

PFTRotW

- **Review loops and lists (see Uppity.py again)**
 - Loop over sequence: string, file, list, "other"
 - Process each element, sometimes selectively
 - Toward understanding the power of lists
- **How do we debug?**
 - What do we do when we have code, but it's wrong?
 - How do we minimize head-beating-against-wall?
 - Mental model of code execution and Pythonic vocabulary
- **The power of randomness**

Anatomy of a Python list

- **Create list with brackets (values optional)**
 - `s1 = []`
 - `s2 = ["a", "b", "c"]`
 - `s3 = list("123")` #from an *iterable*
- **Lists are mutable and iterable**
 - Append to list, change value stored at index
 - `s2[1] = 5, s2.append(77)`
 - `for elem in list:`
- **Use function on lists: len, min, max, sum**
 - Operator: `in`

List methods

- **In object oriented programming methods are functions that operate on an object**
 - Inspect or change the object
 - Sometimes return values
- **List methods that inspect a list**
 - Search: count and index
- **List methods that mutate by adding or removing**
 - append, insert, pop, remove
- **List methods that re-arrange list**
 - reverse, sort

Indexing a list

- **Lists, like strings, start indexing with zero**
 - Strings are immutable, lists are mutable
- **For some problems, looping by index useful**
 - Use range function, range creates open-ended list
 - `range(0,10), range(5,20), range(10,100,5)`
 - Advice/warning: in Python 3 range doesn't create list
- **For some problems index and list useful**
 - Use `for x,y in enumerate(list):` idiom
 - Preview: tuple

