

ENSF 409  
Winter Semester 2021  
Assignment 06

**Assignment Instructions**

Complete and submit exercise 14.3 (Lesson 14). This assignment is due on March 01, 2021 by 5:59 PM Mountain Time.

## General Instructions

### Academic Integrity and Collaboration

This is an individual assignment. Your submission must be your own original work. You may not work with others to complete the assignment, although you may collaborate on exercises which are not submitted. It is not academic misconduct to ask for help on the discussion boards, but you may not copy code or complete answers from your peers.

### Submission

You must submit your assignment in the specified format. Due to the number of students in the course, we are using automated grading. If your submission does not have the correct folder structure and file names, it will not be marked. This is a strict requirement.

File and directory names are case-sensitive.

You must submit your assignment to the appropriate D2L dropbox folder. You may submit multiple times before the assignment deadline, but only the last submission will be graded. Previous uploads are discarded by D2L. Therefore, each upload must be a complete submission. Do not submit multiple files across separate submissions.

The assignment must be submitted as a single zip folder named with your student ID. Furthermore, when the file is unzipped, the contents must be in a folder with the same name. You may wish to verify that your zip file has been correctly generated (includes the student ID directory, contains all files). You can do this by copying the zip file into another folder, unzip it, and examine the structure.

Within the folder with your ID number, you must create a subdirectory for each exercise within the assignment. Use lowercase and employ a `_` to separate sub exercise numbers. For example, Exercise 1.3 should be in a folder `exercise1_3`.

Below is an example for submitting Assignment 1, assuming a student ID of *1234765*.

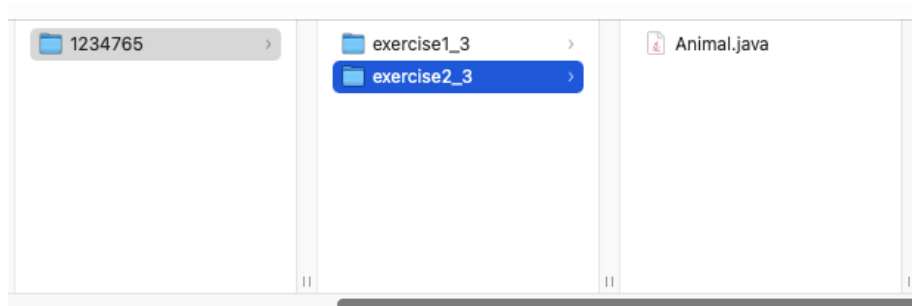


Figure 1: The folder structure for Assignment 1

```
[ENSF409 > ls
1234765
[ENSF409 > zip -r 1234765.zip 1234765
  adding: 1234765/ (stored 0%)
  adding: 1234765/exercise2_3/ (stored 0%)
  adding: 1234765/exercise2_3/Animal.java (stored 0%)
  adding: 1234765/exercise1_3/ (stored 0%)
  adding: 1234765/exercise1_3/Hello.java (stored 0%)
[ENSF409 > ls
1234765          1234765.zip
[ENSF409 >
```

Figure 2: Creating the zip file in Mac/Linux

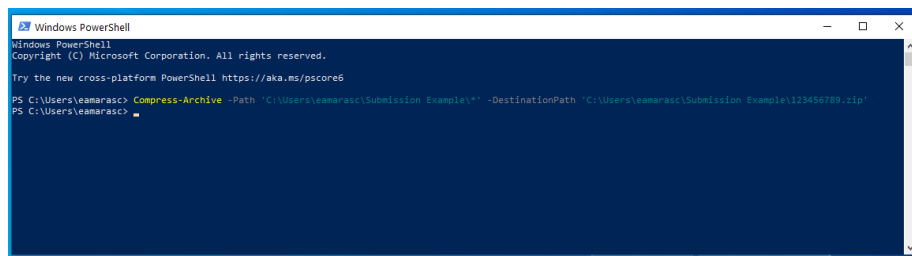


Figure 3: Creating the zip file in Windows

## Evaluation

Code which does not compile will not receive any points. Your code must compile and execute correctly to receive full marks. Assignments are graded using OpenJDK 11. While we do not anticipate that any of the assignments in this course will execute differently in any version of Java from 8 on, you should use this version for optimal results.

When style conventions are introduced in lessons, all subsequent submissions should adhere to the conventions.

## Deadline

All homework assignments are due at 17:59 (5:59 PM) Mountain Time. You are responsible for the conversion between Mountain Time and your local time zone, if applicable. Be aware that the switch from Mountain Standard Time to Mountain Daylight Time occurs during the term.

It is recommended that you do not leave your submission until the last minute in case of technical issues. Once the dropbox folder closes, you will not be able to submit your assignment. Late submissions will not be accepted.

If there is a technical problem and you are unable to submit via the dropbox, do not email your assignment. The University of Calgary email system will remove zip files. Instead, you may email a link to download your .zip folder from OneDrive. The email must be received by Dr. Barcomb and Dr. Marasco before the assignment deadline and the OneDrive folder should not show any changes to the .zip file after the deadline has passed.

# ENSF 409

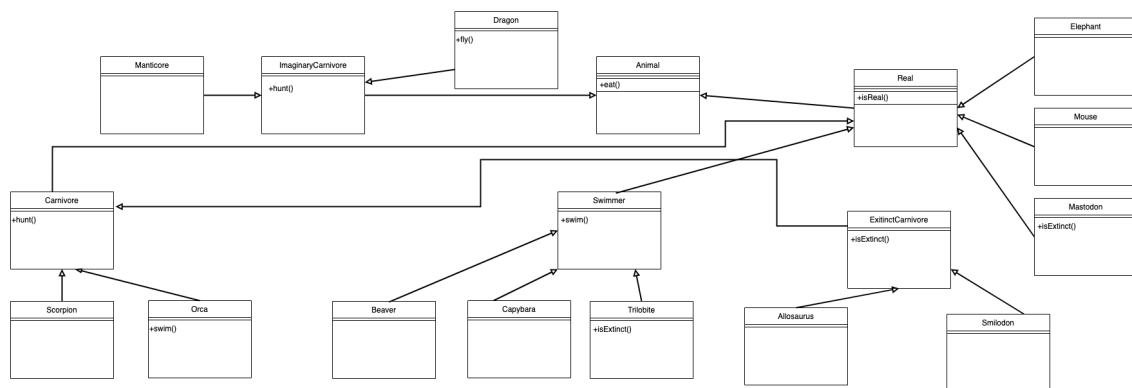
## Exercises - Lesson 13

The following exercises are described in the video for Lesson 13.

### Exercise 13.1

1. Modify the bestiary UML diagram in Lesson 12 to incorporate interfaces.
2. Ensure that each method only appears once in the diagram.
3. Prefer inheritance over interfaces.

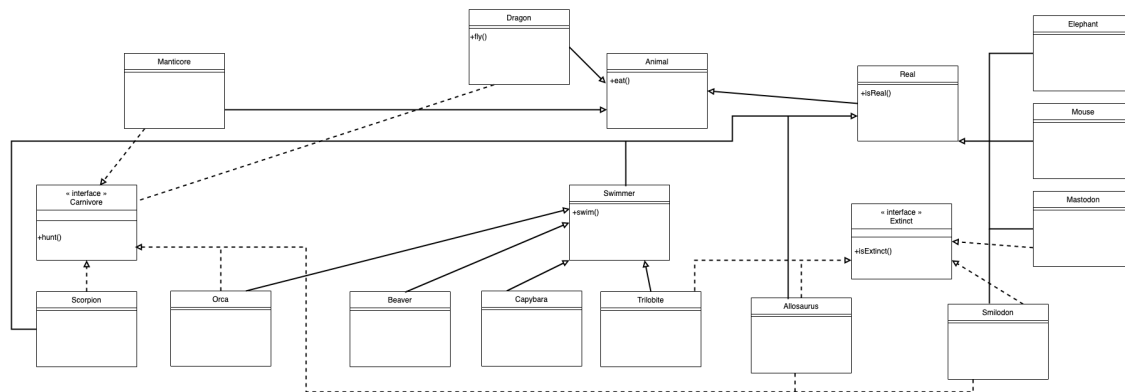
The UML diagram can be found in the repository under Lesson12/04\_Bestiary.



## Exercise 13.2

6

1. Implement the UML diagram provided in the repository as 03\_BestiaryUML.
2. Check your code against the example provided in 04\_Bestiary.



# ENSF 409

## Exercises - Lesson 14

The following exercises are described in the video for Lesson 14.

### Exercise 14.1

1. Implement a Character ArrayDeque.
2. Convert a Character array to ArrayDeque.
3. Use several different methods to add elements to the collection.
4. Use several different methods to inspect and print elements, including descendingIterator.
5. Use several different methods to remove elements from the collection.

- What is ArrayDeque good at?
- What is ArrayDeque bad at?
- When would you use ArrayDeque?

### Exercise 14.2

1. Read the documentation for HashMap.
2. Look at the code in the repository.
3. Without running the code, determine what the output should be.

- Were you able to predict the output based on the documentation?
- If you were to add HashMap to the reference table presented earlier, what values would you fill in for order maintained, duplicates, allows null, and thread-safe?
- When would you use a HashMap?

### Exercise 14.3

1. Implement the program shown in the UML diagram
2. Use the code in `MyOutput.java` to test your code

3. Compare your output with the output in `output.txt`
4. Your submission should be named `MyOutput.java` and the file should contain all your code
5. You may modify the `MyOutput` class; this is not graded

- Assumptions
  - Each log line consists of a valid IP address, a valid date/time, a valid action, a device, a room, and a building, and is formatted as shown in the example
  - Room, building, and device can consist of any English words (letters and spaces only, no hyphens or numbers)
  - IPv4 addresses consist of four numbers separated by periods; each number ranges from 0-255 inclusive
  - Only actions contained in the `Action` enumeration can appear in the log
  - Tests will only use good data (but not the same data in the example)
- If your code is working properly, it will produce exactly the same output as shown in the output file, provided you do not change the existing code
- Anything public will potentially be tested, not only the methods shown in the example main.



# ENSF 409

## Exercises - Lesson 15

The following exercises are described in the video for Lesson 15.

### Exercise 15.1

1. Create a generic method which has the characteristics specified on the right.
2. You can assume that only objects which implement `toString()` will be specified.

- Has a constructor which can be called with one argument of any type
  - The constructor adds the argument four times to a data structure
- Overloads the constructor with a zero argument version
  - The constructor adds the string `String "Hello, ENSF 409"` four times to a data structure
- Provides a method `arrayConcat`
  - Accepts an array index; if no array index is supplied, 0 is used by default
  - This method should have default behavior if the index supplied exceeds the bounds of the array (`IndexOutOfBoundsException` error)
  - Returns a `String` consisting of all the elements of the array, from the provided index to the end, concatenated, separated by the `#` character (no whitespace should be introduced)
- Provides a method `arrayCrop`
  - Accepts two integers as arguments, corresponding to starting and ending array indices. The ending index is inclusive.
  - Returns a `String` consisting of all the elements between the two indices specified, concatenated, separated by the `#` character (no whitespace should be introduced)
  - If the ending integer is smaller than the starting integer, switch the two (use the start as the end, and the end as the start)
  - If either integer is not a valid index, return `"Fail"`
  - If the two integers are the same, return `"Match"`