# Lab Exercise — Sales

**Name:** _____ **Date:** _____

**Section:** _____

The following problem is intended to be solved in a closed-lab session with a teaching assistant or instructor present. The problem is divided into six parts:

1. Lab Objectives
2. Problem Description
3. Sample Output
4. Program Template (Fig. L 5.2)
5. Problem-Solving Tips
6. Follow-Up Questions and Activities

The program template represents a complete working Java program with one or more key lines of code replaced with comments. Read the problem description and examine the output, then study the template code. Using the problem-solving tips as a guide, replace the `/* */` comments with Java code. Compile and execute the program. Compare your output with the sample output provided. Then answer the follow-up questions. The source code for the template is available at `www.pearsonhighered.com/deitel`.

## Lab Objectives

This lab was designed to reinforce programming concepts from Chapter 5 of *Java How To Program: 8/e*. In this lab, you will practice:

- Using `switch` statements.

The follow-up questions and activities also will give you practice:

- Validating the input from the user.
- Extending an existing program.

## Problem Description

A mail-order house sells five products whose retail prices are as follows: Product 1, $2.98; product 2, $4.50; product 3, $9.98; product 4, $4.49 and product 5, $6.87. Write an application that reads a series of pairs of numbers as follows:

a) product number
b) quantity sold

Your program should use a `switch` statement to determine the retail price for each product. It should calculate and display the total retail value of all products sold. Use a sentinel-controlled loop to determine when the program should stop looping and display the final results.

## Lab Exercise — Sales

### Sample Output

```
Enter product number (1-5) (0 to stop): 1
Enter quantity sold: 5
Enter product number (1-5) (0 to stop): 5
Enter quantity sold: 10
Enter product number (1-5) (0 to stop): 0

Product 1: $14.90
Product 2: $0.00
Product 3: $0.00
Product 4: $0.00
Product 5: $68.70
```

### Template

```
1  // Lab 2: Sales.java
2  // Program calculates sales, based on an input of product
3  // number and quantity sold
4  import java.util.Scanner;
5
6  public class Sales
7  {
8     // calculates sales for 5 products
9     public static void main( String args[] )
10    {
11       Scanner input = new Scanner( System.in );
12
13       double product1 = 0; // amount sold of first product
14       double product2 = 0; // amount sold of second product
15       double product3 = 0; // amount sold of third product
16       double product4 = 0; // amount sold of fourth product
17       double product5 = 0; // amount sold of fifth product
18
19       /* Ask the user to enter product number */
20
21       /* Create while statement that loops until sentinel is entered */
22
23          /* Determine whether user's product number is in 1-5 */
24
25             /* If so, ask user to input the quantity sold */
26
27             /* Write a switch statement here that will compute the total
28                for that product */
29
30          /* If product number is not in 1-5, test if product number is not 0 */
31             /* Display error message for invalid product number */
32
33          /* Ask the user to enter another product number */
34
35       /* end while loop */
36
37       // print summary
38       System.out.println();
39       System.out.printf( "Product 1: $%.2f\n", product1 );
```

**Fig. L 5.2** | `Sales.java`. (Part 1 of 2.)

## Lab Exercise — Sales

```
40         /* write code here for the rest of the summary message it should contain
41            the totals for the rest of the products, each on it's own line */
42      } // end method calculateSales
43   } // end class Sales
```

**Fig. L 5.2** | `Sales.java`. (Part 2 of 2.)

### Problem-Solving Tips

1. Before your `while` loop, request the first product number from the user.
2. Use a sentinel value to control the loop. This loop should terminate when the product number entered is zero.
3. If the user provides a valid product number, inside the loop, request a quantity for that product. Then, perform the appropriate calculation in the `switch` statement.
4. Your `switch` statement should consist of five cases, each setting the correct dollar value, depending on the quantity that the user entered.
5. Inside the closing right brace (}) of the loop's body, request the next product number from the user.
6. Be sure to follow the spacing and indentation conventions mentioned in the text. Before and after each control statement, place a line of vertical space to make the code more readable.
7. If you have any questions as you proceed, ask your lab instructor for assistance.

### Follow-Up Questions and Activities

1. What happens when the user inputs a number that is other than 1 through 5 or 0? Why? What is the output when the user enters a number like 7?
2. Modify the program so that there is a sixth product, priced at $20.00, that represents a package of all the products. For each of the packages purchased, add $4.00 to the totals for all the other products. Do not keep track of the packages separately.

**Important:**

1) The program must compile without errors.

2) Your program must have the following comments at the top. Don't forget to include them because they will count toward the grade of this lab.

```
/*************************************************************
 * First and Last Name   CSCI 6302.01 Fall 2019     Lab#: XX                 *
 *                                                                           *
  ************************************************************/
```

3) Rename your files as Sales_lastnameFirstname.java

4) When done, submit your *.java file via blackboard