Hoàng Văn Khang – 20210466

OOP lab week 4 – Inheritance and Polymorphism

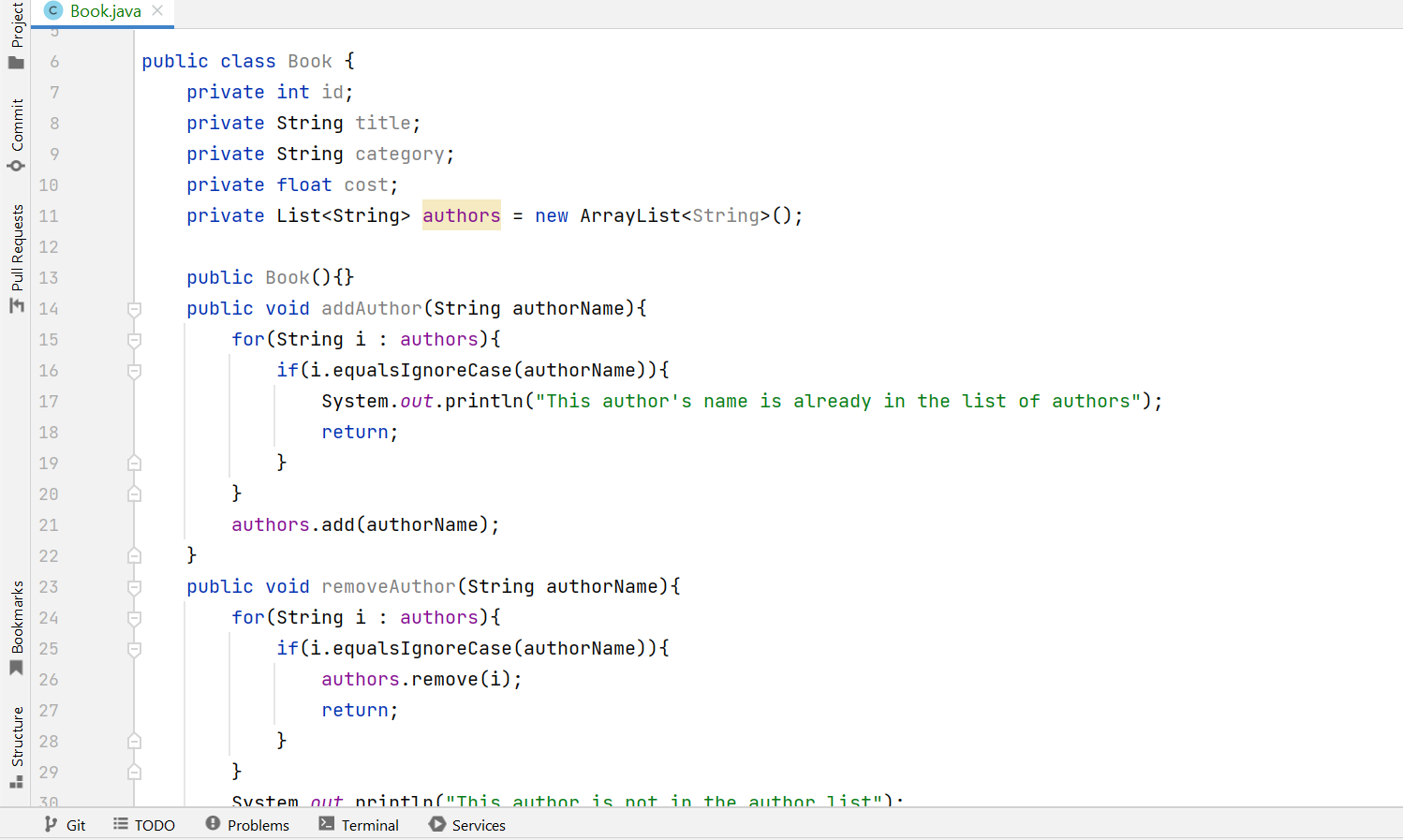
EX1: Import the existing project into the workspace

<done in code>

EX2: Additional requirements of AIMS

<done in code>

EX3: Creating the Book class



EX4: Creating the abstract Media class

A screenshot of a computer program

Description automatically generated with low confidence

A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

EX5: Creating the CompactDisc class

A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer program

Description automatically generated with low confidence

A screenshot of a computer program

Description automatically generated with medium confidence

EX6: Create the Playable interface

A screenshot of a computer code

Description automatically generated with low confidence

A picture containing text, screenshot, font, line

Description automatically generated

A picture containing text, screenshot, font, line

Description automatically generated

A screen shot of a computer program

Description automatically generated with low confidence

A picture containing text, font, screenshot, line

Description automatically generated

A screen shot of a computer code

Description automatically generated with low confidence

EX7: Update the Cart class to work with Media

A screenshot of a computer

Description automatically generated

EX8: Update the Store class to work with Media

A screenshot of a computer

Description automatically generated

EX9: Constructors of whole classes and parent classes

EX10: Unique item in a list

A picture containing text, screenshot, display, software

Description automatically generated

A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer code

Description automatically generated with low confidence

EX11: Polymorphism with toString() method



EX12: Sort media in the cart

A screen shot of a computer code

Description automatically generated with low confidence

A picture containing text, screenshot, font, display

Description automatically generated

Blue text on a white background

Description automatically generated with medium confidence

A screenshot of a computer program

Description automatically generated with low confidence

Question: Alternatively, to compare items in the cart, instead of using Comparator, we can use the Comparable interface and override the compareTo()method. You can refer to the Java docs to see the information of this interface.

Suppose we are taking this Comparable interface approach.

- What class should implement the Comparable interface? -> Media

- In those classes, how should you implement the compareTo()method be to reflect the ordering that we want?

-> compareTo(Media media2) : we use “this” keyword to refer the media1, and compare it to media2

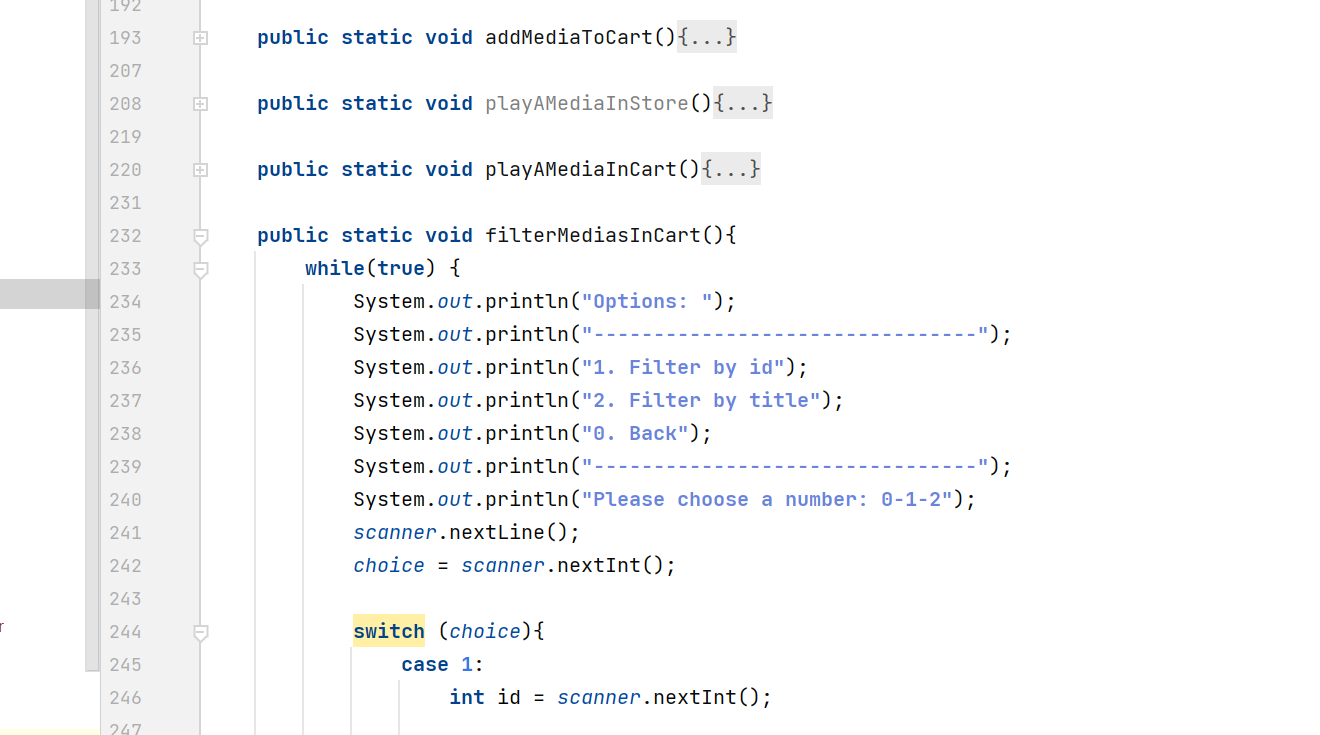
- Can we have two ordering rules of the item (by title then cost and by cost then title) if we use this Comparable interface approach?

-> no, if we do that, they would have the same method signature, and the system will not know to choose which to implement

- Suppose the DVDs has a different ordering rule from the other media types, that is by title, then decreasing length, then cost. How would you modify your code to allow this?

-> we can write a new part for DVDs the same as Media, but we should use Comparator. For example: class DVDComparator implements Comparator<DigitalVideoDisc>

EX13: Create a complete console application in the Aims class



Check for the code since there are many added methods.