### 13. Lists of Numbers

Topi cs:

Lists of numbers Lists and Strings List Methods Setting up Lists Functions that return a list

### We Have Seen Them Before

Recall that the rgb encoding of a color involves a triplet of numbers:

MyColor = [.3, .4, .5]

DrawDisk(1,2,color=MyColor)

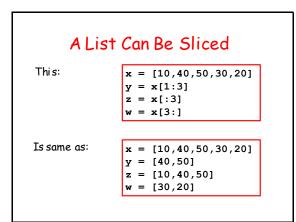
It is a way of assembling a collection of numbers.

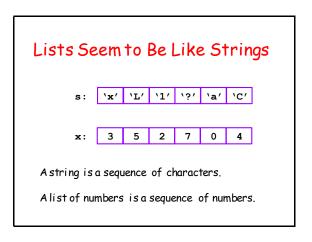
## A List has a Length The following would assign the value of 5 to the variable n: x = [3.0, 5.0, -1.0, 0.0, 3.14]n = len(x)

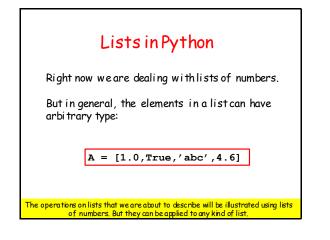
### The Entries in a List Can Be Accessed Using Subscripts

The following would assign the value of -1.0 to the variable a:

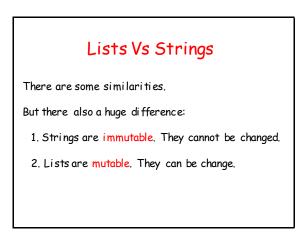
x = [3.0, 5.0, -1.0, 0.0, 3.14]a = x[2]

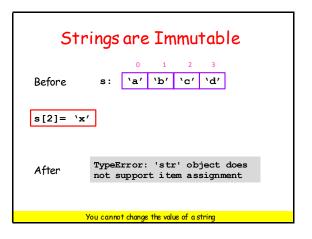


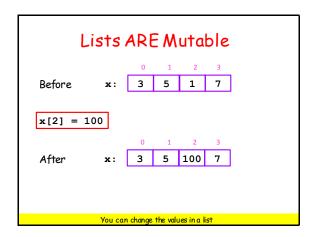


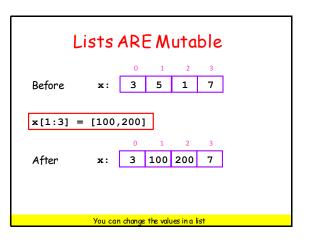


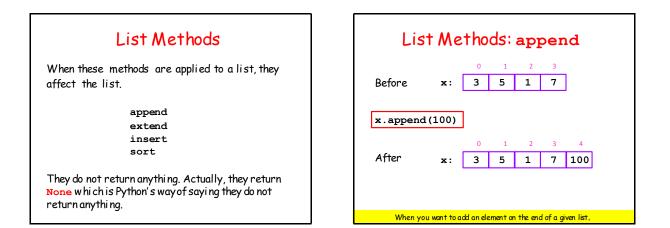
Informal:	0     1     2     3       x:     3     5     1     7
Formal:	x> 0> 3 1> 5 2> 1 hat shows 3> 7
A state diagram tl the "map" from in elements.	

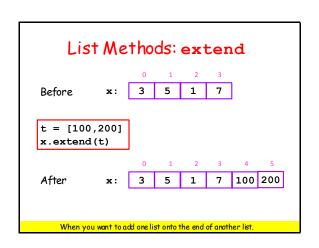


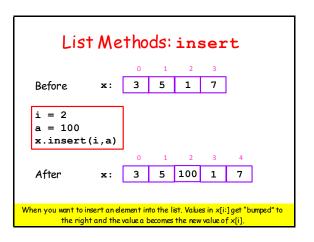


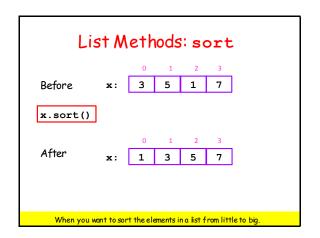


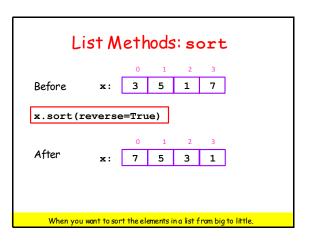


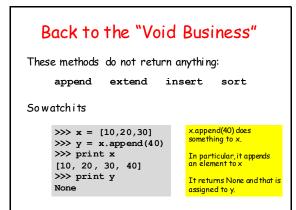




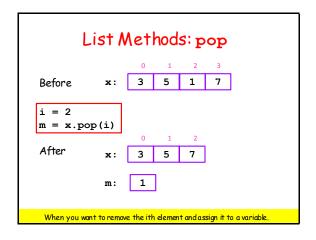




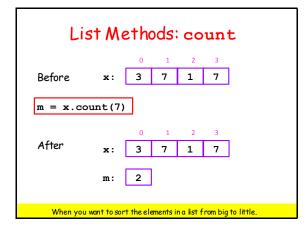


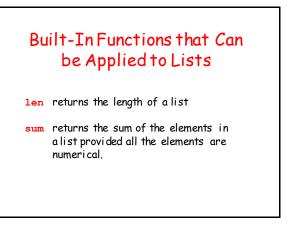


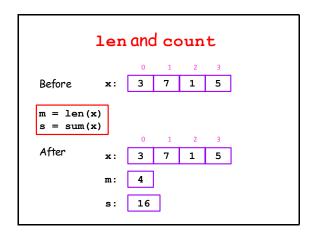




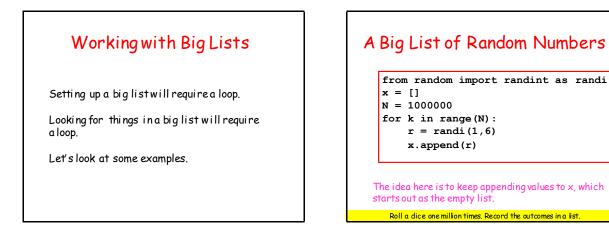


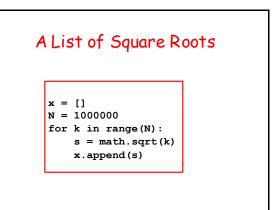












### This Does Not Work

```
from random import randint as randi
x = []
N = 1000000
for k in range(N):
    r = randi(1,6)
    x[k]=r
```

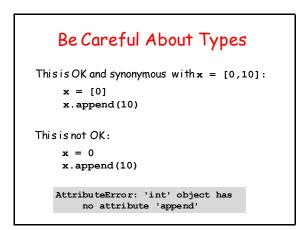
x[k] = rIndexError: list assignment index out of range

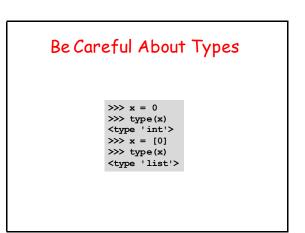
#### A Random Walk from random import randint as randi x = [0] k = 0 # x[k] is robot's location after k hops while abs(x[k])<=10: # Flip a coin and hop right or left r = randi (1,2) if r==1:

## new\_x = x[k]+1 else: new\_x = x[k]-1 k = k+1

x.append(new\_x)

# A Random Walk from random import randint as randi x = [0] k = 0 # x[k] is robot's location after k hops while abs(x[k])<=10: # Flip a coin and hop right or left r = randi(1,2) if r==1: new\_x = x[k]+1 else: new\_x = x[k]-1 k = k+1 x.append(new\_x)</pre>





## Functions and ListsALet's start with a function that returns a list.frIn particular, a function that returns a<br/>list of random integers from a given<br/>interval.frThen we will use that function to estimate<br/>various probabilities when a pair of dice are<br/>rolled.fr

### A List of Random Integers

```
from random import randint as randi
def randiList(L,R,n):
    """" Returns a length-n list of
    random integers from interval [L,R]
    PreC: L,R,n ints with L<=R and n>=1
    """
    x = []
    for k in range(n):
        r = randi(L,R)
        x.append(r)
    return x
```

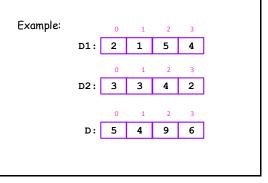
### Outcomes from Two Dice Rolls

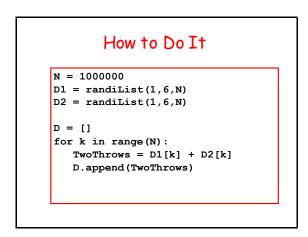
Roll a pair of dice N times

Store the outcomes of each dice roll in a pair of length-N lists.

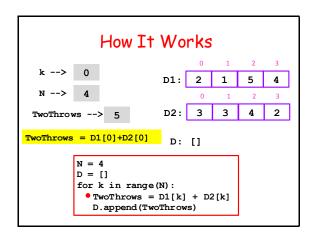
Then using those two lists, create a third list that is the sum of the outcomes in another list.

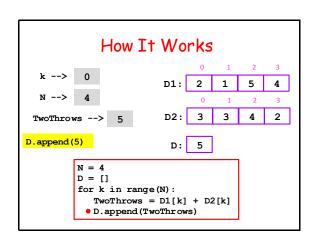
### Outcomes from Two Dice Rolls

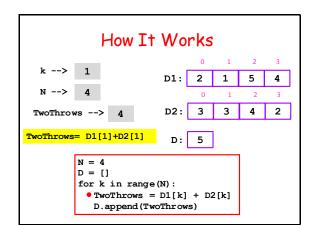


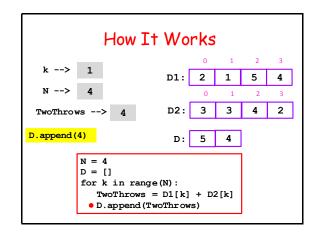


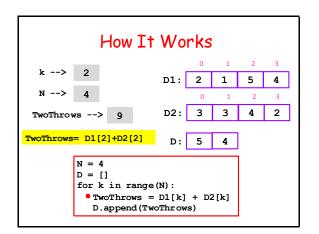
How It Works								
		0	1	2	3			
k> 0	D1:	2	1	5	4			
N> 4		0	1	2	3			
	D2:	3	3	4	2			
At the start of N = D = for								

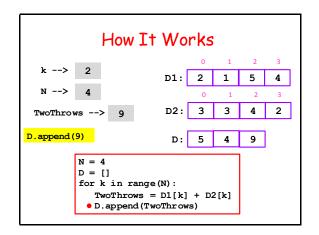


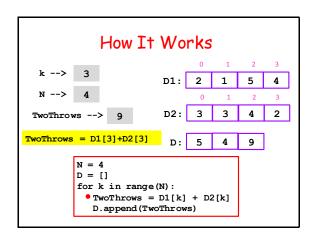


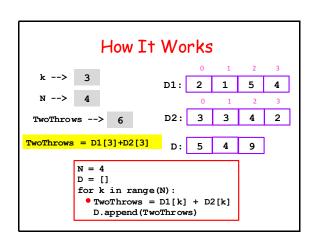


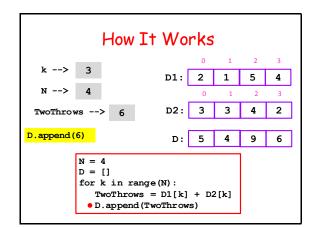


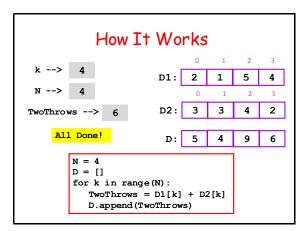


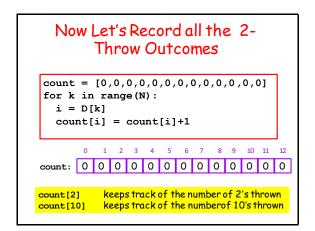


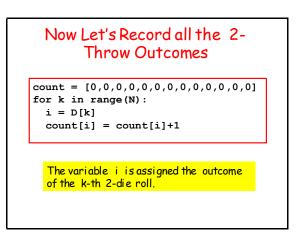


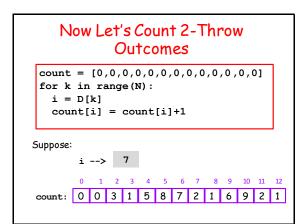












	Count 2-Throw tcomes
<pre>count = [0,0,0,0,0 for k in range(N</pre>	
Suppose	i> 7
then the assignment effectively says	<pre>count[i] = count[i]+1 count[7] = count[7]+1</pre>

N	Now Let's Count 2-Throw Outcomes												
for k i =	<pre>count = [0,0,0,0,0,0,0,0,0,0,0,0,0] for k in range(N):     i = D[k]     count[i] = count[i]+1</pre>												
	i	>		7									
Before:	0	1	2	3	4	5	6	7	8	9	10	11	12
count:	0	0	3	1	5	8	7	2	1	6	9	2	1
After:	0	1	2	3	4	5	6	7	8	9	10	11	12
count:	0	0	3	1	5	8	7	3	1	6	9	2	1

### Now Let's Count 2-Throw Outcomes

count = [0,0,0,0,0,0,0,0,0,0,0,0,0]
for k in range(N):
 i = D[k]
 count[i] = count[i]+1

Sample Results	, N = 10000
	k count[k] 
<pre>for k in range(2,13):     print k,count[k]</pre>	3 629 4 820 5 1100 6 1399
	7 1650 8 1321 9 1149 10 820
	11 527 12 292
	12 292