**Session\_5\_Assignments:**

1. **Write a simple Timer that can periodically print a timeout message.**

**//**timer to perform a task after a delay:

import java.util.Timer;

import java.util.TimerTask;

/\*\*

\* Simple demo that uses java.util.Timer to schedule a task

\* to execute once 5 seconds have passed.

\*/

public class Reminder {

Timer timer;

public Reminder(int seconds) {

timer = new Timer();

timer.schedule(new RemindTask(), seconds\*1000);

}

class RemindTask extends TimerTask {

public void run() {

System.out.println("Time's up!");

timer.cancel(); //Terminate the timer thread

}

}

public static void main(String args[]) {

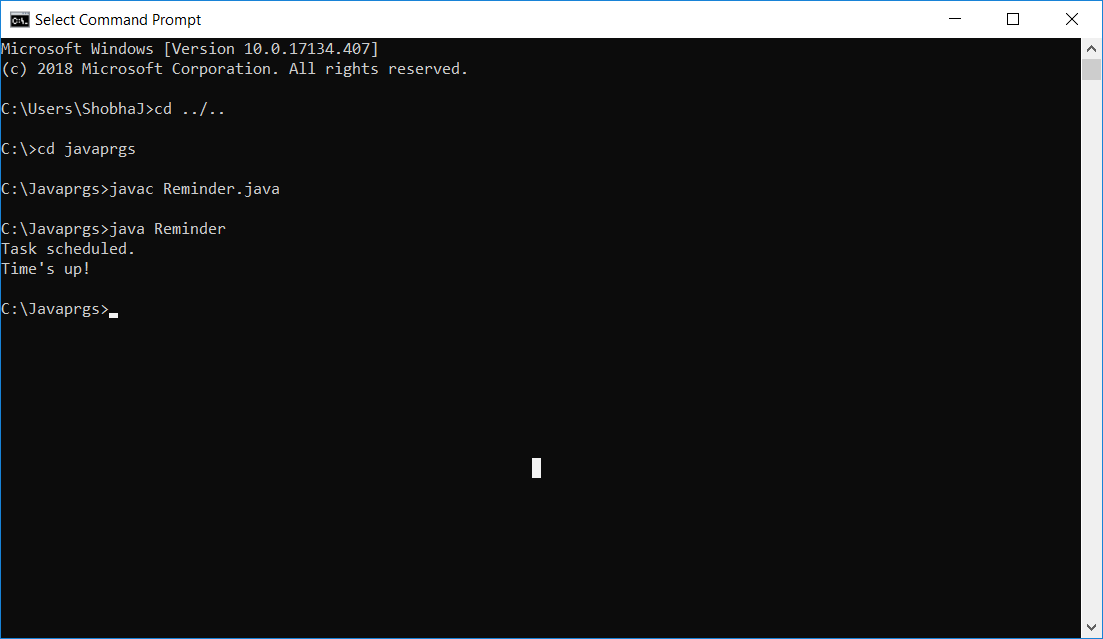
new Reminder(5);

System.out.println("Task scheduled.");

}

}

****



**2. Write a program to build any collection containing duplicates. Create its copy with all duplicates Removed**

import java.util.ArrayList;

import java.util.LinkedHashset;

import java.util.List;

import java.util.Set;

Public class ArrayListDuplicates {

Public static void main(String args[]){

//creating Arraylist with Duplicate elements

List<Integer> primes = new ArrayList<Integer>();

Primes.add(2);

Primes.add(3);

Primes.add(5);

Primes.add(7); //duplicate

Primes.add(7);

Primes.add(11);

System.out.prntln("list of prime number : " + primes);

//now to remove duplicates

//LinkedHashset it sets

Set<Integer> primesWithoutDuplicates = new LinkedHashset<Integer>(primes);

//to clear Arraylist ,so that we can copy all elements from LinkedHashset

primes.clear();

primes.addAll(primesWithoutDuplicates);

System.out.prntln("list of primes without duplicates: " + primes);

}



}