Week 3 Practice Problems

#1. Write a function that takes in a three-digit integer and returns the "flipped" version (that is, with its digits reversed). For example:

> Enter a positive number less than 1000: **177**
> That number flipped is 771

Hint: Represent the user's input as a integer and use the arithmetic operators // and % to do it.

Make sure it works if the user enters a one or two digit integers. For example, 37 flipped is 730, and 8 flipped is 800.

#2. Write a function that asks a user for a positive integer, and then prints the minimum number of quarters, dimes, nickels, and pennies needed to make up that amount. For example:

> Enter an amount: **67**
>
> 2 quarter(s), 1 dime(s), 1 nickel(s), 2 penny(ies)

#3. Write a program that asks a user for a floating point number and that then prints two values: the number truncated to the first decimal place and the number rounded to the first decimal place. For example:

> Enter a number: **4.158**
> Truncated to one decimal place: 4.1
> Rounded to one decimal place: 4.2

Hint: Have a look at the math module.

#4. Write a little trigonometry progam. Ask a user for an angle, specified in degrees. Then print the sine, cosine, and tangent of that angle. For example:

Enter an angle: **60.**
sin(60.00) is 0.866025
cos(60.00) is 0.500000
tan(60.00) is 1.732051

#5. Write a program that takes the role of a store clerk. Ask a user to enter two floating point numbers: the cost of an item and the amount of money remitted to pay for the item. Then respond appropriately: calculate the change due to the customer or ask the customer for more money. For example:

Cost of the item: 3.56
Amount tendered: 5.00
Change: 1.44

or

Cost of the item: 3.56
Amount tendered: 3.00
Still due: 0.56

#6. Combine #5 with #2: if in #5 you owe the customer some change, calculate the minimum number of $100, $50, $20, $10, $5, toonies, loonies, quarters, dimes, nickels, and pennies needed to make up that amount. Modify the function from #2 to print out the way to make the change.

#7. Your code for #6 doesn't work in Canada since we no longer have pennies. Fix it to work in Canada. (Hint: How do stores make change when the amount isn't a multiple of 5?)

#8. Write a function that takes a positive integer input less than 100000 and returns an integer corresponding to the number of digits in the number. Use this function in a program that prompts the user for an integer and prints the number of digits in the user's input. [Note: there are a number of ways to solve this problem. It is pretty easy if you do it with a string. You should also try it directly with an integer.]

#9. Water exists in three states- solid, liquid, and gas. Write a function that takes in the temperature in Celsius and returns a string "solid", "liquid", or "gas" depending on the temperature. Write a program using this function that prompts the user for a temperature and prints out the resulting state of water.

#10. In fluid mechanics the dimensionless Reynolds Number is defined below, where v is the speed of the fluid (m/s), L is the characteristic length of the flow situation (m), and n is the kinematic viscosity (m^2/s).

Reynolds Number = vL/n

The magnitude of the Reynolds Number tells if a flowing fluid is moving in a laminar, transitional, or turbulent mode. Two flow situations are of interest, flow in a pipe and flow on a flat plate.

For a pipe:	When the Reynolds number is less than 2000, then the flow is laminar. When the Reynolds number is greater than or equal to 2000 and less than or equal to 4000, then the flow is transitional. When the Reynolds number is greater than 4000, then the flow is turbulent.
For a plate:	When the Reynolds number is less than 5×10^5, then the flow is laminar. When the Reynolds number is greater than or equal to 5×10^5, then the flow is turbulent.

Write a function that takes the speed of the fluid, the characteristic length, the kinematic viscosity, and the flow situation as arguments and returns a string indicating if the flow is "`laminar`", "`transitional`", or "`turbulent`".

You may assume that:

- the first three arguments are positive numbers in the correct units.
- the fourth argument is a string equal to "`pipe`" or "`plate`"