Functions

Announcements

Did you watch the lecture videos for today and follow along by typing Python?

Yep, both!

0%

I watched them, but passively (didn't type anything)

0%

I watched them all, but faster than 1x

0%

I watched some

0%

I watched none

0%

CS 10: The Beauty and Joy of Computing

Designed for students without prior experience

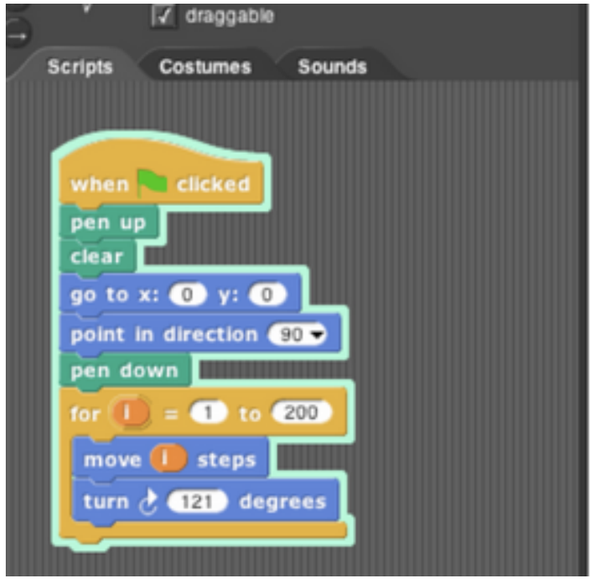
A programming environment created by Berkeley, now used in courses around the world and online

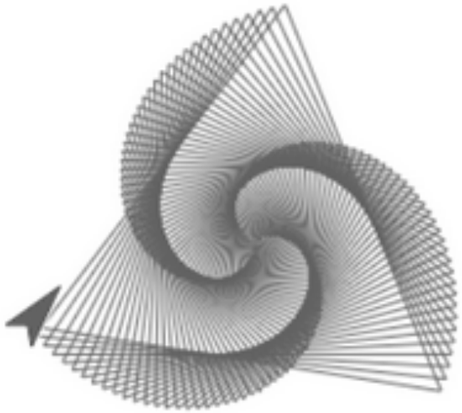
An introduction to fundamentals (& Python) that sets students up for success in CS 61A

If you might switch to CS 10, start attending its lectures ASAP and enroll soon before it fills!

More info: http://cs10.org/

Dan is teaching it also!!





Types of expressions

An expression describes a computation and evaluates to a value

`2` `25 + 3` `pi` `'are you human'`

Number or Numeral

Name

String

2^{100} $\frac{6}{23}$ $\sin \pi$ $\log_2 1024$

$\sqrt{3493161}$

$f(x)$

$7 \bmod 2$ $\lim_{x \to \infty} \frac{1}{x}$

$|-1869|$ $\sum_{i=1}^{100} i$ $\binom{69}{18}$

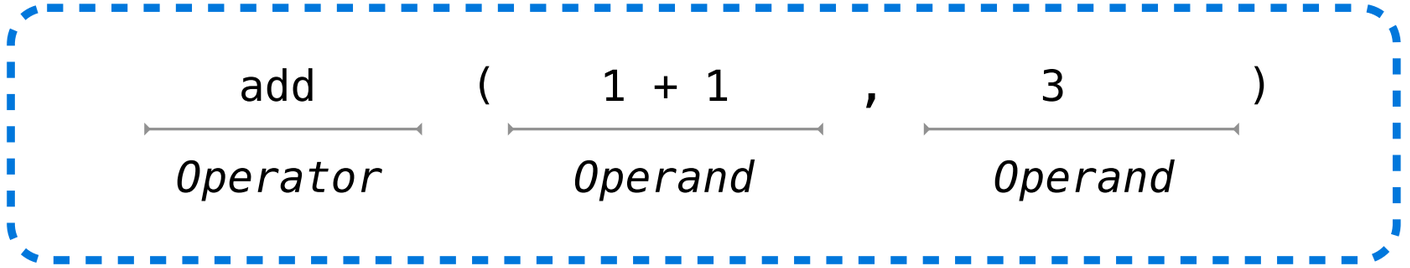
Call Expressions in Python

(Demo of Editor vs Terminal)

Anatomy of a Call Expression

Expression: describes how to compute something, evaluates to a **value**

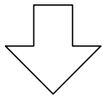
Call Expression



Operator and Operands are also expressions!

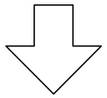
Evaluation Procedure for call expressions

(1) Evaluate operator



function

(2) Evaluate each operand



argument

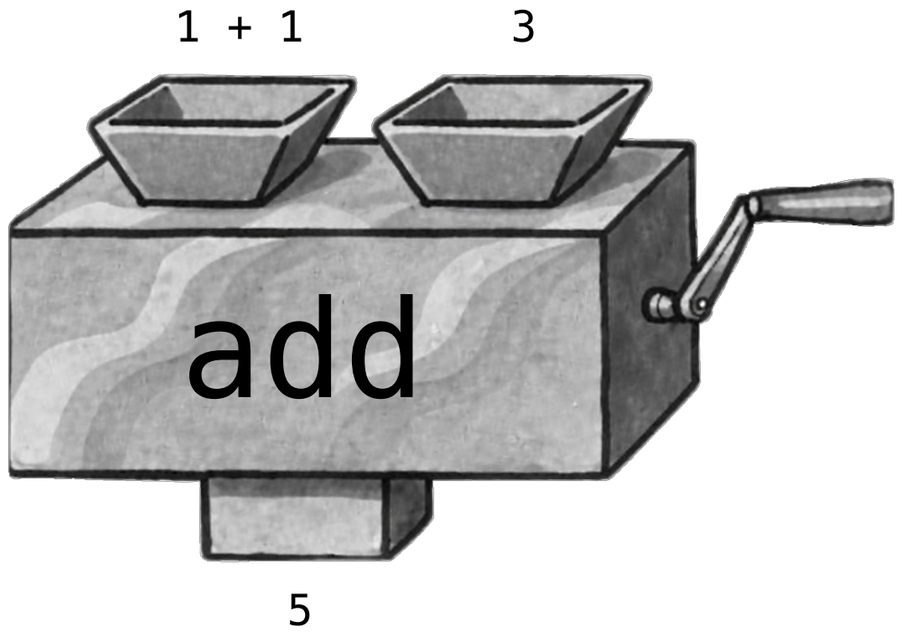
(3) **Apply** the **function** to the **arguments**

Function Call thought of as Function Machine

Call Expression

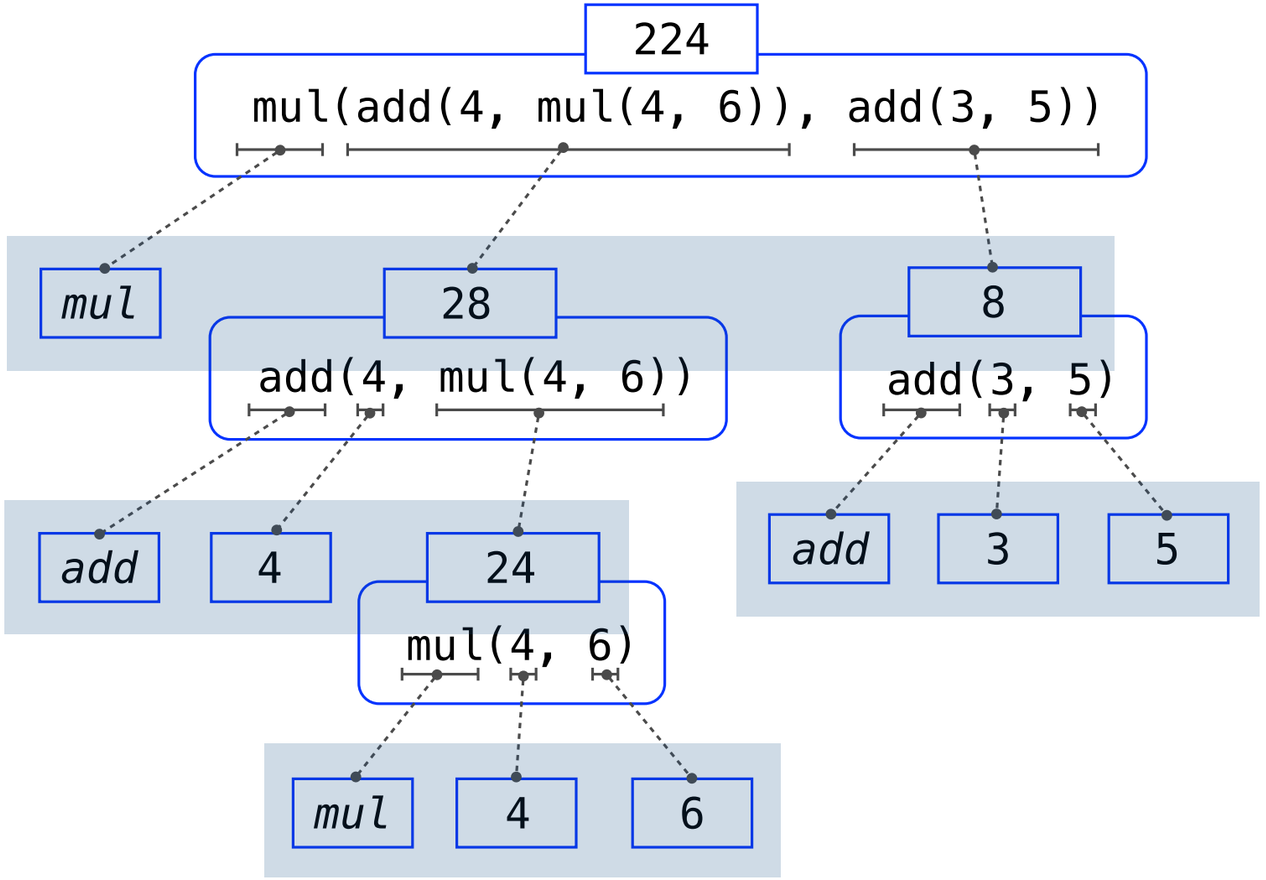
```
add ( 1 + 1 , 3 )
```

Operator *Operand* *Operand*

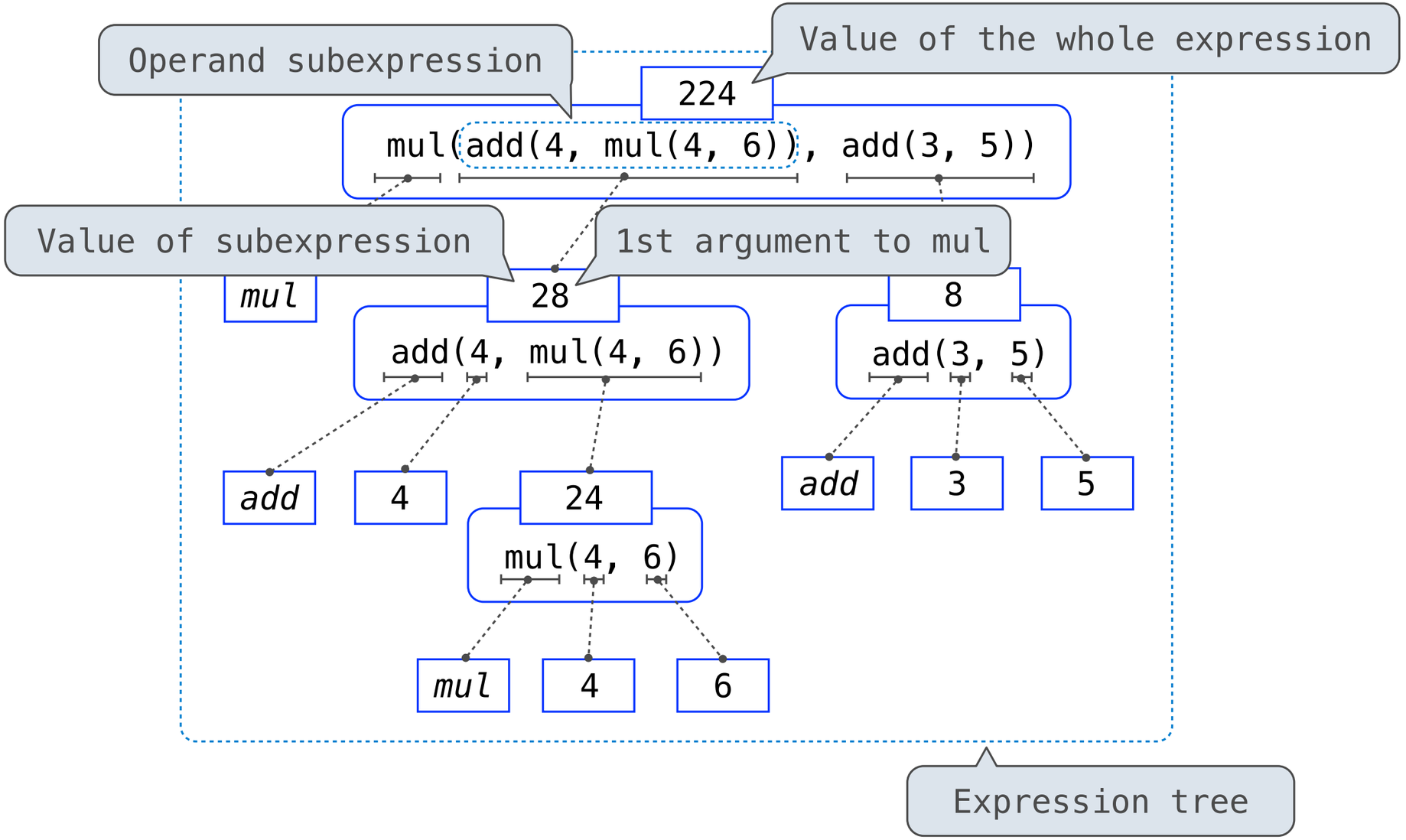


Evaluating Nested Expressions

```
(1)Evaluate operator
(2)Evaluate each operand
(3)Apply the function to
   the arguments
```

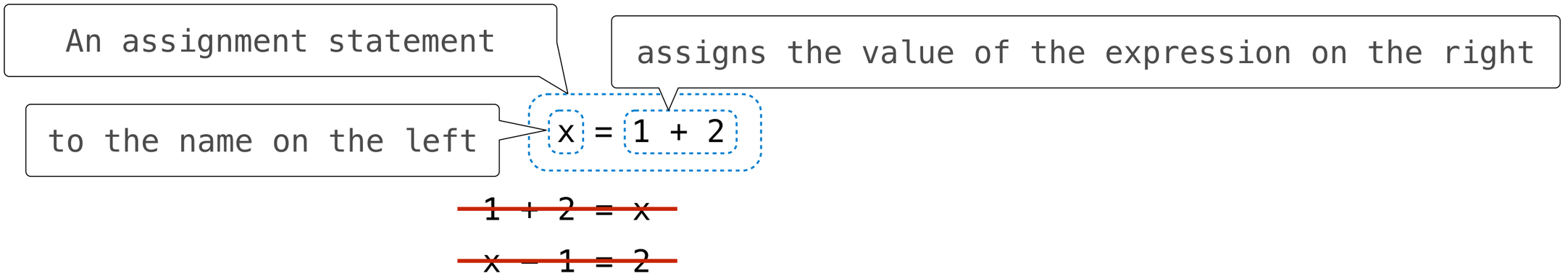


Evaluating Nested Expressions



Assignment Statements

Assignment Statements



The expression (right) is evaluated, and its value is assigned to the name (left).

```
>>> x = 2
>>> y = x + 1
>>> y
3
>>> x = 5
>>> x
5
>>> y
3
```

(Demo)