Lab 09: Inheritance

Due date: By the end of your lab session of the week of Monday, October 24th

Goals

- Learning to work with inheritance in user defined classes
- Using `ArrayList`

Description

In this lab, you are going to implement a music player simulator that provides basic functionality for different kinds of music players: a CD Player and an Ethernet Player. In order to do this, you will create a base **MusicPlayer** class in which you define the general functionality of a music player. For example, you can **turn on** the music player, **turn off** the music player, **play** music, and **adjust volume** of the player. Then, you will create two sub-classes **CDPlayer** and **EthernetPlayer** each of which *extends* the **MusicPlayer** class.

Download the skeleton from the wiki page and complete the missing methods.

Details

The diagram in the next page shows the fields in each class as well as the relationship among the classes. As you can see in the diagram, some methods in the subclasses of MusicPlayer **override** their corresponding methods in the MusicPlayer class. Remember that when one of these overridden methods is called on an object of one of these subclasses, the body of the **subclass** method will be executed. **All methods and fields from the parent class are inherited by its child classes.**

The field **playlist** in the class MusicPlayer is an ArrayList of Strings of a default/initial size of 10. It is used to maintain the songs you can play. A music player has a status, it can be "OFF" or "ON" or "PLAYING" (since these are constants, they are defined as static final int). The volume is an integer value which can be adjusted by the user; it has a maximum value of 100 and a minimum value of 0

Whenever a play method is called, the status of the music player should be set to "PLAYING" and the name of the song that is playing should be printed to stdout. All Music Players have a method to play a specific track `play(int trackNumber)` and in the case of a CD Player, a method to play the current track: `play(),` skip to the next track or return to the previous track.

The methods `turnOff` and `turnOn` are overriden by the child classes. A CD Player should set its current track back to 0 every time it is turned off or turned back on. An Ethernet player should connect to the internet when it is turned on and disconnect from it when it is turned off.

The method `download(String trackName)` looks for the track name in the download list and adds it to the playlist.

MusicPlayer
- `playlist`: *ArrayList of Strings, initially fixed* - `status`: *int (ON, OFF or PLAYING)* - `volume`: *int*
- `public MusicPlayer()` - `public void play(int trackNumber)` - `public void turnOn()` - `public void turnOff()` - `public void increaseVolume(int increment)` - `public void decreaseVolume(int increment)` `-public void showPlaylist()`

CDPlayer extends MusicPlayer	**EthernetPlayer extends MusicPlayer**
- `thisTrack`*: private int* - `deviceID`*: private int*	- `deviceID`: *private int* - `connStatus`: *private int (CONNECTED or NOT_CONNECTED)* - `downloadList`: *ArrayList of Strings, initially fixed*
- `public CDPlayer(int id)` - `public void turnOn()`: *overrides* - `public void turnOff()`: *overrides* - `public void play()` - `public void previousTrack()` - `public void nextTrack()`	- `public EthernetPlayer(int id)` - `public void turnOn()`: *overrides* - `public void turnOff()`: *overrides* - `public void download(String trackName)` - `public void addToPlaylist(String trackName)` - `public void deleteFromPlaylist(String trackName)`

Run the main program in MusicPlayer to test your code. An example of a correct execution can be observed below

```
champu:lab09skeleton marialeonor$ java MusicPlayer

CDPlayer
Player ON

CDPlayer Playlist:
0: Words
1: Stairway To Heaven
2: Dream On
3: Thrift Shop
4: Buzzcut Season
Volume: 20
Playing: Words
Playing: Stairway To Heaven
Playing: Words
Track 21 doesn't exist
Volume: 0
Player OFF

EthernetPlayer
Player ON
the playlist has: 5 songs
Volume: 50
Playing: Dream On
Track 21 doesn't exist
Added Broken to playlist
Added Sweet Child'O Mine to playlist
Removed Broken from playlist
Track haha not available for download
Downloaded Dark Horse
Downloaded Royals
Downloaded Let Her Go
Downloaded The Fox
Downloaded Counting Stars
Volume: 100

EthernetPlayer Playlist:
0: Words
1: Stairway To Heaven
2: Dream On
3: Thrift Shop
4: Buzzcut Season
5: Sweet Child'O Mine
6: Dark Horse
7: Royals
8: Let Her Go
9: The Fox
10: Counting Stars
Player OFF
```

Turning in Your Work

You **must** turn in your "lab09" directory before leaving the lab session. Change your current directory to "cs180" and execute the following command:

```
turnin -c cs180=COMMON -p lab09 lab09
```

Rubric

- 15 points MusicPlayer
  - 5 points: showPlaylist
  - 2 points: play
  - 2 points: turnOn
  - 2 points: turnOff
  - 2 points: increaseVolume
  - 2 points: decreaseVolume
- 15 points CDPlayer
  - 5 points: Constructor
  - 2 points: turnOn
  - 2 points: turnOff
  - 2 points: nextTrack
  - 2 points: previousTrack
  - 2 points: play
- 20 points EthernetPlayer
  - 2 points: turnOn
  - 2 points: turnOff
  - 8 points: download
  - 4 points: addToPlaylist
  - 4 points: deleteFromPlaylist