Lab 4

Implementing Inheritance

Objectives

Upon completion of this lab, you should be able to:

- Complete review questions
- Design superclasses and subclasses
- Implement superclasses and subclasses

Lab Overview

In this lab, you complete review questions and two exercise.

- In the first exercise, you design a class hierarchy from a scenario.
- In the second exercise, you create the class hierarchy and use it.

Completing Review Questions

Complete the following questions:

1. State whether the following statements are true or false:
   a. In the Java programming language, a superclass can inherit from its subclasses.
   b. An object is an abstract concept.
2. Which of the following Java programming language keywords indicates inheritance?
   a. The `implements` keyword
   b. The `inherits` keyword
   c. The `extends` keyword
   d. The `imports` keyword
3. The `import` keyword fulfills what task?
   a. It informs the virtual machine of the classes your program uses.
   b. It informs the virtual machine that you are using a graphic file in your program.
   c. It makes your program easier to read by allowing you to use shortened names when referring to classes in the Java API.
   d. It informs the compiler that you are compiling a program using classes from the Java API.
4. From which Java API package are the `String`, `Math`, and `Integer` classes implicitly imported?
   a. The `java.awt` package
   b. The `java.applet` package
   c. The `java.lang` package
   d. The `java.io` package

Exercise 1: Designing Superclasses and Subclasses

The objective of this exercise is to understand a scenario and design a class hierarchy.

Task 1 – Identifying Superclasses and Subclasses

In this task, you design subclasses and a superclass using the information in the following paragraphs.

Publishing, Inc. has employees in several different positions. These positions are: Technical Writers, Graphic Illustrators, Managers, and Editors.

Publishing, Inc. wants you to create a program for tracking information about each of its workers. This information consists of: the worker's name, job title, employee ID (in the range of 1 to 10,000), and level (1, 2, or 3). Additionally:

- Managers must have a list of employees that they manage.
- Technical Writers, Graphic Illustrators, and Editors must have a list of skills that they have.
- Editors must have a value indicating whether they prefer to do electronic editing or paper-based editing.
- All subclasses must have a method to display the information for the employee type.

To design your subclasses and superclasses:

1. Create a class hierarchy of superclass and subclass relationships for the employees in Publishing, Inc.

   Use the "is a" phrase to test your class hierarchy.
2. Model the class hierarchy using class diagrams similar to those in this module.

Demonstrate encapsulation for each of the classes in your design by including `get` and `set` methods in your design.

Notes

Exercise 2: Creating and Using Superclasses and Subclasses

The objective of this exercise is to create and implement the class hierarchy designed in the previous exercise.

Preparation

Ensure that the `EmployeeTest.java` file exists in the `SL110/exercises/11_hierarchy/exercise2/` directory. This is your working directory.

Task – Creating Classes

In this task, you develop subclasses and a superclasses for the employee class hierarchy. Follow these steps to develop your classes:

1. Go to the `inheritance` directory.
2. Create class files for each of the classes outlined in the first part of this exercise.
3. Create methods in each of the subclasses for displaying the information about the worker.
4. Create `get` and `set` methods, as detailed in your design, and use decision constructs to ensure that invalid data is not set.
5. (Optional) Try using methods from the `String` and `Character` class to ensure that the name and job titles that are set do not contain numerical characters.
6. Declare the `employeeID` as `static` to calculate unique Employee IDs.
7. Save and compile all the class files namely `Employee`,.`Manager`, `Editor`, `TechnicalWriter`, and `TechnicalIllustrator`.
8. Compile the `EmployeeTest.java` file, execute it, and test the output.

Exercise Summary

Take a few minutes to identify what experiences, issues, or discoveries you had during the lab exercises.

- Experiences
- Interpretations
- Conclusions
- Applications