Lecture 6.3. While loops (continued)

Q1. Continue statements

The following code block on the right iterates over numbers from 0 to n and prints out odd numbers.

Line	n	i	print statements

```
1  n = 10
2
3  i = 0
4
5  while i < n:
6
7      i = i + 1
8
9      if i % 2 == 0:
10         continue
11
12     print(i)
```

Q2. Break statements

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

Q3. Nested Loops

prime_count	nth_prime	i	j	is_prime

```python
def get_nth_prime(n):

    prime_count = 1
    nth_prime = 2

    i = 2

    while prime_count <= n:

        j = 2
        is_prime = True

        while j < i:

            if i % j == 0:
                is_prime = False
                break

            j = j + 1

        if is_prime:
            prime_count = prime_count + 1
            nth_prime   = i

        i = i + 1

    return nth_prime
```

prime_count	nth_prime	i	j	is_prime