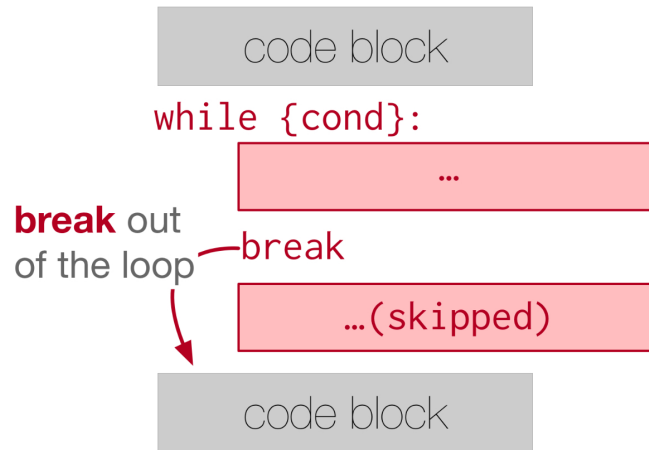




## 14.5. Break and Continue

Python provides ways for us to control the flow of iteration with a two keywords: `break` and `continue`.

`break` allows the program to immediately 'break out' of the loop, regardless of the loop's conditional structure. This means that the program will then skip the rest of the iteration, without rechecking the condition, and just goes on to the next outdented code that exists after the whole while loop.



Save & Run Original - 1 of 1 Show in CodeLens

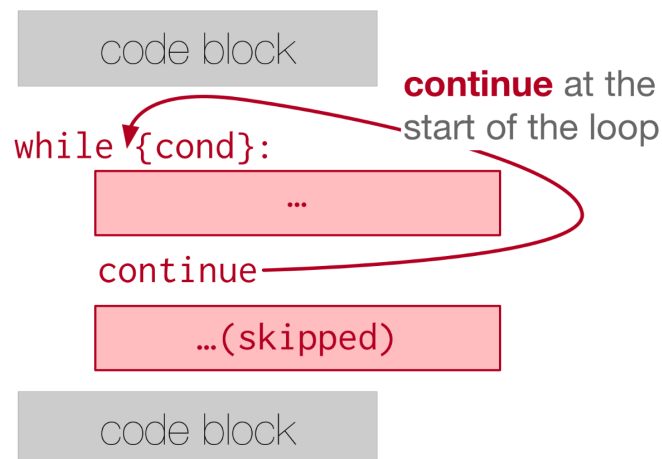
```
1 while True:
2     print("this phrase will always print")
3     break
4     print("Does this phrase print?")
5
6 print("We are done with the while loop.")
7
```

this phrase will always print  
We are done with the while loop.

Activity: 1 -- ActiveCode (ac14\_5\_1)

We can see here how the print statement right after `break` is not executed. In fact, without using `break`, we have no way to stop the while loop because the condition is always set to `True`!

`continue` is the other keyword that can control the flow of iteration. Using `continue` allows the program to immediately "continue" with the next iteration. The program will skip the rest of the iteration, recheck the condition, and maybe does another iteration depending on the condition set for the while loop.



Save & Run Original - 1 of 1 Show in CodeLens

```
1 x = 0
2 while x < 10:
3     print("we are incrementing x")
4     if x % 2 == 0:
5         x += 3
```

```
6         continue
7     if x % 3 == 0:
8         x += 5
9         x += 1
10 print("Done with our loop! X has the value: " + str(x))
11
```

we are incrementing x  
we are incrementing x  
we are incrementing x  
Done with our loop! X has the value: 15

Activity: 2 -- ActiveCode (ac14\_5\_2)

Try stepping through the above code in codepens to watch the order that the code is executed in. Notice in the first iteration how the program doesn't move to evaluate the divisible by 3 statement or add 1 to x. Instead, it continues to the next iteration.

You have attempted 3 of 2 activities on this page



14.4. Randomly Walking Turtles">  
Randomly Walking Turtles">

✔ Completed. Well Done!

14.6. Infinite Loops">

14.6. Infinite Loops">Next Section - 14.6. Infinite Loops