

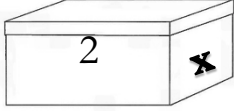
ECS10

2-6

Variable

- Metaphor:


x = 2


- Reality: the integer 2 is saved in memory, in the location program calls x.

Storing a lot of stuff


- Lots of boxes is messy.

Jan = 31
 Feb = 28
 Mar = 31
 Apr = 30
 ...


- So are lots of variables.

Organized memory

- List organizes memory.
- One name for whole filing cabinet.
- Each drawer has a number in the filing cabinet.
- Modifying a list element = opening one drawer, replacing contents with something else.



Changing list element

```
freq[result] = freq[result]+1
```

- Just like:

$$x = x+1$$
- Take value out of drawer, do computation, put result into drawer.

But you can grow filing cabinet...

- The append() method sticks a new element onto the end of a list.

```
>>> shop=["cabbage","tea","yoghurt"]
>>> shop.append("bread")
>>> shop
["cabbage","tea","yoghurt","bread"]
```

- Notice you don't need an assignment statement.

Building a list of five zeros

```
i = 0
zeroList = [] # the empty list
while i < 5:
    zeroList.append(0)
    i = i+1
```

Special Python trick

```
zeroList = [0]*5
```

- Just the same as the previous program.

Mutable

- Lists are mutable. Strings aren't.

```
shop.append("kholrabbi") # changes shop
```

- Strings are not mutable

```
line.strip() # does not change line
```

- What should the line of the program be if you want to change the string contained in line?

Mutable

```
shop[3] = "beer"
```

- Perfectly OK

```
line[-1] = "\n"
```

- Crashes!

tuples

- A tuple is just like a list, is a sequence, but NOT mutable
- The in operator works, indexing works, the length function works, concatenation works
- The append method does not work.
- Written with () instead of []

```
[ 5, 6, 7 ] # list
```

```
( 5, 6, 7 ) # tuple
```

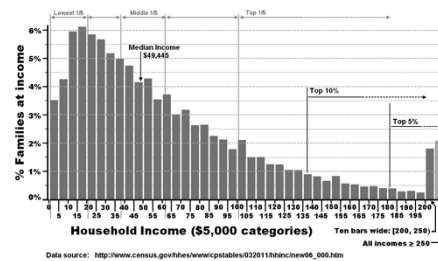
In assignment....

- Make a file reading loop
- For every line
 - extract the temp anomaly data
 - Use the append method to build up a list of temperature anomalies

Counting flips

- What does exact distribution of number of flips look like?
- Let's make a graph, like we will do with temperature.
- Run experiment 100K (10,000) times, make a histogram of number of flips required.

Histogram



.csv file

- Text file
- Each line ends with newline character, “\n”
- Data items on line separated by commas

1888,-0.566

1889,-0.698

Writing an output file

- Need to open file

```
outFile = open("histo.csv","w")
```
- “w” means write
- write() method takes a SINGLE STRING as input
- To get lots of stuff into single string, concatenate

```
outFile.write(str(i)+","+str(freq[i])+"\n")
```
- End text file lines with newline!
- write is not so nice as print!

str() function

str(22)

- Is an expression whose value is the string “22”

str(7/3)

- Is an expression whose value is “2.333333333333”

“{:.2f}”.format(7/3)

- Is an expression whose value is “2.333”