

Homework 07 - inherit this

Prof Bill - Feb 2020

Due: **Mon Feb 24, 2020** (bring to class)

Reading:

- ❑ Sedgewick Java 3.3 Designing data types, introcs.cs.princeton.edu/java/33design
- ❑ Oracle Java, Interfaces and Inheritance section, docs.oracle.com/javase/tutorial/java
- ❑ Prof Bill blather, wtkrieger.faculty.noctrl.edu/csc210-spring2020/docs/java_inheritance_notes.pdf

thanks...yow, bill

1. We like two different class relationships: **is-a** and **has-a**. Which is **composition** and which is **inheritance**? How are each implemented in Java?

2. Explain how **protected** is in between public and private.

3. Explain how an **abstract class** is in between an interface and a class.

4. Create a **UML diagram** for the following situation.
 - interface Vehicle has methods numWheels() and maxSpeed()
 - interface CarPart has method partNumber()
 - class Wheel is-a CarPart
 - class Auto is-a Vehicle; has-a Wheel (4 of them, actually)

5. Can you **create objects** (with keyword new) for an interface or abstract class? If not, (hint hint), then what the heck are they used for?

6. Copy the **homework/hw07 code** from my k: drive and complete it. Search for comments that begin with “HW7” for directions.

Little help:

- Make Student class **Comparable**: google “java comparable example”
- Turn an ArrayList into **an array** (so we can do cool things): google “java arraylist to array”
 - This is tricky. Must create an array of Students first, and pass it into the toArray(T[] a) method in ArrayList.
 - Javadoc: docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html
- **Print** an array: google “java print array”
- **Sort** an array, implicitly uses : google “java sort array”
- Sort array using a **Comparator**: google “java comparator examples”
- Sort and **reverse order** to be descending: google “java sort array descending with comparator”

7. Can we use the tech in Question 6 for P2?

- Convert your Freq results to an array of WordCount210 objects.
- Then, liberally sort using Comparable or Comparator.