Lab Session #8

Part 1: Theory

1. Suppose you are trying to use a command from the class Dawson with the following *signature*:  
     
   fun(double, int) -> double  
     
   Suppose further that you have the following variables declared and assigned values.  
     
   double x;  
   int y;  
     
   Which of the following are valid ways to use the command? If it is not valid, what could you do to correct it?
   1. Dawson.fun(10.0, 3 + 2);
   2. Dawson.fun(x, y);
   3. Dawson.fun(double x, int y);
   4. Dawson.fun();
   5. Dawson.fun(“x”, “y”);
   6. double r = Dawson.fun(2,3);
   7. int q = Dawson.fun(3, 4);
   8. int r = (int) Dawson.fun(2,10) + 10;
   9. double z = Dawson.fun(3, 4.0 + 2);
   10. double a = Dawson.fun( Dawson.fun(10, 3), 3);
   11. double b = Dawson.fun( Dawson.fun(10,3), Dawson.fun(10,2));
   12. double c = Dawson.fun( Dawson.fun(10, 3), (int) Dawson.fun(10,2));

**Part 2: Application**1. Last lab, you wrote a program for calculating the change given a specific coin. If you did not finish that program, then you should start by completing it! 😊 Modify your application to use % operation for figuring out the remainders at each step. (Bonus: Extend your program to include 5, 10, and 20 dollar bills)

2. Write a Java application to accept two doubles from the user, x and y, and then calculates and prints

3. Write a Java application to figure out how much money will be in your bank account after interest as described below:  
  
**Background**: When your money is in a savings account at a bank, it earns *interest* every year. This means, that the amount grows. For example, if you earn 2% interest on $200, you will end up with $204. An important concept related to interest is that the interest is *compounded*. This means that after you earn that first 2%, you now have $204. So the next time interest is added, it will be on $204, and not $200. (Note: If you have a credit card with interest, then the interest works *against* you since it means you owe more.) The amount that you have at the end is given by the following formula.

where:

P = amount of money you have to start (“the principal”)

r = the amount of interest earned per year (as a decimal point. So 5% for example, would be .05)

and t = the number of times the interest is compounded.

Write a Java program that asks the user to enter the initial amount of money (P), the rate

per year (as a percentage), and the number of times the interest is compounded. The

program should then, using the Math.pow() method mentioned above, calculate (and

display) the total amount of money in the account.

4. In Lab 6, you wrote a program to calculate the average of three integers. Since the three inputs were integers, it resulted in an integer division when divided by 3. Modify the program to cast the numbers to double, so the result produced is the correct double value.