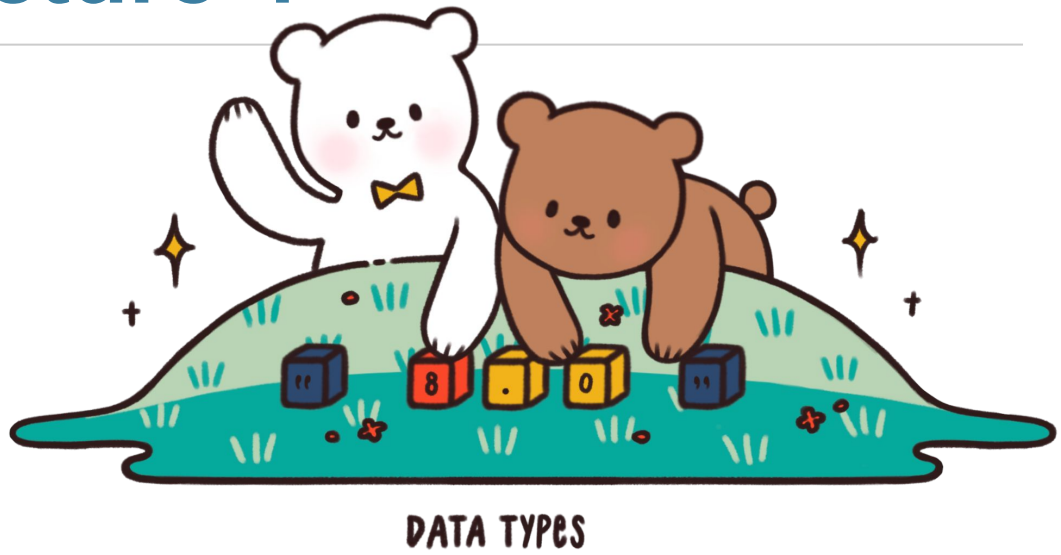
DATA 8
Spring 2022

Lecture 4



Announcements

- HW 1 is due tomorrow (Thursday) night @ 11:59pm
  - Submit by tonight 11:59pm for a bit of extra credit
  - An example of how to collaborate successfully
- Many students benefit from practice; sign up for small-group tutoring, which starts next week
  - https://tutoring.data8.org/

Weekly Goals

- Monday:
  - Python basics
  - Tables
- Today:
  - Types of data
  - Arrays
- Friday:
  - Creating new tables
  - Manipulating columns of tables

Why Names?

Names Help Document Code

- While complex expressions can be written without using any names, the result can be difficult to interpret
- Names document the meaning of quantities in a program and aid in interpreting arithmetic
- *Tip*: Choose names that would help you understand your code if you had to read it a year later

(Demo)

Numbers

(Demo)

Ints and Floats

Python has two real number types

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

An `int` never has a decimal point; a **float** always does

A `float` might be printed using scientific notation

Three limitations of float values:

- They have limited size (but the limit is huge)
- They have limited precision of 15-16 decimal places
- After arithmetic, the final few decimal places can be wrong

Strings

(Demo)

Text and Strings

A string value is a snippet of text of any length

- `'a'`
- `'word'`
- `"there can be 2 sentences. Here's the second!"`

Strings consisting of numbers can be converted to numbers

- `int('12')`
- `float('1.2')`

Any value can be converted to a string

- `str(5)`

Conversions

Strings that contain numbers can be converted to numbers

- `int('12')`
- `float('1.2')`
- ~~`float('one point two')`~~ `# Not a good idea!`

Any value can be converted to a string

- `str(5)`

Numbers can be converted to other numeric types

- `float(1)`
- `int(1.2) # DANGER: loses information!`

Discussion Question

Assume you have run the following statements:

```
x = 3
y = '4'
z = '5.6'
```

What's the source of the error in each example?

A. `x + y`
B. `x + int(y + z)`
C. `str(x) + int(y)`
D. `y + float(z)`

Types

(Demo)

Every value has a type

We've seen 5 types so far:

- `int: 2`
- `float: 2.2`
- `str: 'Red fish, blue fish'`
- `builtin_function_or_method: abs`
- `Table`

The `type` function can tell you the type of a value

- `type(2)`
- `type(2 + 2)`

An expression's "type" is based on its value, not how it looks

- `x = 2`
- `type(x)`

Arrays

(Demo)

Arrays

An array contains a sequence of values

- All elements of an array should have the same type
- Arithmetic is applied to each element individually
- Adding arrays adds elements (if same length!)
- A column of a table is an array

Friday: putting together arrays to make tables!