YSC2221-1 Introduction to Python
Lab Exercise 01

- Add you code into the skeleton file and submit it. You should NOT change the function names.
- For our assignments, you are NOT allowed to use other libraries. Please do not import other libraries except those are already imported in the skeleton code. And for this current assignment, you should NOT use sequences in Python either.
- For the input/arguments of your functions, you can assume that they are always numbers. For Questions 1-3, you can even assume that the inputs are integers.
- For Questions 1 and 3, you should "return" a value rather than "print". **Basically you should NOT use "print()" in Q1/3.**

Question 1 Simple Selection

Write 2 functions, **is_even** and **is_odd**, which takes in an integer, n, and returns True or False accordingly. You must implement the functions using if-else.

Sample output:
```
>>> is_even(0)          >>> is_odd(1)
True                    True
>>> is_even(1)          >>> is_odd(4)
False                   False
```

For this question, it is ok if you implemented the functions correctly. However, you may think of the following variations of implementing the functions because there could be a lot of ways to implement a function

- One-line implementation: Implement the function body (excluding the def line) with only one single line of code
- Using other function: Maybe you can implement is_odd() by calling is_even()?

Question 2 Loops (Repetition)

a) Write a function, **print_n**, that takes in an integer n, and print out the first n positive integers.

Sample output:
```
>>> print_n(3)
1
2
3
>>> print_n(6)
1
2
3
4
5
6
```

Question 2 (Cont.)

b) Write another function, **print_squared**, that takes in an integer, n, and print out the squares of the first n positive integers

```
Sample output:
>>> print_squared(2)
1
4
>>> print_squared(4)
1
4
9
16
```

Question 3

a) Write a function **is_divisible(n,d)** to return true if n is divisible by d.

```
>>> is_divisible(100,25)
True
>>> is_divisible(100,3)
False
>>> is_divisible(123,3)
True
```

b) A positive integer n is *prime* if it is divisible by itself and 1 only. Write a function, **is_prime**, which takes in an integer n > 1, and returns True if the number is a prime number and False otherwise. (Hint: This question is highly related to Q3a)

```
Sample output:
>>> is_prime(2)
True
>>> is_prime(4)
False
>>> is_prime(7)
True
>>> is_prime(331)
True
```

Question 4

Remember how to solve a quadratic equation in the lecture? Use the if-else statements to handle ALL possibilities of the coefficients of a, b and c. Here are some, **but not all**, possibilities of the program output. Basically, your program should not "crash" (error message by Python) if the three arguments of a, b and c are all numbers. For packages, you can only use the function sqrt() from math but NOT any other packages/imported functions.

```
>>> solve_qe(1,5,6)
The two solutions are -2.0 and -3.0
>>> solve_qe(1,4,4)
The solution is -2.0
>>> solve_qe(3,1,4)
The solution is not real
```