Part 4

August 2017

1. Write a program that initializes a list L with integer entries (make it as long as you like and the entries can be arbitrary). Ask the user for input. If the user inputs something that is an element of the list, remove it from the list and print out the edited list. If the user inputs something that is not an element of the list, print out the list without editing it.

```python
L = [1, 2, 22, 15]
n = int(input("Enter an integer: "))
if n in L:
    L.remove(n)
print(L)
```

2. Given the list ['John Smith', 'Michael Anderson', 'Archibald Farnsworth the Fourth'] add another name in between the second and third entries.

```python
li = ['John Smith', 'Michael Anderson', 'Archibald Farnsworth the Fourth']
li.insert(2, 'Some Name')
print(li)

## ['John Smith', 'Michael Anderson', 'Some Name', 'Archibald Farnsworth the Fourth']
```

3. Given the list ['s', 'h', 'j', 'k', 'd', 'q'], rearrange it in reverse alphabetical letter.

```python
li = ['s', 'h', 'j', 'k', 'd', 'q']
li.sort()
li.reverse()
```

4. Given the list ['s', 'h', 'j', 'k', 'd', 'q'], write a program that will return every other letter in the list starting with the second. Do not use the length of the list in your code.

```python
li = ['s', 'h', 'j', 'k', 'd', 'q']
new_li = li[1::2]
```

5. Use `range` to create the list [2,4,6,8,10,12,14,16,18,20,22].

```python
li = list(range(2,24,2))
```

6. What is the result of `list(range(13,40,5))`.

```python
print( list(range(13,40,5)) )

## [13, 18, 23, 28, 33, 38]
```

7. Given the list ['John Smith', 'Michael Anderson', 'Archibald Farnsworth the Fourth'], use one line of code replaces the middle entry with some other name.

```python
li = ['John Smith', 'Michael Anderson', 'Archibald Farnsworth the Fourth']
li[1] = 'Some Name'
```

8. Write a program that prints out the squares of all the positive integers from 1 to 10.

```python
for i in range(1, 11):
    print(i**2)

## 1
## 4
## 9
```

```
## 16
## 25
## 36
## 49
## 64
## 81
## 100
```

9. a. Write a program that sums the first 100 positive integers and prints the sum.

```python
sumval=0
for i in range(1,101):
    sumval = sumval + i
print(sumval)
```

```
## 5050
```

b. Now write a program that sums the first n integers and prints the sum. n should be read from user input.

```python
sumval=0
n = int(input("Enter a positive integer: "))
for i in range(1,n+1):
    sumval = sumval + i
print(sumval)
```

10. Ask the user to input a positive integer n and print out its factorial n! = 1 x 2 x 3 x ... x n.

```python
n = int(input("Enter a positive integer: "))
factorial = 1
for i in range(1, n + 1):
    factorial *= i
print(factorial)
```

11. What is wrong with the following code?

```python
for i in range(0, 10, 0.1):
    print(i)
```

12. Ask the user for a positive integer. Print out all the integer divisors of this number, including 1 and itself.

```python
n = int(input("Enter a positive integer: "))
for i in range(1, n+1):
    if n%i == 0:
        print(i)
```

13. a. Write a program that accepts input entered by the user and stores each entry in a list. The program should stop when the user enters 'end'.

```python
entry = ''
        L = []
        while entry != 'end':
            entry = input("Type something: ")
            L.append(entry)
```

b. Add code to the program that will print out all the data entered by the user (print each element on a separate line).

```python
for entry in L:
        print(entry)
```

    c. Add code to the program that will print out only those entries that are numbers (hint: you will need the `isalpha()` method; type `help(str)` in the Python shell)

```python
for entry in L:
        if entry.isalpha():
            print(entry)
```

14. Write a program that prints out the first 100 prime numbers. A number is prime if it is a positive integer such that its only integer divisors are 1 and itself. By convention, 1 is not considered a prime number.

```python
for i in range(2, 101):
    divisors = []
    for j in range(1, i+1):
        if i%j == 0:
            divisors.append(j)
    if len(divisors) == 2:
        print(i)
```

```
## 2
## 3
## 5
## 7
## 11
## 13
## 17
## 19
## 23
## 29
## 31
## 37
## 41
## 43
## 47
## 53
## 59
## 61
## 67
## 71
## 73
## 79
## 83
## 89
## 97
```

15. Write a program that accepts two integers as user input and prints their greatest common divisor.

```python
n1 = int(input("Enter an integer: "))
n2 = int(input("Enter another integer: "))
divisors1 = []
divisors2 = []
for i in range(1, n1+1):
    if n1%i == 0:
```

```python
        divisors1.append(i)
for i in range(1, n2+1):
    if n2%i == 0:
        divisors2.append(i)
gcd = 1
for divisor in divisors1:
    if divisor >= gcd and divisor in divisors2:
        gcd = divisor
print("The greatest common divisor of", n1, " and", n2, "is", gcd)
```

16. You have already written a program that calculates all prime numbers less than 100. Write a function primes that takes one integer parameter and returns the number of primes less than that integer.

```python
def primes(n):
    number_of_primes = 0
    for i in range(2, n1):
        divisors = []
        for j in range(1, i+1):
            if i%j == 0:
                divisors.append(j)
        if len(divisors) == 2:
            number_of_primes += 1

    return number_of_primes

n1 = int(input("Enter an integer: "))
print( primes(n1) )
```