CSCA20 Lab 2

To earn your lab marks, you must actively participate in the lab and achieve moderate success. You will find your worksheets/slides from the last lecture helpful.

1 Objectives

- Explore module `math` using `help` and `dir`.
- Write functions.
- Calling functions within the same file (creating a program).

2 Driver and navigator

As always, if you wish to complete this lab with a partner, then you and your partner will take on distinct roles.

driver: The person typing at the keyboard.

navigator: The person watching for mistakes, and thinking ahead.

The rest of these instructions call you `s1` and `s2`. Pick which one is which. `s1` should log in, start up Wing, and be the first driver.

3 Explore `math` module using `dir` and `help`

Complete the tasks below using functions from Python's `math` module.

Recall a *module* is simply a file that contains functions in it. We can use the functions in the file by using `import filename`. For example, to use a function like `sqrt(x)` from the `math` module you simply write `math.sqrt(`some parameter`)`. Let's try it!

1. In the shell, type `import math`.

2. Type `dir(math)` and `help(math)`.

3. Type `help(math.function_name)` to get help with module `math` and a specific function. Now write code to do the following by finding appropriate functions from the module math:

   (a) Calculate 2π.

   (b) Take the floor and ceiling of 56.4.

   (c) Calculate the square root of the logarithm of 244.

   (d) Calculate the base 2 logarithm of 256.

4 Writing Programs

Switch driver and navigator.

In this section you will write a little loan calculator program using the following formulae:
Simple Interest Accrued $= A * i * t$

where A is the principal or initial amount of the loan, i is the nominal interest rate per year and t is the number of years.

Compound Interest Accrued $= A * ((1 + i/n)^{nt} - 1)$

where A is the principal or initial amount of the loan, i is the *nominal* interest rate per year, n is the number of times interest is compounded per year and t is the number of years.

Create a file called `loans.py` and write the following functions. If you have seen `docstrings` in class, then include a docstring comment for each. The docstring should be an appropriate rephrasing of the specification on the right-hand side of this table.

We give the parameter types for each function; make sure you pick good names for them.

Function Name	**Description**
`simple_interest(float, float, float)`	The first parameter is the principal or size of a loan. The second parameter is the interest rate per time period. The third parameter is the length in time periods of the loan. Return a `float` that is the simple interest accrued on the loan. For example, a \$2000 loan with an interest rate of 5% per year, will accrue over 3 years: \$2000*0.05*3 = \$300
`compound_interest(float, float, int, float)`	The first parameter is the principal or value of the loan, the second is the interest rate per time period, the third is the number of times the interest is compounded per time period, and the fourth is the number of time periods. Return a `float` that is the compounded interest accrued on the loan.

Switch driver and navigator.

Now let's create a *program*. Below your functions in `loans.py` write the statement

```
if __name__ == '__main__':
```

This turns your file into a program you can run. We will consider the following values:

$$A = 1000; i = 0.06; t = 3$$

The initial value of the loan is \$1000, the annual interest rate is 6% and the loan term is 3 years. For example, if I wish to find out the final value of the loan, I would type:

```
if __name__ == '__main__':
    interest_charges = simple_interest(1000, 0.06, 3)
    total_cost = 1000 + interest_charges
    print("The total cost of a $1000 loan with interest rate 6% over 3 years is", total_cost)
```

Notice that the `print` function displays the argument in the shell so that we can **see** the result of our calculations.

Your turn. Answer the questions below by *calling your functions* and using the `print` statement. Your program should display one sentence per question below (so two sentences total).

Compound Interest

Let $A = 1000; i = 0.06; n = 12; t = 3$. How much compound interest is accrued?

Let $A = 1000; i = 0.06; n = 12; t = 3$. What is the total cost of the loan?

Submit your `loans.py` file to MarkUs.