LAB 3: Controlling the equipment inventory of your band

(20 points) Your neoclassical jazzhopmetal band is becoming more popular, which is great! But, all this touring is hard. To make sure that you don't forget any of your equipment after a show, or accidentally take someone else's, you decide to create an application that helps you keep track of your equipment. Basically, for every piece of equipment that you retrieve from the venue and stow into the trunk of the tour bus, you mark it as such in the application. The application has the following requirements:

- The inventory must keep track of all the equipment that you have. This includes:
  - The type of equipment
  - The number of each type that you own (inInventory)
  - The number of each type that you already loaded in the bus (inBus)
- Some types of equipment (in particular, musical instruments) are easy to damage so they need to be wrapped before we can leave.
- Your application should be able to show:
  - Which equipment (and how many) you own
  - Which equipment (and how many) is missing from the bus
  - Which equipment still needs to be wrapped

You own the following types of equipment:

- Two microphones
- Four guitars
- Twelve chairs (for the traditional jazzhopmetal musical chairs game)

A basic framework is provided for the application, but it is incomplete. For this assignment, you have to finish the framework. You are allowed to add fields, variables, classes and interfaces as you wish, but you cannot change the signatures of any classes, fields or methods in the framework.

Some hints for implementation:

- You can assume that all equipment of the same type is wrapped at once. Microphones are considered instruments as well.
- Make sure to check when adding equipment to the bus if you are not accidentally taking someone else's equipment. Luckily, it is accepted among jazzhopmetal bands to share equipment. So you only have to count the equipment (e.g. if you are trying to stow three microphones in the trunk of the bus, you can pick any two and just leave one out).
- Clearly, the different types of equipment share some characteristics and behaviour (e.g., turnOn()/turnOff()), but they also have their own characteristics and behaviour (e.g., makeSound()). However, in this assignment you don't have to implement any of that behaviour - the only important thing is that the types can distinguish themselves.
- Keep in mind that in the future you may need to add new types of equipment. So you want to design your application in such a way that adding new equipment would have as little impact as possible (it is OK if you would have to add some code for that though – it does not have to be fully automated).
- Some methods that may be helpful to look at (or perhaps, override) are the getClass(), toString() and equals() methods in the Object class.
- Make sure to encapsulate your fields and methods correctly, and make sure to use inheritance.

- If you are looping through your inventory, make sure to not keep doing it unnecessarily (e.g., after you already found the element you were looking for).

**Please submit:**
**1) A zip file containing your code and a PDF with the answers to the questions above.**
Name the file 'FirstName_ID_lab_asg3.zip' and keep the exact same file structure as the zip that was provided for the assignment. For example,
Filename: Cor-Paul_1234567_lab_asg3.zip

```
|------- solution.pdf
|------- src
|        |------- ece325_lab_assignment3
|        |        |------- *.java
```

**2) A screencast/movie that shows the following steps:**
- Open your eClass with your name shown
- Open your IDE
- Show your code briefly
- Execute your code and show the results of question 3 of this assignment

**Please do not modify any of the names/methods we've defined in the provided *.java files.**