

QUIZ 4

Issue Date : 17.05.2024 - Friday

Due Date : 19.05.2024 - Sunday (23:00)

Advisor: Umut Güngör, R.A. Aslı TAŞGETİREN

Programing Language : Java 8



1 Introduction

Generics in Java are a powerful feature that allows us to create classes, interfaces, and methods that operate on specified data types while providing compile-time type safety. They enable us to design reusable, type-safe code that can work with different data types without sacrificing type safety. The primary benefit of using generics is to write code that is more flexible and less error-prone by providing compile-time checks on the data types used. This helps in catching type-related errors early in the development process and promoting more robust and maintainable code. Overall, Generics play a crucial role in Java programming by facilitating code reuse, flexibility, and type safety in a concise and efficient manner.

2 Inventory Management System

You are going to implement an Inventory Management System for a company. There are three types of items: Books, Toys, and Stationery. You can do some operations with these items in Inventory Management System. There are five operations: Add an item, remove an item, search an item by its barcode, search an item by its name, and display items. In add operation, you will add one of the three items to the system. Similarly, in remove operation, if there is an item with specified barcode, you will remove this item from the system. In search operations, you will search the system by specified criteria, and if there is an item with that criteria, you will write it to the output file. Finally, in display operation, you will list all of the items in the system according to their types. So, first, you will list all books, then all toys, then all stationery. In your implementation, you must use Generics.

3 Input Output Format

You will have one input file, which is `commands.txt`, and one output file which is `output.txt`.

3.1 `commands.txt`

The input files includes operations on the system. If a line starts with **ADD**, it indicates the add operation which adds an item to the system. If the item in the add operation is **Book**, the line includes book's name, author, barcode and price. If the item in the add operation is **Toy**, the line includes toy's name, color, barcode and price. If the item in the add operation is **Stationery**, the line includes stationerie's name, kind, barcode, price. If a line starts with **REMOVE**, it indicates the remove operation which removes an item from the system if there is an item with specified barcode. If a line starts with **SEARCHBYBARCODE**, it indicates the search operation which displays an item if there is an item with specified barcode. If a line starts with **SEARCHBYNAME**, it indicates the search operation which displays an item if there is an item with specified name. Finally, there is a display command which shown as **DISPLAY**. In display command, you should show all of the items in the system.

Format of `commands.txt`:

ADD[tab]**Book**[tab]**name**[tab]**author**[tab]**barcode**[tab]**price**

ADD[tab]**Toy**[tab]**name**[tab]**color**[tab]**barcode**[tab]**price**

ADD[tab]**Stationery**[tab]**name**[tab]**kind**[tab]**barcode**[tab]**price**

REMOVE[tab]**barcode**

SEARCHBYBARCODE[tab]**barcode**

SEARCHBYNAME[tab]**name**

DISPLAY

Example lines form `commands.txt`:

```
ADD Book Tutunamayanlar Oguz 123 150.99
ADD Toy puzzle blue 124 100.49
ADD Stationery stationery1 pencil 125 50.99
ADD Book Sefiller Hugo 126 199.90
ADD Toy car red 127 10.95
ADD Stationery stationery2 eraser 128 12.12
ADD Toy lego purple 129 4.55
DISPLAY
REMOVE 123
SEARCHBYBARCODE 127
SEARCHBYBARCODE 1
DISPLAY
SEARCHBYBARCODE 131
SEARCHBYNAME stationery1
REMOVE 2
SEARCHBYNAME Seas
ADD Book Sapiens Harari 130 200.99
DISPLAY
```

3.2 output.txt

The output file format should be same as the following example. Except the add operation, you should indicate the command and its output. In add operation, you are not going to write anything to the output file.

Example lines form output.txt:

```
INVENTORY:
Author of the Tutunamayanlar is Oguz. Its barcode is 123 and its price is 150.99
Author of the Sefiller is Hugo. Its barcode is 126 and its price is 199.9
Color of the puzzle is blue. Its barcode is 124 and its price is 100.49
Color of the car is red. Its barcode is 127 and its price is 10.95
Color of the lego is purple. Its barcode is 129 and its price is 4.55
Kind of the stationery1 is pencil. Its barcode is 125 and its price is 50.99
Kind of the stationery2 is eraser. Its barcode is 128 and its price is 12.12
=====
REMOVE RESULTS:
Item is removed.
=====
SEARCH RESULTS:
Color of the car is red. Its barcode is 127 and its price is 10.95
=====
SEARCH RESULTS:
Item is not found.
=====
INVENTORY:
Author of the Sefiller is Hugo. Its barcode is 126 and its price is 199.9
Color of the puzzle is blue. Its barcode is 124 and its price is 100.49
Color of the car is red. Its barcode is 127 and its price is 10.95
Color of the lego is purple. Its barcode is 129 and its price is 4.55
Kind of the stationery1 is pencil. Its barcode is 125 and its price is 50.99
Kind of the stationery2 is eraser. Its barcode is 128 and its price is 12.12
=====
SEARCH RESULTS:
Item is not found.
=====
SEARCH RESULTS:
Kind of the stationery1 is pencil. Its barcode is 125 and its price is 50.99
=====
REMOVE RESULTS:
Item is not found.
=====
SEARCH RESULTS:
Item is not found.
=====
INVENTORY:
Author of the Sefiller is Hugo. Its barcode is 126 and its price is 199.9
Author of the Sapiens is Harari. Its barcode is 130 and its price is 200.99
Color of the puzzle is blue. Its barcode is 124 and its price is 100.49
Color of the car is red. Its barcode is 127 and its price is 10.95
Color of the lego is purple. Its barcode is 129 and its price is 4.55
Kind of the stationery1 is pencil. Its barcode is 125 and its price is 50.99
Kind of the stationery2 is eraser. Its barcode is 128 and its price is 12.12
=====
```

Execution and Test

While testing the input file's and output file's relative path (commands.txt and output.txt) should be given as arguments, *in that order*. You should do the following steps:

- Upload your java files to your server account (dev.cs.hacettepe.edu.tr)
- Compile your code (javac8 *.java, or javac8 Main.java)
- Run your program (java8 Main commands.txt output.txt)

- Control your output file output.txt, it should have same data and format as the one given to you with your quiz.

Grading Policy

Task	Point
Correct output	100*
Total	100

*Even though it looks like only getting correct output will be enough to get 100% in this assignment, you are still required to use Generics and other rules stated in Notes and Restrictions section. Additionally, you must have as little code repetition and replication as possible, you will be graded on that as well. If you do not follow these rules you will face point deductions even if your output is 100% correct.

Submit Format

File hierarchy must be zipped before submitted (Not .rar, only .zip files are supported by the system)

```
- b<studentid>.zip
  - <src>
    - Main.java, *.java
```

Notes and Restrictions

- Since the main objective of this experiment is to gain experience on Generics in Java, any solutions that does not use Generics will not be accepted. Additionally, you must be careful to follow DRY(Don't Repeat Yourself) principle in this quiz.
- Do not miss the submission deadline.
- You must obey given submit hierarchy and get score (1 point) from the submit system, if not, as stated in BBM 104 Laboratory Rudiments you will lose 10% of your grade.
- Save all your work until the quiz is graded.
- Compile your code on DEV server before submitting your work to make sure it compiles without any problems on our server.
- Source code readability is a great of importance for us. Thus, write READABLE SOURCE CODE, comments and clear MAIN function. This expectation will be graded as "clean code".
- Regardless of the length, use UNDERSTANDABLE names to your variables, classes and functions. The names of classes, attributes and methods should obey Java naming convention. This expectation will be graded as "coding standards".

- You can ask your questions through course's piazza group and you are supposed to be aware of everything discussed in the piazza group. General discussion of the problem is allowed, but DO NOT SHARE answers, algorithms, source codes and reports.
- All assignments must be original, individual work. Duplicate or very similar assignments are both going to be considered as cheating.