Laboratory work #6 | Collections

Standard collections

Write a program implementing described functionality using standard Java collection.

Create a class representing element of data having fields suggested in the case, methods for reading and changing values of that fields, constructor for initialization of newly created data element. Create class having at least field with collection of elements of data and methods for each operation, described in the case. Write main program using created classes.

Take a look at the examples if interaction with program from the variants 1-3.

Variants

1. Calculator

User enters the sequence of operands (numbers) and operators (characters of arithmetic operations). After each operand program prints current number of operation and result value. There is also a command to change current value to any previous result by its number.

Command retrieving previous result should get it from the collection with log of operations. Each record in the log of operations should have few fields: operation, second argument and corresponding result.

Right after the program is started, it displays usage tips.

```
V:\>java lab.Program
Usage:
        when first symbol on line is '>' — enter operand (number)
        when first symbol on line is '@' — enter operation
            operation is one of '+', '-', '/', '*' or
                    '#' followed with number of evaluation step
                    'q' to exit
>: 3
[#1] = 3
@: +
>: 5
[#2] = 8
@: /
>: 2
[#3] = 4
@: #2
[#4] = 8
@: *
>: 10
[#5] = 80
@: q
V:\>
```

2. Contact-list

Program shows hint with the instruction of using and prints menu of the numbered entries, each stating for the available commands: print all contacts, search for a contact, add new contact, exit from the program. To perform a particular command user should enter corresponding number of the menu entry.

New contact addition performed by entering the consecutive information elements such as: name, surname, telephone number, email and so on.

Contact searching could be performed by name or by phone number, or by name and surname, or by all fields at once. After selection of the search mode, user should enter desired string (part of the phone number or such), that would be used during search, and information of the corresponding conracts should be printed.

Example of the interaction with program:

```
V:\>java lab.Program
Enter the number of action and press [Enter]. Then follow instructions.

Menu:
1. View all contacts
2. Search
3. New contact
4. Exit
> 3
New contact
Name: Alex
Surname: Morgan
Phone: +71234567890
E-mail: Alex.Morgan@gmail.com
Contact created.

Menu:
1. View all contacts
2. Search
3. New contact
4. Exit
> 2
Search by
1. Name
2. Surname
3. Name and Surname
4. Phone
5. E-mail
> 3
Request: mor

Searching…

Results (1) :
#1  Name: Alex
Surname: Morgan
Phone: +71234567890
E-mail: Alex.Morgan@gmail.com

Menu:
1. View all contacts
2. Search
3. New contact
4. Exit
> 4
V:\>
```

3. Todo-list

Main menu of the program has following commands: create todo-item, search todo-items by tags, print number of actual todo-items, exit. User should enter number of the desired entry to execute corresponding command.

Item creation procedure requires item title, description, deadline date and tags. They should be entered by the user one by one. Tags entering considered finished on empty string.

Search by tags starts from the request of the keywords to search from the user. User enters them using space separator. Then, if some todo-item has one of the given tags, such item should be printed.

Actual todo-items output should be performed for the requested number of items considering them in the order of deadline date growth.

Example of the interaction with program:

```
V:\java lab.Program
Enter the number of action and press [Enter]. Then follow instructions.
Menu:
        1. Add task
        2. Search task
        3. Last tasks
        4.Exit
> 1
New task
        Title: some task
```

```
        Description: some text
        Deadline: 10.11.2012
        Tags (finish on empty line)
                1: tagA
                2: tagB
                3:
Menu:
      1. Add task
      2. Search task
      3. Last tasks
      4.Exit
> 2
Search tasks by tag: tagC

No such tasks

Menu:
      1. Add task
      2. Search task
      3. Last tasks
      4.Exit
> 3
Actual tasks:
      1. Title: some task
      Description: some text
      Deadline: 10.11.2012
      Tags: tagA, tagB

Menu:
      1. Add task
      2. Search task
      3. Last tasks
      4.Exit
> 4
V:\
```

4. Library catalog

Information about book consists of title, author name, annotation, ISBN and publication date.

Main menu of the program consists of numbered entries: add book to the catalog, get book information by its ISBN, search book by any keywords, exit.

Search operation should display brief book descriptions (without annotation) in the order of decreasing the number of keywords found. If a particular keyword found in the annotation, reflect this fact during the output.

5. Playlist manager

When the program is launched, user is asked to enter the number of command from the list displayed. List of commands has following entries: search composition by a certain criterion, print information about all compositions from the playlist, add new composition to the playlist, remove composition from the playlist, exit the program.

The search criteria can be: name author of composition or the title of composition.

As a result of the search, the list of the compositions should be shown, having "number – author – title – duration" for each composition. Removal of composition requires its number. Addition requires entering all information about the composition.

6. Shopping-list

Shopping list contains list of the purchases performed. Each item has name, comment, amount of credits spent, and date of purchase. During interaction with program user able to perform one operation in turn by entering the number of menu entry from the tip displayed. After each operation, the tip displayed once more.

Menu entries are: add new purchase info, remove purchase by its number, print purchases in the specified range of dates. Printing of information about each purchase accompanied by its number in the list of all purchases.

7. Cookbook

Each recipe in the cookbook represented with title, brief description, and number of steps to cook.

Interaction between user and program organized by iterative displaying menu with available entries to do with the state of cookbook and asking the user to select one of them. Menu has following entries: add new recipe, delete recipe by name, search for the recipe.

Recipe search performed by any word contained in its title, description or steps and results with list of detailed description of corresponding recipes.

8. Gradebook

Gradebook is the book with information about students' marks: it helps to the teacher (and student) answer two the question about the marks by the subjects.

Technically, the book – is a set of records that combines information about the student (name), subject and mark.

Interaction between user and program organized by iterative displaying menu with available entries to do with the gradebook and asking the user to select one of them. Menu has following entries: add new item, delete the item by student's and subject name, search by the subject (to display marks by this subject for all students).

9. Meteorological log

It is a journal with information about the meteorological data: temperature, pressure, precipitation (rain, snow) in particular date/time.

Interaction between user and program organized by iterative displaying menu with available entries to do with the gradebook and asking the user to select one of them. Menu has following entries: add new item, delete the item by date, show the list of data by month.

10. Discount journal

It is a journal with information about your discounts in different markets (like Stocard application for smartphones). It stores information about the shop, size of discount and the discount's expiration date.

Interaction between user and program organized by iterative displaying menu with available entries to do with the gradebook and asking the user to select one of them. Menu has following entries: add new item, delete the item by shop, show the list of your discounts sorted by the alphabet.