**public** **int** mostNumberVisitsByIP(HashMap<String, Integer> max){

//HashMap<String, Integer>mostFreq = new HashMap<String, Integer>();

max = **new** HashMap<String, Integer>();

**int** count = 0;

**int** maxIP = 0;

**for**(LogEntry le : *records*){

String ip = le.getIpAddress();

**if** (!max.containsKey(ip)){

max.put(ip, 1);

}**else**{

max.put(ip, max.get(ip)+1);

}}

**for** (String s : max.keySet()){

count = max.get(s);

maxIP+=1;

**if** (count > maxIP){

count = maxIP;

}

}

**return** count;

}

**public** ArrayList<String> iPsMostVisits(HashMap<String, Integer> numberIPappear){//is it possible to obtain without HashMap to hold the key value in this case ?

ArrayList<String>ipList = **new** ArrayList<String>();

HashMap<String, Integer>IPappear = **new** HashMap<String, Integer>();

// int maxIp = 0;

**int** maxIp = mostNumberVisitsByIP(numberIPappear);

**for**(LogEntry le : *records*){

String ip = le.getIpAddress();

**if** (!IPappear.containsKey(ip)){

IPappear.put(ip, 1);

}**else**{

IPappear.put(ip, IPappear.get(ip)+1);

}

**int** count = IPappear.size();

**if** (maxIp == count){

ipList.add(ip);

// ipList.get(maxIp);

}

}

**return** ipList;

}

**public** HashMap<String, ArrayList<String>>iPsForDays(){//Date is String and ArrayList is IP address

HashMap<String, ArrayList<String>>ipDay = **new** HashMap<String, ArrayList<String>>();

**for**(LogEntry le : *records*){

Date ipdate = le.getAccessTime();

SimpleDateFormat formatter = **new** SimpleDateFormat("MMM dd");

String day = formatter.format(ipdate);

String daytoString = day.toString();

ArrayList<String>ipAddMost = **new** ArrayList<String>();

String ip = le.getIpAddress();

**if** (!ipAddMost.contains(ip)){

ipAddMost.add(ip);

}

//ipAdd = ipDay.get(records);

ipDay.put(daytoString, ipAddMost);

}

**return** ipDay;

}

**public** String dayWithMostIPVisits(HashMap<String, ArrayList<String>>dayMostVisit){

dayMostVisit = **new** HashMap<String, ArrayList<String>>();

String maxday = "";

**int** maxdaycount = 0;

//ArrayList<String> ip = iPsMostVisits(dayMostVisit)

// int maxVisit = mostNumberVisitsByIP(numberIPappear);

**for**(LogEntry le : *records*){

Date ipdate = le.getAccessTime();

SimpleDateFormat formatter = **new** SimpleDateFormat("MMM dd");

String dayt = formatter.format(ipdate);

String daytoString = dayt.toString();

ArrayList<String>dateDayMostVisit = **new** ArrayList<String>();

**if**(!dateDayMostVisit.contains(daytoString)){

dateDayMostVisit.add(daytoString);

maxdaycount+=1;

//String ip = le.getIpAddress();

**for**(String s: dateDayMostVisit){

// for (int i = 0; i < dateDayMostVisit.size(); i++){

// String date1 = dateDayMostVisit.get(i);

maxday = dateDayMostVisit.get(maxdaycount);

}}}

**return** maxday;

}

// public void printAll() {

// ArrayList<LogEntry>records1 = new ArrayList<LogEntry>();

// for (LogEntry le : records1) {

// System.out.println(le);

// }

// }

**public** **static** **void** main (String [] args){

LogAnalyzer la = **new** LogAnalyzer();

la.readFile("C:\\Users\\Karen Goh Seow Hui\\ELunar\\Assignment9\\src\\LogEntry\\weblog3-short\_log");

// int uniqueIps = la.countUniqueiPS();

// System.out.print("There are " + uniqueIps + " unique IPs");

// System.out.print("\n");

// // la.printAllHigherThanNum(200);

//

// System.out.print("\n");

//

// // la.printAll();

//

//

//// la.readFile("C:\\Users\\Karen Goh Seow Hui\\ELunar\\Assignment5\\src\\LogEntry\\weblog3-short\_log");

HashMap<String, Integer>max = **new** HashMap<String, Integer>();

//// // for (LogEntry le : records){

// HashMap<String, Integer>maxcount = countVisitsPerIP();

// System.out.println(maxcount);

// System.out.println(la.countUniqueIps());//print out the no of different Ip in the log

**int** high = la.mostNumberVisitsByIP(max);

System.***out***.println(high);

ArrayList<String>ipMost = la.iPsMostVisits(max);

System.***out***.println(la.iPsMostVisits(max));

System.***out***.println(la.iPsForDays());

//}

}}