Week6Report

Charles Ceccardi
Andrew Martin
Samual Mitchell
Mitchell Oneka

3 Code Comprehension Problem (Estimated Time: 40 minutes):

Predict: What do you think the output of the program in Listings 1-4 will be?
There will be a compile error in the fourth listing.
Animal makes a sound, Dog barks, Cat Meows, and the fourth gives an illegal access error.

Run: Execute the program and observe the output. Did it match your expectations?
IllegalAccessError: cannot access class Dog from TestPolymorphism. Yes

Investigate: Describe how the `sound` method in the subclasses overrides the `sound` method in the parent class. Explain
By calling the subclass, it pulls the sound method from the subclass rather than the superclass

The use of the `Animal` reference to call the overridden method.

Modify: Add another animal, for example, a `Bird`, and make it produce its sound when the `sound` method is called.

Make: Describe the concepts of inheritance and polymorphism as demonstrated in the code snippet.
We have to make sure everything is given proper clarification when defining classes and methods.

4 Code-from-Scratch Problem (Estimated Time: 40 minutes)

Predict: Before writing the code, plan the classes and methods you will need.

A Book class that's the parent class to a RegularBook, EBook, and AudioBook class.

Run: Write and test the code. Did it work as expected?

yes.

Investigate: Explain the relationship between the parent class `Book` and its child classes. Describe the role of the `read` method in each class.

Modify: Add methods specific to each book type. For example, a method to adjust brightness for `EBook` or adjust speed for `AudioBook`.

Make: Use inheritance and polymorphism to create a different type of eBook, such as PDF, EPUB, or text.

5 Debugging and Extending Problem (Estimated Time: 40 minutes)

Predict: Identify the errors in the code
The honk method does not exist in the Vehicle Class.

Run: Execute the code and confirm the errors you predicted.
Compilation failed, cannot find symbol, symbol honk(). Couldn't find honk in Vehicle. No main method in vehicle or Car class.

Investigate: Describe the reason for the error in the code.
The Road Class makes an object for Vehicles instead of the Car class, and Vehicles doesn't have a honk() method. No main method in vehicle or Car class.

Modify: Fix the error in the code and test it.
Added public to everything then gave the vehicle a hook method.

Make: Extend the code to add another class `Bicycle` that inherits from `Vehicle`. Implement its own `start` method and add a method to ring a bell.

6 Reflection Questions

1. How does inheritance promote code reusability?
Inheritance allows child classes to reuse code from parent classes, meaning that the code does not need to be rewritten for each permutation that uses it.

2. What is the significance of the `super` keyword in the context of inheritance in Java?
The super keyword refers to the parent class that a child class inherits from.

3. How does polymorphism allow objects of different classes to be treated as objects of a common super class?

4. What is the difference between method overloading and method overriding in Java?
Method overloading allows for multiple methods with the same name that take different arguments, whereas overriding allows one method to replace another in a certain context.

5. Why is it not possible to call a subclass-specific method using a superclass reference, even if it points to an instance of
the subclass?

Because the super class reference refers to the encapsulating class instead of the parent class.