• Questions/Announcements

• Python data collections: lists
Lists – Why

For example, grades for grade_calculator
Lists – Why

For example, check passwords against frequently used passwords
Lists – Why

flexible collections of data

modeling real world
Lists – What

```python
>>> frequent_psw = ["abc123", "password", ""]

>>> if pwd in frequent_psw:
    return False
```
Lists – What

>>> frequent_psw = ["abc123", "password", ""]

>>> if pwd in frequent_psw:
    return False

>>> return not (pwd in frequent_psw)
Python lists can be heterogeneous

```python
>>> mixitup = ["hello", "goodbye", True, 99, 99.0, ['a']]

>>> mixitup

>>> [2, 4, 6, 8]
```
Lists are sequences

so the sequence operators and built in functions we’ve seen for strings work for lists, too –

  [], +, *, [:], len, max, min, in

>>> example = [5, True, ‘string’, 99.0]
>>> example[1:]
>>> example[:3]
>>> example[1:3]
Lists are sequences

so the sequence operators and built in functions we’ve seen for strings work for lists, too –

\[
[], +, *, [:], \text{len}, \text{max}, \text{min}, \text{in}
\]

also - del, sorted, sum

and – another string method, split -

```python
>>> 'C2O H2O'.split()
['C2O', 'H2O']
```
mixitup = [“hello”, “goodbye”, True, 99, 99.0]
len(mixitup)
“goodbye” in mixitup
“good” in mixitup
mixitup[1:3]
mixitup[::]
mixitup[:]
mixitup + mixitup
mixitup * 2
>>> stuff = [2, 25, 80, 12]

>>> sorted(stuff)

>>> sum(stuff)

>>> stuff.count(12)
```python
count = stuff.count(12)

def mycount(seq, v):
    ''' (sequence) -> int
    return # of occurrences of value v in sequence seq
    >>> mycount([1, 2, 2, 2], 2)
    3
    '''
    pass
    return #count
```
What kind of loop?

- while loop
- for loop
What kind of loop?

while loop – indefinite # of iterations
  for example: how long to double savings?

amount = 100
rate = .03
orig_amount = amount
while orig_amount < (amount * 2):
    amount += amount * rate
What kind of loop?

while loop – generally applicable... but

s = ‘example’
ctr = 0
while ctr < len(s):
    print(s[ctr])
What kind of loop?

for loop – sequential operator

```python
s = 'example'
for char in s:
    print(char)
```
What kind of loop?

while loop – indefinite # of iterations  for example: how long to double savings?

for loop – operator for sequences  definite number of iterations
• Questions/Announcements
• Python data collections: lists
What value of x that will cause mystery(x) to run forever?

def mystery(x):
    a = [0, 4, 0, 3, 2]
    while x > 0:
        x = a[x]
    return "Done"
def mycount(seq, v):
    '''(sequence) -> int

    return # of occurrences of value v in sequence seq
    '''
    v_ct = 0
    for ch in seq:
        if ch == v:
            v_ct += 1

    return v_ct
def evens(li):
    """ (list of numbers) -> list of numbers
    what does this function do?
    ""
    res = []
    for n in li:
        if isEven(n):  # if (n%2) == 0
            res.append(n)
    return res

>>> testlist = [1, 2, 2, 4, 5, 5]
>>> evens(testlist)
Lists are mutable!

>>> stuff = [2, 25, 80, 12]

>>> stuff[1] = 99

>>> stuff[stuff[0]] = stuff[3]

>>> stuff

[2, 99, 80, 12]
>>> stuff = [2, 25, 80, 12]
>>> stuff[stuff[0]] = stuff[3]
>>> stuff

>>> stuff.count(12)

>>> del stuff[0]

>>> stuff
Lists are mutable

```python
mixitup = ["hello", "goodbye", True, 99, 99.0]
mixitup[0] = "bonjour"
mixitup[1] = "adios"
del mixitup[-1]
```