Control

Announcements

Print and None

(Demo)

Example: Print Then Return

Implement a function h(x) that first prints, then returns, the value of f(x).

```
def h(x):
    return print(f(x))
```

(A)

```
def h(x):
    print(f(x))
    return f(x)
```

(B)

```
def h(x):
    y = f(x)
    print(y)
    return y
```

(C)

What's a function f for which implementations (B) and (C) would have different behavior?

```
>>> h(2)
...
```

```
>>> h(2)
...
```

(Demo)

Control

Conditional Statements

Conditional statements (often called "If" Statements) contain statements that may or may not be evaluated.

		x=10	x=1	x=-1

`if x > 2:`
`    print('big')`
`if x > 0:`
`    print('positive')`
	Two separate (unrelated) conditional statements	big
		positive
			positive	

`if x > 2:`
`    print('big')`
`elif x > 0:`
`    print(‘less big’)`
	One statement with two clauses: if and elif
	Only one body can ever be executed
		big	less big	

`if x > 2:`
`    print('big')`
`elif x > 0:`
`    print(‘less big')`
`else:`
`    print(‘not pos')`
	One statement with three clauses: if, elif, else
	Only one body can ever be executed
		big	less big	not pos

While Statements

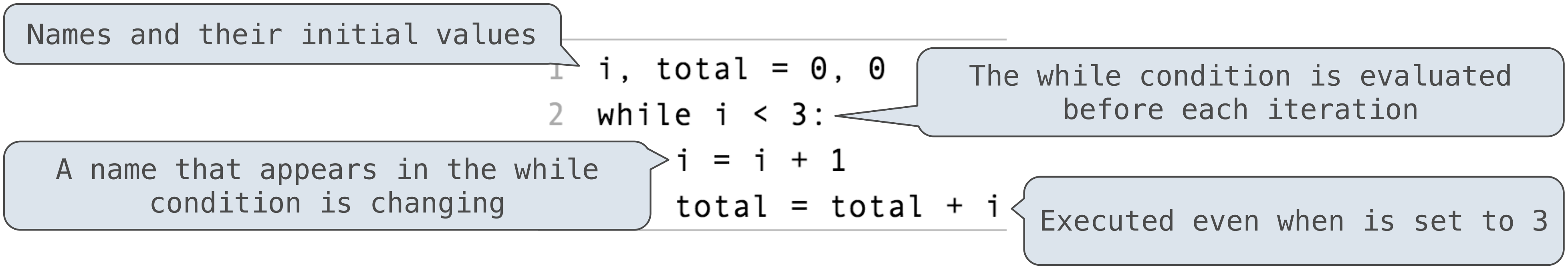
While statements contain statements that are repeated as long as some condition is true.

Important considerations:

- How many separate names are needed and what do they mean?

- The while condition **must eventually become a false value** for the statement to end (unless there is a return statement inside the while body).

- Once the while condition is evaluated, the entire body is executed.



Names and their initial values

```
i, total = 0, 0
while i < 3:
    i = i + 1
    total = total + i
```

The while condition is evaluated before each iteration

A name that appears in the while condition is changing

Executed even when is set to 3

Example: Prime Factorization

Prime Factorization

Each positive integer n has a set of prime factors: primes whose product is n

```
...
8  = 2 * 2 * 2
9  = 3 * 3
10 = 2 * 5
11 = 11
12 = 2 * 2 * 3
...
```

How can we determine whether a number is divisible by another?

One approach: Find the smallest prime factor of n, then divide by it

```
858  = 2 * 429  = 2 * 3 * 143  = 2 * 3 * 11 * 13
```

(Demo)