CS 1063 Lab 2: Counting Coins and Making Change

The `Coins` Class

Objectives

- Use methods.
- Use local variables.
- Use arithmetic expressions.
- Use Scanner to input values.
- Use a class constant.

Hand-in Requirements

All projects and laboratories will be submitted electronically through Blackboard. Zip up your entire lab directory to submit as the source. (Right click on the lab folder and follow **Send To > Compressed (zipped) Folder** or **7-Zip > Add to "lab2.zip"**). The lab folder should include the following:

- `Coins.java`
- `CoinsOutput.txt`

Tasks

Write a program that prints

```
Lab 2 written by YOURNAME
```

and calls two methods:

1. Input the number of quarters, dimes, nickels, and pennies from the user. Print out the number of coins and total value in dollars.
2. Input the number of cents from the user. Determine and print out the number of quarters, dimes, nickels, and pennies to add up to that number of cents. No, you can't use all pennies.

Details

Getting Input from the User

You can get input from the user using a Scanner object. Here is an example program with key lines in bold.

```
import java.util.*;

public class HalfDollar {
  public static final Scanner CONSOLE = new Scanner(System.in);

  public static void main(String[] args) {
    halfDollarAmount( );
  }

  public static void halfDollarAmount( ) {
    System.out.print("Enter the number of half dollars: ");
    int halfDollar = CONSOLE.nextInt( );
    double amount = halfDollar * 0.50;
    System.out.println(halfDollar + " half dollars is $" + amount);
  }
}
```

The import statement tells Java that we want to use Java's `java.util` package (Scanner is part of this package).

The statement:

```
public static final Scanner CONSOLE = new Scanner(System.in);
```

declares a special kind of variable (a *class constant*) named `CONSOLE` of type `Scanner` and stores an "object" for input from the keyboard in `CONSOLE`. `CONSOLE` can be used anywhere in the class.

The first two statements of the `halfDollarAmount` method:

1. prompt the user for information, and
2. input a number from the keyboard and stores the number in a variable named `halfDollar`.

You only need the declaration/assignment for `CONSOLE` once in the class. On the other hand, you need prompt and input statements for each value you want the user to enter.

Local Variables

You should have local variables for each value that is entered and each value that you calculate. For example, for the first method there should be a local variable that stores the total dollar amount of the change, as well as a local variable for the total number of coins.

Making Change

For the second method, we can use integer division and the mod operator to make change. Suppose the user enters 99 as the number of cents. The sequence of calculations should be as follows.

1. The integer division `99 / 25` can be used to determine the number of quarters (3).
2. Using the mod operator `99 % 25` determines the remaining number of cents to be converted into change (24).
3. Integer division `24 / 10` can be used again to determine the number of dimes (2).
4. The mod operator `24 % 10` can be used again to determine the remaining amount (4).
5. Next, for nickels, integer division `4 / 5` for the number of nickels (0) and a mod operation `4 % 5` for the remaining amount (4).
6. Whatever is left is the number of pennies (4).

Of course the values that were calculated (3, 24, 2, 4, 0, 4) would be different if the user enters something different from 99. This means we should have variables to store all of these values. For example, we might have a variable named `cents` to hold the value that the user entered, a variable named `remainingAfterQuarters` for the remaining amount after the quarters are determined, which could be assigned by:

```
int remainingAfterQuarters = cents % 25;
```

Output File

Include the output of your program in a file named `CoinsOutput.txt`.

Rubric

Your program should compile without any errors. A program with more than one or two compile errors will likely get a zero for the whole assignment.

The following criteria will also be used to determine the grade for this assignment:

- [1 Point] If the main method of your program prints "Lab 2 written by [...]".
- [1 Point] If your submission was a Zip file named `lab2.zip` containing a folder named `lab2`, which contains the other files.
- [1 Point] If your Java program was in a file named `Coins.java`.
- [1 Point] If the output of your Java program was in a file named `CoinsOutput.txt`.
- [1 Point] If your program contains a comment that describes what the program does and contains a comment for each method.
- [1 Point] If your program is indented properly.
- [7 points] If your main method calls a method that does the following:
  - [3 Points] If it prompts the user for the number of quarters, dimes, nickels, and pennies.
  - [4 Points] If it prints out the total number of coins and their value in dollars.

- [7 points] If your main method calls another method that does the following:
  - [3 Points] If it prompts the user for the number of cents.
  - [4 Points] If it prints out the number of quarters, dimes, nickels, and pennies necessary to add up to that number.

Revision Date Tue Sep 13 2011 11:03:26 GMT-0500 (Central Daylight Time)