

Lecture 6 - Loops

I. Loop Basics

* In programming there are 3 ways to manipulate code

- (1) order of execution
- (2) Decision / logic
↳ if statements

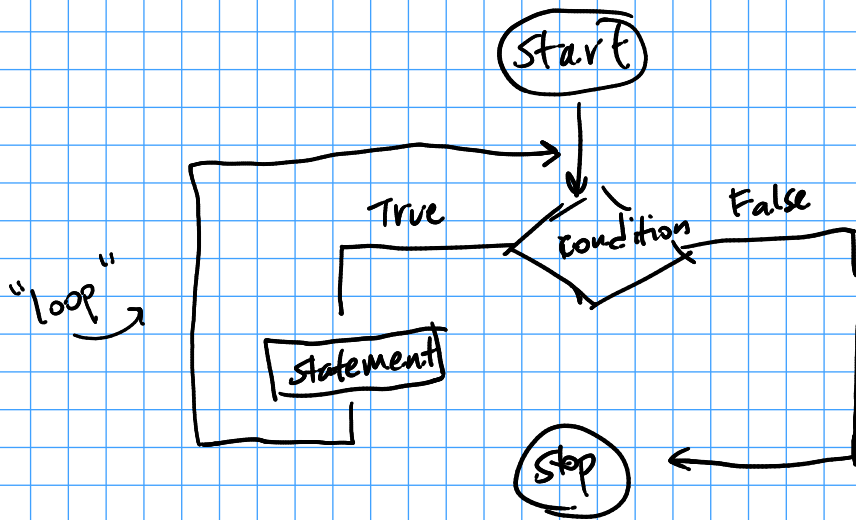
(3) Loops

⊛ repeat code

$x = 3$
 $y = x * 2$



* Loops are simply elements of code that allow us to repeat a task.

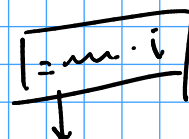


P R
"infinite loop"

* A loop is a way to avoid copy/paste.
MS Excel $\Rightarrow$ No true loops.

⊛ video outline
see link.

$i = m \cdot i$



"unrolling a loop"

II. Loop syntax in Python

2 kinds. —

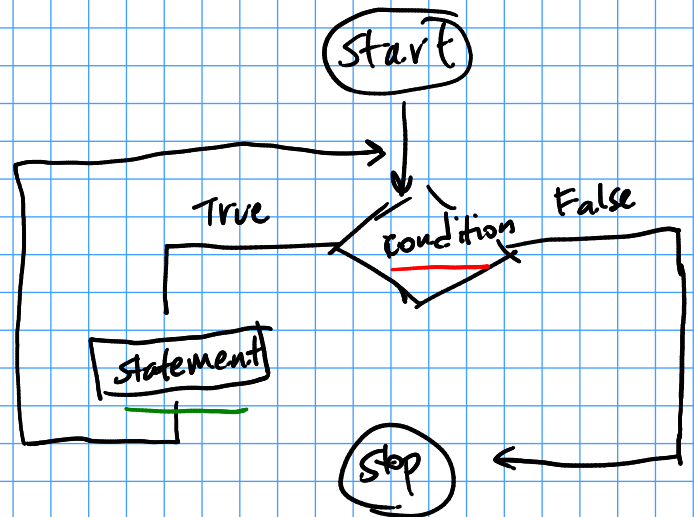
(1) while loop.

start

while (condition):

statement

stop



Example:

i=0 # i, iterator

while (i < 5):

print(i)

i+=1 # increment

i+1 → i

+= 2

- = 2

* =

/ =

Console

>> 0

>> 1

>> 2

>> 3

>> 4

X Don't print 5

try yourself:

i=0
while (i < 5):

i+=1

print(i)

>> 1

>> 2

>> 3

>> 4

>> 5

Example 2

while (True):

print("Forever")

↑ "infinite loop"

Control-C in Spyder

Break/stop → kill signal

* Programmers are lazy, so they came up w/
a more convenient type of loop that
includes a counter. → "For" loop

(2) For loop

$i, j, \dots$
for counter in range(N):
 Statement.

Equiv.
↔

integer
↓
counter = 0
while (counter < N):
 — statement
 — counter += 1

range variable:

range(N) = 0, 1, ..., N-1
 [0, N)

Python Example video talks about range.

⊛ Link in desc.

Example:

0, 1, 2, 3, 4

```
for i in range(5):  
    print(i)
```

↗ No infinite loops!

>> 0

>> 1

>> 2

>> 3

>> 4

⊛ Programmers start
counting @ zero!!

⊛ Tip • write them out by hand

- print inside loops
- play around