data

}

}

fout.close();

} catch (Exception e) {

System.out.println(e);

}

try {

FileInputStream fin = new FileInputStream("employee2.txt");

System.out.println("Contents of the file:");

int data;

while ((data = fin.read()) != -1) {

System.out.print((char) data);

}

fin.close();

} catch (Exception e) {

System.out.println(e);

}

}

}

2. Write a program to copy one file to another.

import java.io.\*;

import java.util.\*;

class CopyFile {

    public static void main(String arg[]) throws Exception {

        Scanner sc = new Scanner(System.in);

        System.out.print("Provide source file name: ");

        String sfile = sc.next();

        System.out.print("Provide destination file name: ");

        String dfile = sc.next();

        FileReader fin = new FileReader(sfile);

        FileWriter fout = new FileWriter(dfile, true); // true = append mode

        int c;

        while ((c = fin.read()) != -1) {

            fout.write(c);

        }

        System.out.println("Copy finished...");

        fin.close();

        fout.close();

    }

}

3. Write a program that reads from a file having integers. Copy even numbers and odd numbers to separate files.

import java.io.\*;

import java.util.Scanner;

public class EvenOddNumbers {

public static void main(String[] args) {

try {

Scanner inputScanner = new Scanner(System.in);

System.out.print("Enter the name of the input file: ");

String inputFileName = inputScanner.nextLine();

FileInputStream inputFile = new FileInputStream(inputFileName);

Scanner fileScanner = new Scanner(inputFile);

FileOutputStream evenFile = new FileOutputStream("even.txt");

FileOutputStream oddFile = new FileOutputStream("odd.txt");

PrintWriter evenWriter = new PrintWriter(evenFile);

PrintWriter oddWriter = new PrintWriter(oddFile);

while (fileScanner.hasNextInt()) {

int number = fileScanner.nextInt();

if (number % 2 == 0) {

evenWriter.println(number); // Write even numbers to the even file

} else {

oddWriter.println(number);  // Write odd numbers to the odd file

}

}

evenWriter.close();

oddWriter.close();

inputFile.close();

System.out.println("Even and odd numbers have been copied to separate files.");

} catch (IOException e) {

System.err.println("Error: " + e.getMessage());

}

}

}