DATA 8

Spring 2018

Lecture 13

Iteration

Slides created by John DeNero (denero@berkeley.edu) and Ani Adhikari (adhikari@berkeley.edu)

Announcements

Comparison Operators

The result of a comparison expression is a `bool` value

```
x = 2                    y = 3
```

Assignment statements

```
x > 1                    x > y                    y >= 3

x == y                   x != 2                   2 < x < 5
```

Comparison expressions

`t.where(array_of_bool_values)` returns a table with only the rows of `t` for which the corresponding `bool` is `True`.

(Demo)

Aggregating Comparisons

Summing an array or list of bool values will count the True values only.

```
1    + 0     + 1          == 2
True + False + True       == 2
sum([1   , 0    , 1   ])  == 2
sum([True, False, True])  == 2
```

(Demo)

Predicates

(Demo)

Appending Arrays

A Longer Array

- **`np.append(array_1, value)`**
  - array with `value` appended to `array_1`
  - `value` has to be of the same type as elements of `array_1`
- **`np.append(array_1, array_2)`**
  - array with `array_2` appended to `array_1`
  - `array_2` elements must have the same type as `array_1` elements

(Demo)

Random Selection

Random Selection

`np.random.choice`

- Selects uniformly at random
- with replacement
- from an array,
- a specified number of times

```
np.random.choice(some_array, sample_size)
```

(Demo)

Control Statements

Control Statements

These statements *control* the sequence of computations that are performed in a program

- The keywords `if` and `for` begin control statements
- The purpose of `if` is to define functions that choose different behavior based on their arguments
- The purpose of `for` is to perform a computation for every element in a list or array

(Demo)