[Jahangirnagar University](https://www.coursera.org/programs/jahangirnagar-university-na-qyc9d)

[Skip to Main Content](https://www.coursera.org/learn/python-crash-course/quiz/C9MVo/practice-quiz-for-loops/attempt?redirectToCover=true#main)

* 

MD. TAZEL HOSSAN

[Crash Course on Python](https://www.coursera.org/learn/python-crash-course/home/welcome)

[Week 3](https://www.coursera.org/learn/python-crash-course/home/week/3)

Practice Quiz: For Loops

[Prev](https://www.coursera.org/learn/python-crash-course/supplement/zCkm0/loops-cheat-sheet)

[Next](https://www.coursera.org/learn/python-crash-course/lecture/7T0ET/what-is-recursion-optional)

**PRACTICE QUIZ • 20 MIN**

**Practice Quiz: For Loops**

**Submit your assignment**

Try again

**Receive grade**

**TO PASS**80% or higher

**Grade**

100%

View Feedback

We keep your highest score

Practice Quiz: For Loops

Practice Quiz • 20 min

**Congratulations! You passed!**

**TO PASS**80% or higher

Keep Learning

**GRADE**

100%

**Practice Quiz: For Loops**

**TOTAL POINTS 5**

1.Question 1

How are while loops and for loops different in Python?

**1 / 1 point**



While loops can be used with all data types, for loops can only be used with numbers.



For loops can be nested, but while loops can't.



While loops iterate while a condition is true, for loops iterate through a sequence of elements.



While loops can be interrupted using break, for loops using continue.

**Correct**

You got it! We can use while loops when we want our code to execute repeatedly while a condition is true, and for loops when we want to execute a block of code for each element of a sequence.

2.Question 2

Fill in the blanks to make the factorial function return the factorial of n. Then, print the first 10 factorials (from 0 to 9) with the corresponding number. Remember that the factorial of a number is defined as the product of an integer and all integers before it. For example, the factorial of five (5!) is equal to 1\*2\*3\*4\*5=120. Also recall that the factorial of zero (0!) is equal to 1.

**1 / 1 point**

1

2

3

4

5

6

7

8

def factorial(n):

    result = 1

    for x in range(1,n):

        result = x \* result

    return result

for n in range(0,10):

    print(n, factorial(n+1))





RunReset

**Correct**

Great work! The pieces of code you're tackling keep getting

more complex, you're doing a great job!

3.Question 3

Write a script that prints the first 10 cube numbers (x\*\*3), starting with x=1 and ending with x=10.

**1 / 1 point**

1

2

for x in range(1,11):

  print(x\*\*3)





RunReset

**Correct**

You nailed it! You got the code to print the first 10 cubes.

4.Question 4

Write a script that prints the multiples of 7 between 0 and 100. Print one multiple per line and avoid printing any numbers that aren't multiples of 7. Remember that 0 is also a multiple of 7.

**1 / 1 point**

1

2

for i in range(0,100,7):

        print(i)





RunReset

**Correct**

Awesome! You're getting Python to do all the work for you.

5.Question 5

The retry function tries to execute an operation that might fail, it retries the operation for a number of attempts. Currently the code will keep executing the function even if it succeeds. Fill in the blank so the code stops trying after the operation succeeded.

**1 / 1 point**

1

2

3

4

5

6

7

8

9

10

def retry(operation, attempts):

  for n in range(attempts):

    if operation():

      print("Attempt " + str(n) + " succeeded")

      break# will stop the loop when or if this is line reached

    else:

      print("Attempt " + str(n) + " failed")

retry(create\_user, 3)

retry(stop\_service, 5)





RunReset

**Correct**

Well done, you! You've fixed the code to stop executing once

the function is successful.