

1. Out of Date Software

Consider the following data in array of strings

```
Server1, Database, MySQL, 5.5
Server2, Database, MySQL, 5.1
Server3, OS, Ubuntu, 12.04
Server1, OS, Ubuntu, 12.04
Server2, OS, Ubuntu, 18.04
Server3, Language, Python, 2.6.3
```

Server1 has version 5.5 of MySQL installed, Server2 has version 5.1 installed, and Server3 has version 12.04 of Ubuntu installed. For the purposes of this program, assume that all version numbers are of the form X.Y or X.Y.Z where X, Y, and Z are made up of only digits.

Write a java program to display a list of software package names for which an out-of-date version (i.e. a version which is not the latest version) is installed on at least 2 different servers.

Thus, in this case, the output of your program should be:

Ubuntu

Because Ubuntu 12.04 is an out-of-date version (the latest version is 18.04), and it is installed on two servers (Server 3, and Server 1).

2. Customers who purchased at least one most discounted product

Consider the following data in array of strings

```
Rajan Patil, Aundh, 1, Phone Cover, Rs 170, Cash
Mohit Gupta, Baner, 1, Samsung Battery, Rs 900, Credit Card
Rajan Patil, Aundh, 3, Samsung Battery, Rs 1000, Cash
Nina Kothari, Baner, 4, Earphones, Rs 500, Credit Card
T Sunitha, Shivajinagar, 5, Earphones, Rs 550, Credit Card
Rohan Gade, Aundh, 10, Motorola Battery, Rs 1000, Credit Card
Rajan Patil, Shivajinagar, 21, Earphones, Rs 550, Credit Card
Rajan Patil, Aundh, 22, USB Cable, Rs 150, UPI
Meena Kothari, Baner, 23, USB Cable, Rs 100, Cash
Nina Kothari, Baner, 24, USB Cable, Rs 200, UPI
Mohit Gupta, Baner, 25, USB Cable, Rs 150, UPI
```

The data is the sales of products in different stores of a company in the previous month. Each line consists of 6 comma-separated fields. The first column is customer name, the second column is store location, the

third column is date (just day of the month is captured), the fourth column is product name, fifth is price, and sixth is payment type. Write program to processes the input data, finds out which items have sold for a discount (i.e. there was at least one sale of the item at a price lower than the highest price of the item), and then lists the names of customers who have bought at least one item at the most discounted price.

Your program should produce the names of the customers on the standard output, one per line. The output should not contain anything else.

For the sample input given above, your program should produce the following output:

Mohit Gupta

Nina Kothari

Meena Kothari

Explanation: For the Samsung Battery, Rs 900 is the most discounted price, and Mohit Gupta bought it. For the Earphones, Rs 500 is the most discounted price, and Nina Kothari bought it. For the USB cable, Rs 100 is the most discounted price, and Meena Kothari bought it. Note: For the Phone Cover and the Motorola Battery, there are no discounted prices.

3. At Least One Outdated Library

A company has a number of products that they sell to customers, and each product uses one or more internal libraries. All internal libraries are given version numbers like v1, v2, v3, ... etc. which are incremented each time the internal library team releases a new version of the library to the product teams

Consider the following data in array of strings which contains information about which versions of the various libraries are being used by the current release of each product of the company

Mail Server, Authentication Library, v6

Video Call Server, Authentication Library, v7

Mail Server, Data Storage Library, v10

Chat Server, Data Storage Library, v11

Mail Server, Search Library, v6

Chat Server, Authentication Library, v8

Chat Server, Presence Library, v2

Video Call Server, Data Storage Library, v10
Video Call Server, Video Compression Library, v3

Each line of the input consists of 3 comma-separated fields. The first field is a product name, the second field is a library name, and the third field is the library version number used by that product. The example given above indicates that the Mail Server uses version v6 of the Authentication Library, and the Video Call Server uses version v7 of the Authentication Library, and the Mail Server also uses version v10 of the Data Storage Library. and so on. For the purposes of this program, assume that all version numbers are of the form v<digits> where <digits> represents one or more decimal digits. Write a program to process the input data, figures out which is the latest version number of each library, and then prints the names of products which are using an outdated version (ie. a version that is not the latest version) of at least one library, to standard output (console). Thus, in case of the example input given above, the output of your program should be

Mail Server

Video Call Server

because Mail Server is using older versions of the Authentication Library and the Data Storage Library, and Video Call Server is using an older version of the Authentication Library.

4. Average marks scored across all subjects by the student with the lowest ID

Consider the following data

22, Data Structures, 45

23, English, 52

22, English, 51

26, Data Structures, 72

23, Data Structures, 61

24, English, 81

Each string consists of three fields: "Student ID," "Subject," and "Marks." "Student ID" and "Marks" are integers and "Subject" is a string that does not contain commas or newlines. There can be any number of students and up to 6 subjects.

Write a program to compute the average marks scored across all subjects

by the student with the lowest ID.

5. Highest Salary in Dept

```
22, Rajan Anand, Engineering, 1600000
23, Swati Patil, Testing, 800000
27, Vijay Chawda, Engineering, 800000
29, Basant Mahapatra, Engineering, 600000
32, Ajay Patel, Testing, 350000
34, Swaraj Birla, Testing, 350000
```

Each string consists of 4 fields: "Employee ID," "Name," "Department," and "Salary." Here, "Employee ID" and "Salary" are integers, while "Name" and "Department" are strings that do not contain commas or newlines.

Write a program to find the Employee IDs of the highest paid employee in each department. Specifically, processData should return a Dictionary where each key is the name of a department, and the value is the Employee ID of the employee in that department who has the highest salary. If your program is run with the input given above, it should produce this output:

Engineering: 22

Testing: 23

This is because Employee ID 22 (Rajan Anand) is the highest paid employee in the Engineering department, and Employee ID 23 (Swati Patil) is the highest paid employee in the Testing department.

Please make sure of the following:

6. Max Votes in Constituency

```
22, Ravi Pawar, Aundh, 1603
23, Suvarna Kale, Baner, 803
27, Vinod Chavan, Aundh, 809
29, Vasant Mahajan, Aundh, 617
32, Aarti Patil, Baner, 351
34, Swaran Bijur, Baner, 352
```

Each line consists of 4 fields: "Candidate ID," "Name," "Constituency," and "Votes." Here, "Candidate ID" and "Votes" are integers, while "Name" and "Constituency" are strings that do not contain commas or newlines.

Write a program to find the Candidate IDs of the candidate with highest votes in each constituency. Specifically, processData should return a Map where each key is the name of a constituency, and the value is the

Candidate ID of the candidate in that constituency who has the highest votes.

If your program is run with the input given above, it should produce this output:

Aundh:22

Baner:23

This is because Candidate ID 22 (Ravi Pawar) has the highest votes in Aundh, and Candidate ID 23 (Suvarna Kale) has the highest votes in Baner

Collections

7. Assignment 1

Instructions

Given two lists determine if the first list is contained within the second list, if the second list is contained within the first list, if both lists are contained within each other or if none of these are true.

Specifically, a list A is a sublist of list B if by dropping 0 or more elements from the front of B and 0 or more elements from the back of B you get a list that's completely equal to A.

Examples:

- A = [1, 2, 3], B = [1, 2, 3, 4, 5], A is a sublist of B
- A = [3, 4, 5], B = [1, 2, 3, 4, 5], A is a sublist of B
- A = [3, 4], B = [1, 2, 3, 4, 5], A is a sublist of B

- A = [1, 2, 3], B = [1, 2, 3], A is equal to B
- A = [1, 2, 3, 4, 5], B = [2, 3, 4], A is a superlist of B
- A = [1, 2, 4], B = [1, 2, 3, 4, 5], A is not a superlist of, sublist of or equal to B

8. Assignment 2

Determine if a word or phrase is an isogram.

An isogram (also known as a "nonpattern word") is a word or phrase without a repeating letter, however spaces and hyphens are allowed to appear multiple times.

Examples of isograms:

- lumberjacks
- background
- downstream
- six-year-old

The word *isograms*, however, is not an isogram, because the s repeats.

9. Assignment 3

An anagram is a rearrangement of letters to form a new word. Given a word and a list of candidates, select the sublist of anagrams of the given word.

Given "listen" and a list of candidates like "enlists" "google" "inlets" "banana" the program should return a list containing "inlets".

Instructions

Implement a doubly linked list.

Like an array, a linked list is a simple linear data structure. Several common data types can be implemented using linked lists, like queues, stacks, and associative arrays.

A linked list is a collection of data elements called *nodes*. In a *singly linked list* each node holds a value and a link to the next node. In a *doubly linked list* each node also holds a link to the previous node.

You will write an implementation of a doubly linked list. Implement a Node to hold a value and pointers to the next and previous nodes. Then implement a List which holds references to the first and last node and offers an array-like interface for adding and removing items:

- `push` (*insert value at back*);
- `pop` (*remove value at back*);
- `shift` (*remove value at front*).
- `unshift` (*insert value at front*);

To keep your implementation simple, the tests will not cover error conditions. Specifically: `pop` or `shift` will never be called on an empty list.

If you want to know more about linked lists, check [Wikipedia](#).

This exercise introduces [generics](#). To make the tests pass you need to construct your class such that it accepts any type of input, e.g. `Integer` or `String`.

Generics are useful because they allow you to write more general and reusable code. The Java [List](#) and [Map](#) implementations are both examples of classes that use generics. By using them you can construct a `List` containing `Integers` or a list containing `Strings` or any other type.

There are a few constraints on the types used in generics. One of them is that once you've constructed a `List` containing `Integers`, you can't put `Strings` into it. You have to specify which type you want to put into the class when you construct it, and that instance can then only be used with that type.

For example you could construct a list of `Integers`:

```
List<Integer> someList = new LinkedList<>();
```

Now `someList` can only contain `Integers`. You could also do:

```
List<String> someOtherList = new LinkedList<>()
```

Now `someOtherList` can only contain `Strings`.

Another constraint is that any type used with generics cannot be a [primitive type](#), such as `int` or `long`. However, every primitive type has a corresponding reference type, so instead of `int` you can use [Integer](#) and instead of `long` you can use [Long](#).

It can help to look at an [example use case of generics](#) to get you started.

