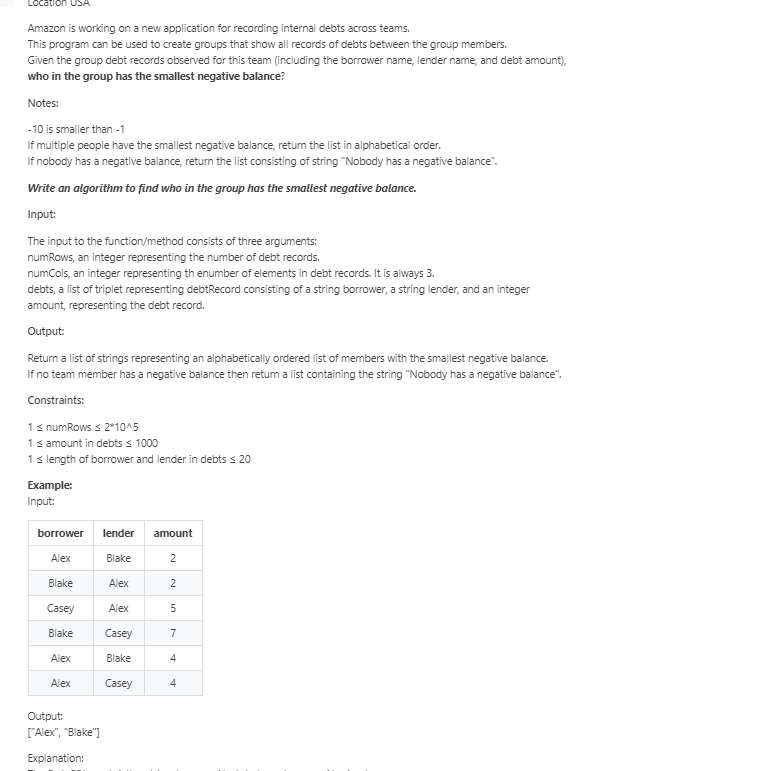
## Amazon | OA 2020 | Smallest Negative Balance || **Amazon OA - Amazon Debt Records**

<https://leetcode.com/discuss/interview-question/874127/amazon-oa2>



// "static void main" must be defined in a public class.

public class Main {

static class debtRecord {

String borrower = "";

String lender = "";

int amount = 0;

debtRecord(){}

debtRecord(String borrower, String lender, int amount) {

this.borrower = borrower;

this.lender = lender;

this.amount = amount;

}

}

static List<String> negativeAccounts(int numRows, int numCols, List<debtRecord> records) {

if(numRows == 0 || records == null) return new ArrayList<>();

HashMap<String, Integer> hm = new HashMap<>();

for(debtRecord r: records) {

String b = r.borrower, l = r.lender;

int amount = r.amount;

hm.put(b, hm.getOrDefault(b, 0) - amount);

hm.put(l, hm.getOrDefault(l, 0) + amount);

}

PriorityQueue<Map.Entry<String, Integer>> pq = new PriorityQueue<>(

(a,b) -> a.getValue() == b. getValue() ?

a.getKey().compareTo(b.getKey()) :

a.getValue() - b.getValue());

for(Map.Entry<String, Integer> entry: hm.entrySet()) {

if(entry.getValue() < 0) {

pq.add(entry);

}

}

List<String> res = new ArrayList<>();

while(!pq.isEmpty()) {

res.add(pq.poll().getKey());

}

if(res.size() == 0) {

res.add("Nobody");

}

return res;

}

public static void main(String[] args) {

List<debtRecord> records = new ArrayList<>();

records.add(new debtRecord("Alex", "Blake", 2));

records.add(new debtRecord("Blake", "Alex", 2));

records.add(new debtRecord("Casey", "Alex", 5));

records.add(new debtRecord("Blake", "Casey", 7));

records.add(new debtRecord("Alex", "Blake", 4));

records.add(new debtRecord("Alex", "Casey", 4));

List<String> res = negativeAccounts(6, 3, records);

for(String r: res) {

System.out.print(r+" ");

}

System.out.println();

}

}