

✓ Congratulations! You passed!

TO PASS 80% or higher

Keep Learning

GRADE 100%

Practice Quiz: Conditionals

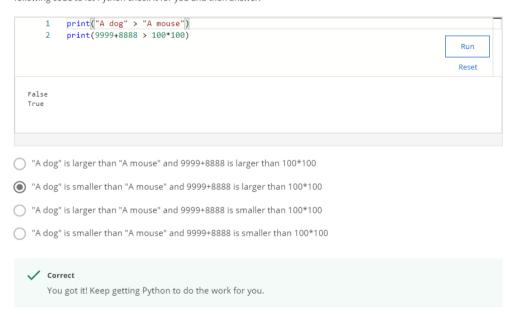
TOTAL POINTS 5

What's the value of this Python expression: $(2**2) == 4?$		1/1 point
O 4		
2**2		
True		
○ False		
Correct You nailed it! The conditional operator == checks if two values are equal. The result of that boolean: either True or False.	t operation is a	
Complete the script by filling in the missing parts. The function receives a name, then returns a grewhether or not that name is "Taylor".	eeting based on	1/1 point
<pre>1 def greeting(name): 2 if name == "Taylor":</pre>		
3 return "Welcome back Taylor!"		
4 else: 5 return "Hello there, " + name		
6	Run	
<pre>7 print(greeting("Taylor")) 8 print(greeting("John"))</pre>	Reset	
Welcome back Taylor! Hello there, John		
✓ Correct		
Great work! You're getting the hang of conditionals in Python.		
What's the output of this code if number equals 10?		1/1 point
1 if number > 11:		_
<pre>2 print(0) 3 elif number != 10:</pre>		
5 elif number := 10: 4 print(1)		
5 -115		

elif number >= 20 or number < 12: 6 print(2) else:

print(3)

Right on! Our number is 10, which is smaller than 12, so it matches that condition.



5. If a filesystem has a block size of 4096 bytes, this means that a file comprised of only one byte will still use 4096 bytes of storage. A file made up of 4097 bytes will use 4096*2=8192 bytes of storage. Knowing this, can you fill in the gaps in the calculate_storage function below, which calculates the total number of bytes needed to store a file of a given size?

1 / 1 point

```
def calculate_storage(filesize):
            block_size = 4096
   2
   3
            # Use floor division to calculate how many blocks are fully occupied
   4
            full_blocks = filesize//block_size
   5
            # Use the modulo operator to check whether there's any remainder
   6
            partial_block_remainder = filesize%4096
            # Depending on whether there's a remainder or not, return
   8
            \ensuremath{\text{\#}} the total number of bytes required to allocate enough blocks
   9
            # to store your data.
   10
            if partial_block_remainder > 0:
  11
               return 4096*(full blocks+1)
            return full_blocks*4096
  12
  13
  14 print(calculate_storage(1))  # Should be 4096
       print(calculate_storage(4096)) # Should be 4096
  15
                                                                                             Run
  16
       print(calculate_storage(4097)) # Should be 8192
                                                                                             Reset
       print(calculate_storage(6000)) # Should be 8192
  17
4096
4096
8192
8192
```

