

LEARN CODING

ale66

LOOPS

THE **while** INSTRUCTION

We saw **for** to *iterate* over a sequence

A block of code is executed as many times as the number of elements of the sequence

```
1 writers = ['Hemingway', 'Dickens', 'King']  
2  
3 for w in writers:  
4     print(w)
```

With the while loop we execute a block only as long as a given expression is true

```
1 i = 1
2 while i < 6:
3     print(i)
4     i += 1
```

increment is not automatic

the expression is evaluated *before* executing the block of code

Try the example above on pythontutor.com

With the break statement we can stop the loop even if the while condition is true:

```
1 i = 1
2 while i < 6:
3     if i == 3:
4         break
5     i += 1
```

INDEFINITE LOOPS

Would execute forever, or until a break condition is met

```
1 while True:
2
3     command = input('Please enter a command, type 'exit' to stop.')
4
5     if command == 'exit':
6         break
7     else:
8         # implement commands here ..
```

What is this code doing?

```
1 n = 5
2
3 while n > 0 :
4
5     print(n)
6
7     n = n-1    # or n -= 1
8
9 print('Blastoff!')
```

QUIZ: WHAT'S WRONG WITH THIS LOOP?

```
1 n = 10
2
3 while n > 0 :
4     print('Hi')
5
6 print('End')
```



Execution won't end as n is never decreased

INDEFINITE LOOPS

- some **while** loops are called *indefinite* because they keep going until a logical condition becomes false
- but will it ever happen?
- codes seen so far are easy to examine to see whether they will terminate or become *infinite*
- termination analysis is about checking, *before running it*, that our code will always reach the final instruction and control will be back to the operating system
- this activity cannot be automated

for vs. while

Use **for** to automate work over collections, here represented by iterables

In principle, all elements will be examined

Use **while** to search for a specific element: it stops as soon as it's found

Also use **while** or to act *conditionally* and stop as soon as the condition is not true anymore

while IN LISTS

```
1 my_list = [1, 3, 7, 9]
2
3 count = 0 # will work as index
4
5 while(True):
6     print(my_list[count])
7
8     if (count == len(my_list) - 1):
9         print("end")
10        break
11
12    count += 1
```

```
1 for n in my_list:
2     print(n)
```