PRACTICAL EXERCISE

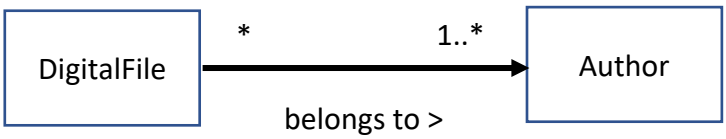
Create a Project called **DigitalMedia** in IntelliJ with the components detailed below.

1. The "model" package

Create a package called **model**. Inside this package you have to create the following class structure:

- Create an interface **Sizable** with a method:
  - getSize()
- Create an abstract class called **DigitalFile** to store information about digital files.
  - Its attributes will be:
    - KB
    - format (for example: "MPEG4")
    - Year of creation
  - You will have to create getters and setters for all these attributes and a constructor to initialize all of them.
  - Define a *toString* method to represent the file information in the format that you prefer
- Create another class called **Image** that inherits from **DigitalFile.**
  - It will have some more attributes:
    - width
    - height
    - compression (the type of compression used)
  - You will have to create getters and setters for these attributes and a constructor for all of them.
  - Define a *toString* method to represent the image information, using parent's method to complete the code.
  - It must implement the **Sizable** interface; the size will be *width * height*
- Create a class called **Sound** that inherits from **DigitalFile**
  - It will have these new attributes:
    - stereo (boolean)

        - kbps (for example: 192)
        - length (in seconds)
    - Create the getters, setters, constructor and *toString* method, similar to the previous classes.
- Create a class called **Video** that inherits from **Image.**
    - It will have these new attributes:
        - codec (for example: "H.264")
        - length (in seconds)
    - Create the getters, setters, constructor and *toString* method, similar to the previous classes.
- Create a class called **Author** to store information about the authors of the digital files. The relationship between authors and digital files is as follows, and you must implement it according to this piece of diagram, adding the necessary elements to the affected class(es).



    - The attributes of this class will be:
        - Author's name
        - Author's country (for instance, Spain, France)
    - Besides, we also need the constructor, getters and setters and a *toString* method to show author's name followed by his/her citizenship between parentheses. For instance: "John Doe (USA)". The authors of each digital file must be also shown in the *toString* method of that object.

2. The "main" package

Create a new package called **main**, and in this package, you have to create a class called **Main** where we will have the main program:

We will have an array of 9 DigitalFile elements (3 for each class) you will create these elements in the code, associating the corresponding author(s) to each element (authors must be created in the code as well). After that you will show this menu to the user:

1. Show all elements
2. Show all elements ordered by KB in ascending order (**implementing Comparable interface)**
3. Show only one kind of element (image, sound or video)
4. Normalize author names (The names of the authors will be in lowercase except the first character that will be in uppercase and every other character after a non-alphabetic character will be in uppercase).
5. Show all elements ordered by:
   - Format (ascending)
   - Year of creation (descending)

Implement an **anonymous comparator** for each type in this menu option

Remember to reuse code whenever possible.

Qualification criteria

Packages' structure	0,5
Interface Sizable	0,5
Abstract class DigitalFile	0,75
Image class	0,75
Sound class	0,5
Video class	0.5
Author class	0,5
Association between authors and files	0,5
Show all elements	0,5
Show all elements ordered by KB	1
Show only one kind of element	1
Normalize author names	1
Reusing of code, code cleanliness	1
Show all elements ordered by different criteria (optional)	1
Total	10