Topics

• Using lists to store related data
• Managing lists: Adding and deleting from related lists
• Calculating statistics on lists (mean)
• Sorting lists to find the median and the mode
• Converting strings to lists (split method)
Lists (Chapter 8 in Gaddis)

• A list is a sequence of values

• Lists are like strings in some ways
  A string is a sequence of characters

• Elements of lists can be anything
  Integers, strings, floating point numbers, booleans

• Lists can be nested
  Lists can contain lists, which can contain lists
Creating Lists

colors = [“Red”, “Blue”, “Green”]

numbers = [456, 42, -11, 45.7]

flags = [True, False, True, True]

mixed_stuff = [“Red”, 5, False, “Purple”]
How to add items to a list?

```python
>>> colors = ['red', 'blue'] + ['orange', 'green']
>>> colors
['red', 'blue', 'orange', 'green']
```

```python
>>> counts = [0] * 10  # initialize a list
>>> counts
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```
List method: append

The *append method* adds an item to the end of the list.

```python
>>> numbers = [1, 2, 3, 4, 5]
>>> numbers.append(6)
>>> numbers
[1, 2, 3, 4, 5, 6]
```
append loop

numbers = []
for num in range(10):
    numbers.append(num)

print(numbers)

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
List method: sort

The *sort method* reorders the list items in ascending order (numbers, strings).

```python
>>> numbers = [1003, 4.3, 33, 9, 33.7]
>>> numbers.sort()
>>> numbers
[4.3, 9, 33, 33.7, 1003]
```
List method: reverse

The *reverse method* reorders the list items in reverse order.

```python
>>> colors = ['red', 'orange', 'yellow', 'green', 'blue']
>>> colors.reverse()
>>> colors
['blue', 'green', 'yellow', 'orange', 'red']
```
List method: count

The *count method* returns the number of times an item occurs in a list.

```python
>>> fruits = ['apple', 'pear', 'apple']
>>> fruits.count('apple')
2
>>> fruits.count('banana')
0
```
List method: index

The *index method* returns the index of an item in a list. It throws an exception if the item is not in the list.

```python
fruit = ['apple', 'banana', 'grapes']
# Must test to avoid an exception!
idx = -1
if 'banana' in fruit:
    idx = fruit.index('banana')
```
How to delete items from a list?

```python
>>> fruits = ['apple', 'banana', 'kiwi', 'banana']
>>> fruits.remove('banana')
>>> fruits
['apple', 'kiwi', 'banana']
```

Removes only the first item!
Deleting items positional

>>> fruits = ['apple', 'kiwi', 'banana', 'grapes']
>>> del fruits[2]
>>> fruits
['apple', 'kiwi', 'grapes']
List method: clear

The *clear method* removes all elements of a list.

```python
>>> fruits = ['apple', 'banana', 'pear']
>>> fruits.clear()
>>> fruits
[]
```
List statistics

len(mylist)      # The number of items
sum(mylist)      # The sum of the items
max(mylist)      # The largest item
min(mylist)      # The smallest item
String method: split

```python
>>> taxday = "4/15/2014"
>>> taxdate = taxday.split('/')
>>> taxdate
['4', '25', '2014']
```

*split* creates a list from a string. By default, the argument is a space ` ' `.
Example: Calculating the Mode

clist = ['a', 'b', 'a', 'c', 'd', 'b', 'a', 'x']

1. Sort the list
clist = ['a', 'a', 'a', 'b', 'b', 'c', 'd', 'x']

2. Create another list without duplicates
newlist = ['a', 'b', 'c', 'd', 'x']

3. Create a corresponding list with word counts
counts = [3, 2, 1, 1, 1]

maxcount = max(counts)
indx = counts.index(maxcount)
mode = newlist[indx]