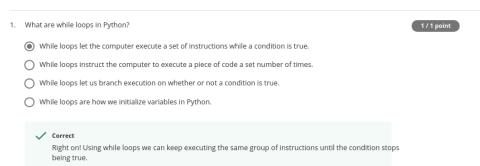
GRADE 100%

Practice Quiz: While Loops

TOTAL POINTS 5



Fill in the blanks to make the print_prime_factors function print all the prime factors of a number. A prime factor is a number that is prime and divides another without a remainder. 1 / 1 point

```
def print_prime_factors(number):
        # Start with two, which is the first prime
        # Keep going until the factor is larger than the number
        while factor <= number:

# Check if factor is a divisor of number
          if number % factor == 0:
    # If it is, print it and divide the original number
    print(factor)
             number = number / factor
          else:
# If it's not, increment the factor by one
11
12
             factor += 1
13
14
        return "Done"
15
     print_prime_factors(100)
16
      # Should print 2,2,5,5
      # DO NOT DELETE THIS COMMENT
18
```

✓ Correct

You nailed it! You've got the code to print all the right prime factors. Well done!

3. The following code can lead to an infinite loop. Fix the code so that it can finish successfully for all numbers.

1 / 1 point

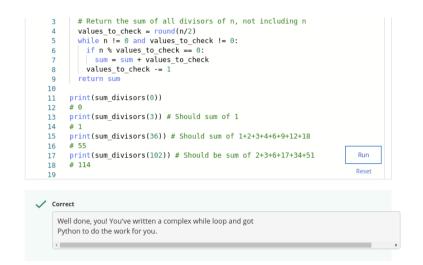
Note: Try running your function with the number 0 as the input, and see what you get!

```
Awesome! You fixed a tricky error that was hard to find and the function now behaves correctly.
```

Fill in the empty function so that it returns the sum of all the divisors of a number, without including it. A divisor
is a number that divides into another without a remainder.

1 / 1 point

```
1 def sum_divisors(n):
2 | sum = 0
```



5. The multiplication_table function prints the results of a number passed to it multiplied by 1 through 5. An additional requirement is that the result is not to exceed 25, which is done with the break statement. Fill in the blanks to complete the function to satisfy these conditions.

1 / 1 point

```
def multiplication_table(number):
             # Initialize the starting point of the multiplication table
             # Intralate the starting point of this mattaplian in multiplier = 1
# Only want to loop through 5
while multiplier <= 5:
    result = number * multiplier
    # What is the additional condition to exit out of the loop?
                   if result > 25 :
 8
                   print(str(number) + "x" + str(multiplier) + "=" + str(result))
# Increment the variable for the loop
multiplier += 1
10
11
12
13
       multiplication_table(3)
# Should print: 3x1=3 3x2=6 3x3=9 3x4=12 3x5=15
14
15
16
17
       multiplication_table(5)
       # Should print: 5x1=5 5x2=10 5x3=15 5x4=20 5x5=25
18
                                                                                                             Run
19
20
       multiplication_table(8)
       # Should print: 8x1=8 8x2=16 8x3=24
21
```

Correct

Excellent! You completed the multiplication table with all of the required criteria, and it looks great!