CMSC 131
Object-Oriented Programming I

ArrayList, File IO

Dept of Computer Science
University of Maryland College Park

Slides based on material provided by Ben Bederson, Bonnie Dorr, Fawzi Emad, David Mount, Nelson Padua-Perez, Jan Plane, Pedram Sadeghian, and various TAs over the years

© Department of Computer Science
UMD
Announcements

- Remember that there is a deadline to address grading concerns for each graded assignment/test/quiz, etc. Information can be found on the class web page.
- We take academic integrity matters seriously.
- Do not post project implementations online.
ArrayList

- A class in the Java class library that implements a list
- It is part of the java.util package, and therefore an appropriate import statement is required (e.g., import java.util.ArrayList;)
- An ArrayList holds references to objects.
  - We need to specify the kind of object the ArrayList will store.
  - If we are storing any primitive type then we need to use the appropriate wrapper (e.g., Integer, Double, Float)
- You specify the class in between < and > after ArrayList
- Example: ArrayListEx1.java, ArrayListEx2.java
- The ArrayList class provides a toString method that calls the toString() for each of the elements in the list, separating each string with commas and surrounding the final string with [ ]
ArrayList Methods

- **ArrayList Default Constructor** - Initializes a list of size 0
- **add** - Adds reference to the end of the list
  - An ArrayList does not store an object; **it stores a reference to an object**
  - An ArrayList **does not make a copy of the object associated with the provided reference**
- **get(int i)** - Returns the reference at index i.
  - The first reference is at index 0, the second at index 1, etc.
- **size()** - Returns the number of references in the ArrayList
- **remove(int i)** - Removes the reference at index i.
  - After removing a reference, references are shifted to close the gap
- **clear()** - Removes all the references from ArrayList
- Java API Entry
  - [https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/ArrayList.html](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/ArrayList.html)
**ArrayList and Sorting**

- You can sort an ArrayList if the references refer to objects that have a `compareTo` method.

  **Example:** SortingEx1.java

- If you want to sort objects of classes you defined:
  - Define a `compareTo` method in the class.
  - Add `implements Comparable<CLASSNAME>` to the class definition.
    - **Example:** public class Banana implements Comparable<Banana>
  - **Example:** SortingEx2.java
File I/O

• We can use the **Scanner** class along with the **File** class to open and read data from a text file

• **Example:** ReadingFriendsInfo.java

• We can read data from an online source using the **Scanner** class and the **URL** class

• **Example:** ReadingFromTextFile.java, ReadingOnlineData.java