

INFO 450 Fall 2020

Week 2

Agenda

- Functions
- Lists
- Dicts
- Tuples
- Input/Output

Zen of Python

- Simple is better than complex.
- Complex is better than complicated.
- Readability counts
- There should be one
- Now is better than never.

way to do it

Lists

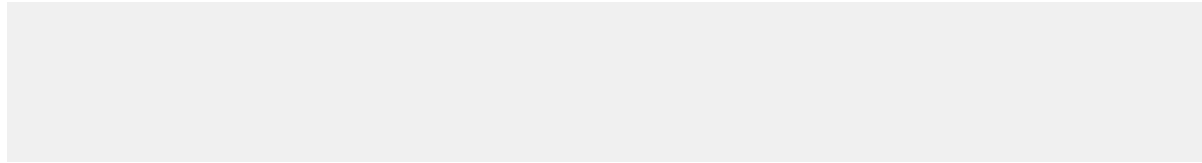
A list is a collection of items in a particular order.

You can put anything you want into a list

Bound by brackets []

Lists should be named something plural e.g. 'students', 'customers', 'deer'

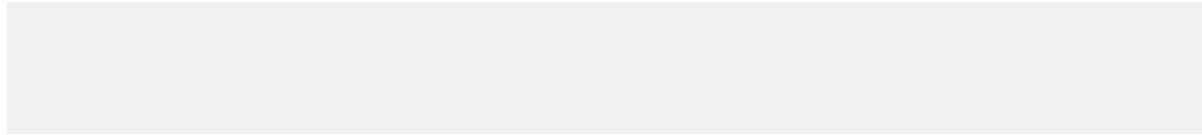
(See what I did there?)



- I probably won't always enter the `import logging` and `logging.basicConfig` calls in each code example, but, you will need to if you use logging.

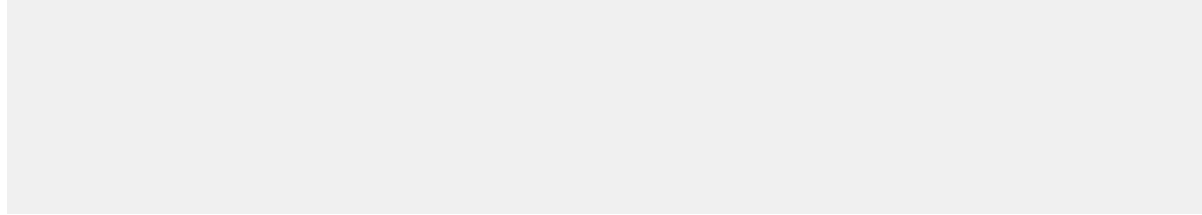
Printing a List

When you print a list in Python (print or logging, I prefer logging) - Python returns a string representation of your list, brackets included.



Access Elements in a list

The list contains elements in an order. A developer can access each item by its index:

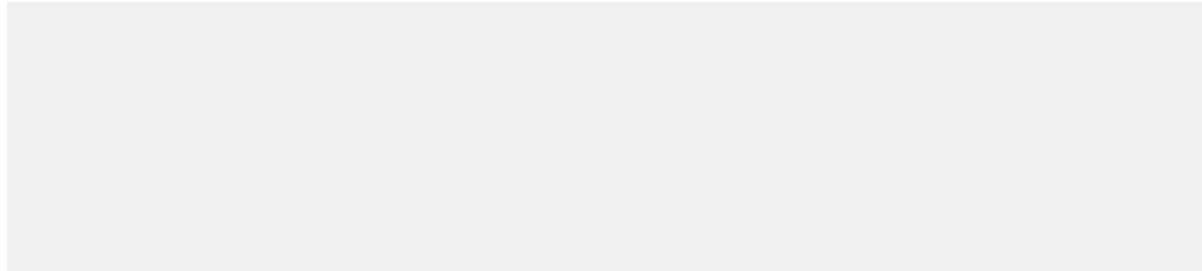


Python starts counting/indexing at 0

because it's a real programming language

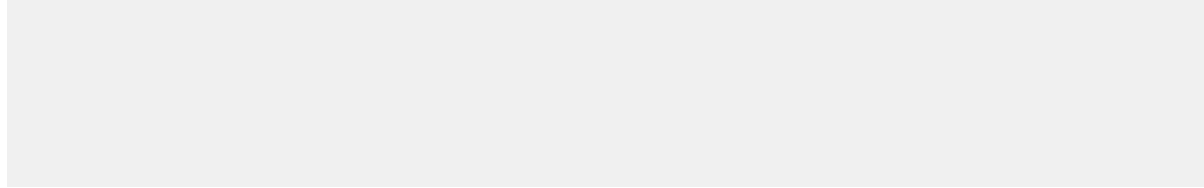
CRUD

- Create
- Read
- Update
- Delete

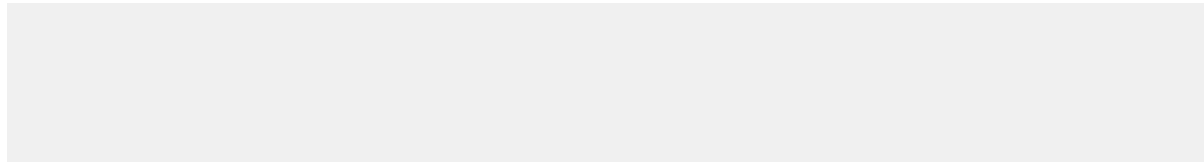


- `append(itm)` - Adds to the end of the list. New, highest index.

Dynamically creating and adding

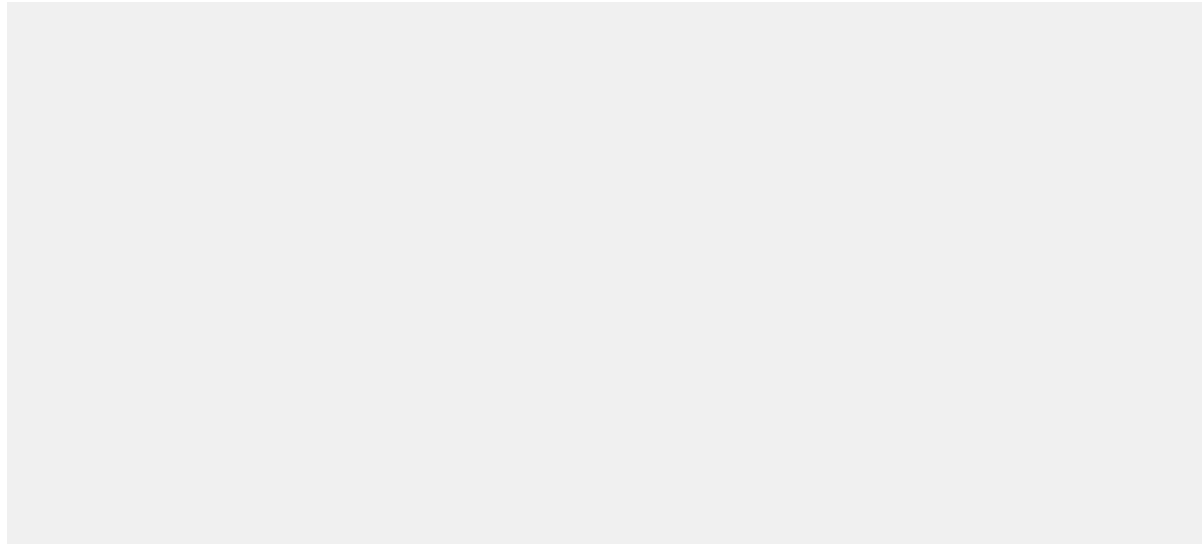


Inserting , not appending



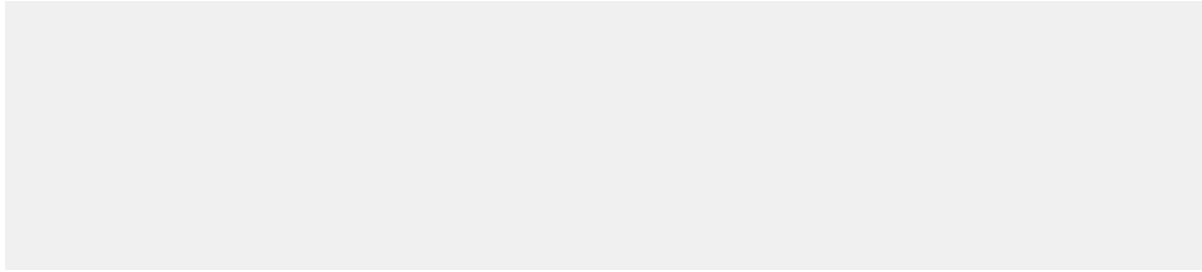
Deleting items from the list

Python has a keyword: `del` to delete. This deletes variables from memory, items in the list, etc



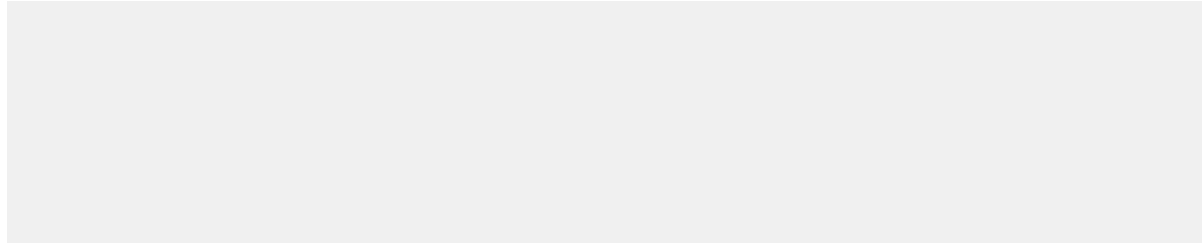
Organizing a list

- Permanent sort

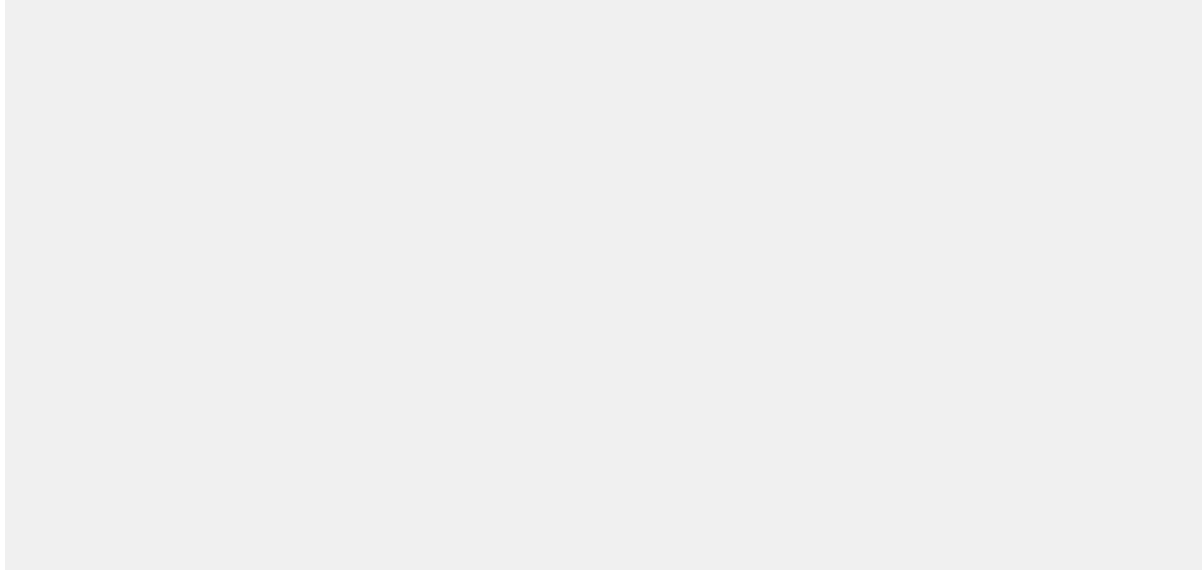


Temporary Sort

Maintain the original ordered list, but provide a sorted 'copy'



Other things



Working with lists

One of the most used control structures in Python has to do with iterating through every element in a list.

Example from the book: in a game, you might want to move every element on the screen.

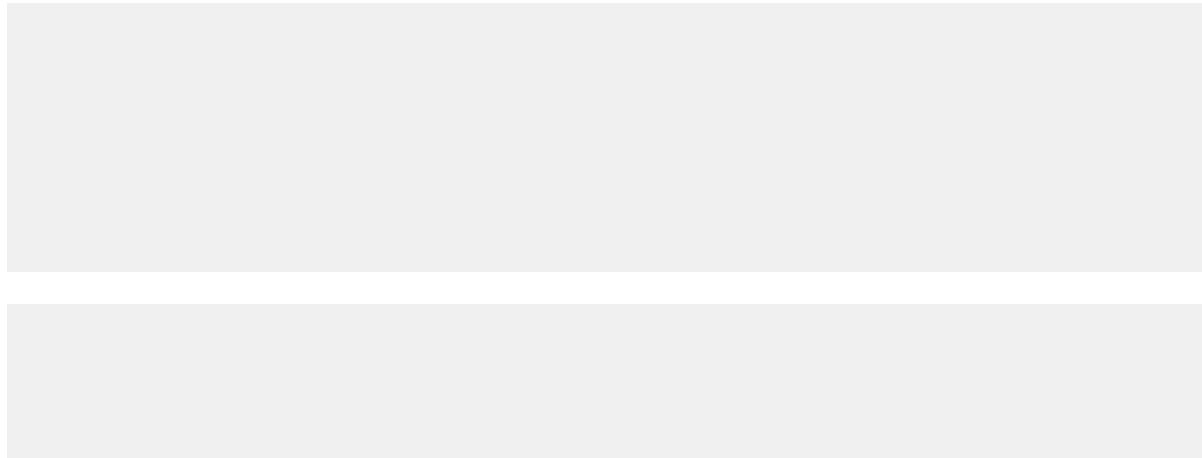
- Print out today's stock market
- List all orders
- Verify inventory

Old way

for

not really sure why I showed the old way.

The `for` loop construct allows you to not worry about how many items are in a list

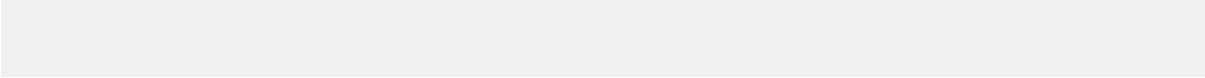


Lots of errors in the book

Indenting where you shouldn't

Forgetting to indent where you should

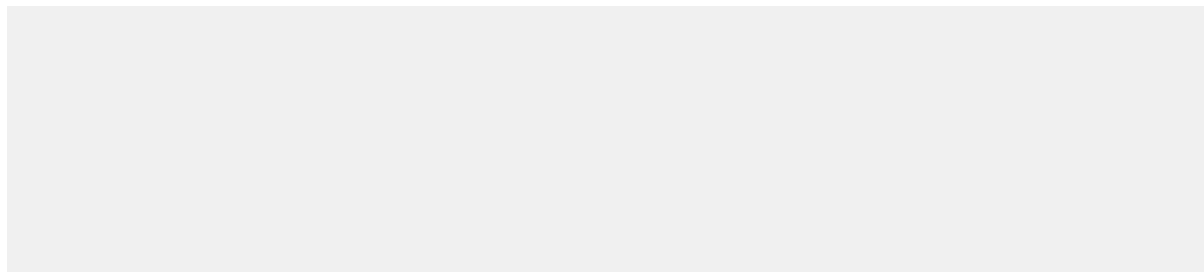
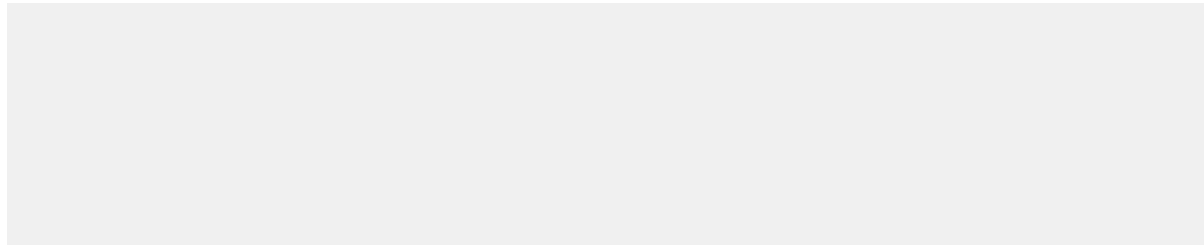
Forgetting the semi-colon



Numerical Lists

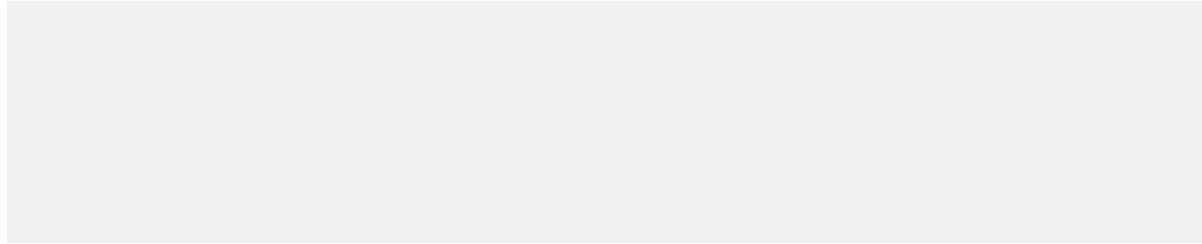
<https://docs.python.org/3/library/functions.html#func-range>

The `range` function generates a series of numbers



.... immutable sequence type?

Let's look at some Python fun to see what's going on.

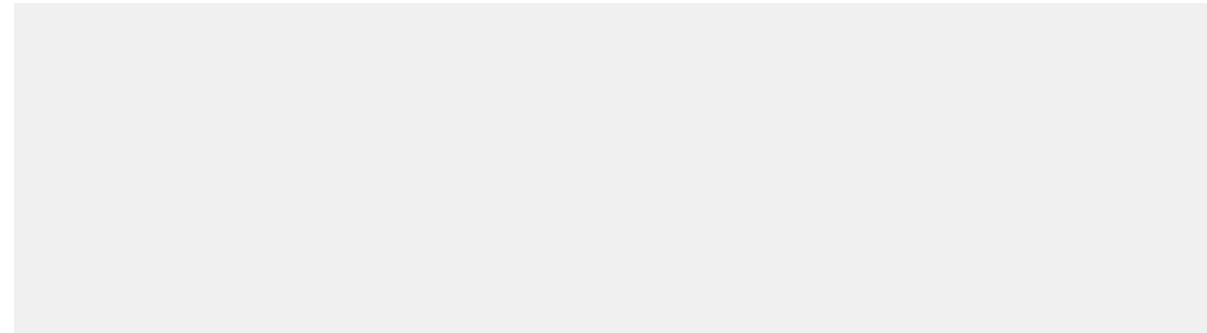


List Comprehension

Allows you to perform some actions on a list in one line of code.

Combines the for loop and creation of new elements into one line.

Book: "List comprehensions are not always presented to beginners, but I have included them here because you'll most likely see them as soon as you start looking at other peoples code."



```
import logging logging.basicConfig(level=logging.DEBUG)
def squared_threes(): return_value = []
```

