Lab 2

Creating and Using Objects

Objectives

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

- Complete review questions
- Create and manipulate objects
- Use the `String` class
- Use the Java SE™ API

Lab Overview

In this lab, you complete review questions and three exercises.

- In the first exercise, you create and manipulate Java technology objects.
- In the second exercise, you create and use String objects.
- in the third exercise, you get familiar with the Java API specification.

Completing Review Questions

Complete the following questions

1. Which of the following lines of code instantiates a `Boat` object and assigns it to a `sailBoat` object reference?

   a. `Boat sailBoat = new Boat();`

   b. `Boat sailBoat;`

   c. `Boat = new Boat()`

   d. `Boat sailBoat = Boat();`

2. Read the following code and identify the result from the options given below:

```
Clothing myHat = new Hat();
Clothing mySock = new Sock();
myHat = mySock;
```

   a. The `myHat` variable now contains a `Sock` object.

   b. The `myHat` variable contains the same address as the `mySock` variable.

   c. The `Sock` and `Hat` variables are the same.

   d. The `myHat` variable remains unchanged.

3. Which of the following statements are true?

   a. The dot (.) operator creates a new object instance.

   b. The `String` class provides you with the ability to store a sequence of characters.

   c. The Java API specification contains documentation for all of the classes in a Java technology product.

   d. `String` types are unique because they are the only class that allows you to build objects without using the `new` keyword.

Exercise 1: Creating and Manipulating Java Technology Objects

In this exercise, you create instances of a class and manipulate these instances in several ways.

This exercise consists of the following tasks:

- "Task 1 – Initializing Object Instances"
- "Task 2 – Manipulating Object References"

Preparation

Ensure that `Customer.java` files exists in the `SL110/exercises/05_objects/exercise1` directory.

Task 1 – Initializing Object Instances

You have been provided with the `Customer` class. In this task, you create, compile, and execute a `CustomerTest` class. In the `CustomerTest` class, you create objects of the `Customer` class and set values to its member variables.

Complete the following steps:

1. Go to the `SL110/exercises/05_objects/exercise1` directory and open the `Customer.java` file.
2. Examine the `Customer` class, its member variables and its method.
3. Create new class `CustomerTest` and save it as `CustomerTest.java`.
4. Create the main method in the `CustomerTest` class.
5. In the main method, perform the following:
   a. Declare two instances of `Customer` class.
   b. Initialize the two instances of the `Customer` class using the `new` operator.

c. Assign values to the member variables of the `Customer` instance. Repeat the same for the second `Customer` object but with different values. For example:
`cObject1.customerID=1;`

d. Invoke the `displayCustomerInfo` method for each of the `Customer` objects

6. Save, compile and run the program.
7. Check the output to be sure that each `Customer` object displays the distinct values you assigned.

Task 2 – Manipulating Object References

In this task, you assign the value of one object reference to another object reference. Complete the following steps:

1. Open `CustomerTest.java` file, that you created and executed in the previous task.
2. In the `main` method, go to the line where you invoke the `displayCustomerInfo` method for both the `Customer` instances.
3. Assign one object reference to another object reference, before you call the `displayCustomerInfo` method for both the instances. For example: `cObject2=cObject1`. Assuming that `cObject1` and `cObject2` are the two instances of the `Customer` class.
4. Save, compile and run the program.
5. Check the output of the `displayCustomerInfo` methods for both the objects.

Exercise 2: Using the String Class

The objective of this exercise is to create and initialize String objects and to print their contents.

Preparation

Ensure that the PersonTwoTest.java file exists in the SL110/exercises/05_objects/exercise2 directory. This is your working directory.

Task – Creating and Using String Objects

In this task, you write a class called PersonTwo that creates and initializes two String variables and displays their values. Follow these steps to create your class:

1. Go to the working directory.
2. Create a class called PersonTwo that creates and initializes two variables:
   - The first variable stores a quotation.
   - The second variable stores the name of the person who said the quotation.
3. Use a method called displayQuote to display the quotation and the name of the person to the screen.
4. Compile your class.
5. Execute your class using the provided PersonTwoTest class.

Exercise 3: Using the Java SE™ API Specification

The objective of this exercise is to allow you to become familiar with the Java SE API specification.

Task – Examining the Java SE API Specification

Note – The intent of this exercise is to become familiar with the documentation and how you look up classes and methods. You are not expected to understand everything you look at. As you progress through this course, the Java API documentation should make more sense.

To view the Java API 6.0 specification:

1. Go to the following URL:
   `http://java.sun.com/javase/6/docs/`
2. From the site listed, find the API specification.
3. Using the API specification, complete the following:
   a. Find the `Math` class under the `java.lang` package. How many methods are there in this class?
   b. What class does every class refer to at the top of the page? Hint: What class is the superclass to all classes?
4. Find the `String` class. Identify the methods of `String` class, which will enable you to compare two strings.

Exercise Summary

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

- Experiences
- Interpretations
- Conclusions
- Applications