DATA 8

Fall 2016

Lecture 3, August 31

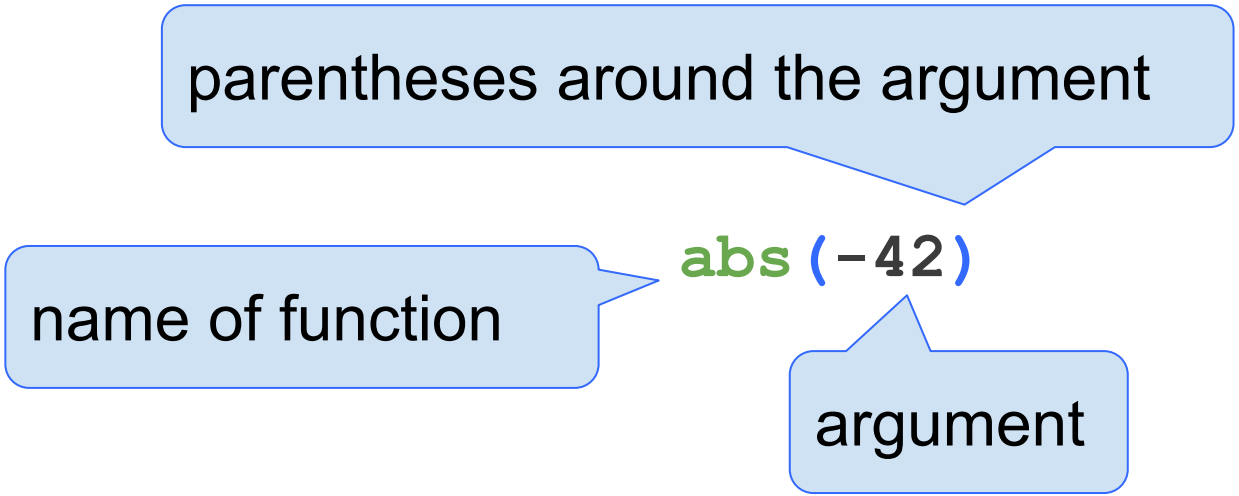
Arrays and Other Types

Slides created by Ani Adhikari and John DeNero

Announcements

- Office hours are in Weekly Schedule in the top menu bar of data8.org
- Homework 1 is due on Thursday at 5 pm; you get a bonus point for turning it in today by 5 pm.
- Submission instructions are in the post “HW 1 Posted” on Piazza and have also been sent by email.

Call Expressions



returns the value 42

Multiple arguments

- Separated by commas:
  - `max(3, 8, 1)` returns 8
- Some arguments are optional:
  - `round(5.7682, 3)` returns 5.768, rounding to 3 decimal places
  - `round(5.7682)` returns 6, rounding to the nearest integer, that is, to 0 decimal places
  - `round` converts to the nearest integer by **default** when the optional argument isn't specified

Modules

- Organized collections of functions or methods
- `math` module contains useful functions for math
- You have to import the module first:
  - `import math`
- Then call a function in it:
  - `math.sqrt(9)` returns 3.0

(Demo)

Different types of data

Longitude	Latitude	City	Direction	Survivors
32	54.8	Smolensk	Advance	145000
33.2	54.9	Dorogobouge	Advance	140000
34.4	55.5	Chjat	Advance	127100
37.6	55.8	Moscou	Advance	100000
34.3	55.2	Wixma	Retreat	55000
32	54.6	Smolensk	Retreat	24000
30.4	54.4	Orscha	Retreat	20000
26.8	54.3	Moiodexno	Retreat	12000

float:
decimal number

string:
text

int:
integer

Ints and Floats

Python has two real number types

- `int`: an integer of any size
- `float`: a fractional number with limited size

An `int` never has a decimal point

A `float` might be printed using scientific notation

Text and Strings

- A string value is a snippet of text of any length
  - `'a'`
  - `'Word97a23  *&*^nonsense'`
  - `"Who's on first?"`

- The + and * operators work on strings, but not in the way they do with numbers

(Demo)

Arrays

An array is a sequence that can be manipulated easily

- All elements of an array should have the same type
- Arithmetic is applied to each element individually
- Elementwise operations can be done on arrays of the same size

(Demo)

Ranges

A range is an array of consecutive numbers

- **`np.arange(end)`**:
  An array of increasing integers from 0 up to **`end`**
- **`np.arange(start, end)`**:
  An array of increasing integers from **`start`** up to **`end`**
- **`np.arange(start, end, step)`**:
  A range with **`step`** between consecutive values

The range always includes **`start`** but excludes **`end`**

(Demo)