Assignment 4

1.

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class FileStats {

public static void main(String[] args) {

String fileName = "File1.txt";

int wordCount = 0;

int charCount = 0;

int lineCount = 0;

try (BufferedReader reader = new BufferedReader(new FileReader(fileName))) {

String line;

while ((line = reader.readLine()) != null) {

lineCount++;

String[] words = line.trim().split("\\s+");

wordCount += words.length;

for (String word : words) {

charCount += word.length();

}

}

} catch (IOException e) {

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

return;

}

System.out.println("Word count: " + wordCount);

System.out.println("Character count: " + charCount);

System.out.println("Line count: " + lineCount);

}

}

2.

class Customer {

private int accountNo;

private String accName;

private double balance;

public Customer(int accountNo, String accName, double balance) {

this.accountNo = accountNo;

this.accName = accName;

this.balance = balance;

}

public synchronized void deposit(double amount) {

System.out.println("Depositing " + amount + " into account " + accountNo);

balance += amount;

notify();

}

public synchronized void withdraw(double amount) {

while (balance < amount) {

System.out.println("Insufficient balance in account " + accountNo + ", waiting for deposit...");

try {

wait();

} catch (InterruptedException e) {

e.printStackTrace();

}

}

System.out.println("Withdrawing " + amount + " from account " + accountNo);

balance -= amount;

}

}

class Main {

public static void main(String[] args) {

Customer customer = new Customer(123456, "John Doe", 1000);

Thread t1 = new Thread(() -> {

customer.withdraw(500);

});

Thread t2 = new Thread(() -> {

customer.deposit(1000);

});

t1.start();

t2.start();

}

}

4.

Public static void main(String[] args) {

String s1 = “abcde”;

String goal1 = “cdeab”;

Boolean result1 = canShiftToGoal(s1, goal1);

System.out.println(result1); // Output: true

String s2 = “abcde”;

String goal2 = “abced”;

Boolean result2 = canShiftToGoal(s2, goal2);

System.out.println(result2); // Output: false

}

5.

Class PrimeExample implements Runnable {

Public void run() {

Int m = 20;

For (int I = 2; I <= m; i++) {

Boolean flag = true;

For (int j = 2; j < I; j++) {

If (I % j == 0) {

Flag = false;

Break;

}

}

If (flag) {

System.out.println(I + “ is prime number”);

} else {

System.out.println(I + “ is not prime number”);

}

}

}

}

Class prime {

Public static void main(String args[]) {

Try {

PrimeExample p1 = new PrimeExample();

Thread t1 = new Thread(p1);

T1.start();

} catch (Exception e) {

System.out.println(e.getMessage());

}

}

}