Lecture 6.1. While loops

Q1. Factorials (from lecture 3.1)

In mathematics, the factorial of a non-negative integer ***n***, denoted by ***n!,*** is the product of all positive integers less than or equal to n.

n! = n * (n-1) * (n-2) * (n-3) * …. * 3 * 2 * 1

The value of 0! is 1.
The following code implements factorials for n from 0 to 5:

```
 2 factorial = 1
 3
 4 if number > 0:
 5     factorial = factorial * number
 6     number = number - 1
 7
 8 if number > 0:
 9     factorial = factorial * number
10     number = number - 1
11
12 if number > 0:
13     factorial = factorial * number
14     number = number - 1
15
16 if number > 0:
17     factorial = factorial * number
18     number = number - 1
19
20 if number > 0:
21     factorial = factorial * number
22     number = number - 1
23
24 print(factorial)
```

Line 1	number = 5	
Line #	number	factorial
	factorial =	

Line 1	number = 2	
Line #	number	factorial
	factorial =	

Q2.

```
3  while i < 5:
4
5      print(i)
6
7      i = i + 1
8
9  print("Outside loop")
```

Line 1	i = 0	
Line #	i	Print statement

Line 1	i = 5	
Line #	i	Print statement

Q3a. The code block on the right computes factorials for *any* non-negative number n (not just from 0 to 5).

Also note how we now just need 8 lines of code with a while loop, compared to 24 before.

```
2  factorial = 1
3
4  while number > 0:
5      factorial = factorial * number
6      number = number - 1
7
8  print(factorial)
```

Line 1	number = 5	
Line #	number	factorial
	factorial =	

Line 1	number = 2	
Line #	number	factorial
	factorial =	

Q4. (Lecture 3.3, Q1)

Prime numbers are natural numbers greater than 1 that are only divisible by 1 and themselves.

For a given positive integer `num`, the code block on the right checks if it is prime or not.

```
2  i    = 2
3
4  is_prime = True
5
6  while i < num:
7
8      if num % i == 0:
9          is_prime = False
10         break
11
12     i = i + 1
13
14 print(is_prime)
```

Line 1	num = 4		
Line #	num	i	is_prime

Line 1	num = 5		
Line #	num	i	is_prime