Data 8 Foundations of Data Science
Spring 2016

Discussion 2

Types

What does it mean for a value to have a type? For the purposes of this class, the type determines how the value is printed and combined with other values using arithmetic. Here are some common types you will encounter when working with data in Python.

Data Type	Explanation	Example
Integer	an integer value	5, -2
Float	a floating point, or decimal value	4.3, -1.0
String	a sequence of characters	'hello', '4.0'
Boolean	a truth value (either true or false)	True, False
Array	a container object that holds a number of values of a data type	array([1, 2, 3, 4])

Working with Different Data Types

Different data types behave differently when we combine them using arithmetic operators. Consider the following expressions. What do you think Python will print?

```
>>> 2 + 2
```

```
>>> 2 + 3.0
```

```
>>> '2' + '3'
```

```
>>> 'two' + 'three'
```

Converting Between Types

As you can see, types are very important in determining how Python treats a value when performing arithmetic operations. Its very important that we choose the right data types to represent our data. Luckily, Python has several functions available to us to convert between different types.

Function	Explanation	Example

str()	Converts a value to a String	`>>>str(3.0)`
		`'3.0'`
		`>>> str(1)`
		`'1'`

int()	Converts a number or string to an Integer value	`>>> int(5.6)`
		`5`
		`>>> int('2')`
		`2`

round()	Converts a number to a float, rounded to the nearest whole number	`>>> round(5.6)`
		`6.0`
		`>>> round(5.423, 2)`
		`5.42`

float()	Converts a number or string to a Float	`>>> float('-2.4')`
		`-2.4`
		`>>> float(1)`
		`1.0`

True and False Values
In Python, every value has an implicit truth value - it is either considered True or False. For the 5 data types that we have learned so far, these are the values that Python considers to be False.

Type	False Value
Integer	0
Float	0.0
String	”
Boolean	False
Array	[]

Any other value for these 5 types is considered True. In order to check whether a value is True or False, Python provides us with the bool() function. Here are a couple of examples of the bool() function in action:

```
>>> bool([])
False

>>> bool(1.0)
True
```

Practice Problems

1. Combine the following three values using arithmetic operators (+, -, *, /), parentheses, and the functions you learned into an expression that evaluates to the integer 51. Can you find more than one? You can use the names a, b, and c more than once or not at all.

a = '4'
b = 7.8
c = "one"

2. The mysterious case of the missing decimal points.

Assume we have executed the following statements:

a = '4 4'
b = '3 3'
c = '2 4'
d = '5 6'

The first number in each of the strings represents a digit before the decimal point, the *ones* place. The second number represents a digit after the decimal point, the *tenths* place. How would you write an expression that sums these numbers? (**Hint:** Ask your TA about .replace())

3. Write the shortest expression you can involving a and b, functions from discussion, operators, and parentheses that equals 838.

a = 8
b = 3