

CSE 20

5-21-20

Pa6: ext. 1 last day

Pa7: Due next Thurs.

Ex. Dice.Py, DiceProbability.Py

outcome of throwing 2 dice:

sum =	2	3	4	5	6	7	
	11	12	13	14	15	16	8
	21	22	23	24	25	26	
	31	32	33	34	35	36	9
	41	42	43	44	45	46	10
	51	52	53	54	55	56	11
	61	62	63	64	65	66	12

sum =	2	3	4	5	6	7	8	9	10	11	12
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Prob =	$\frac{1}{36}$	$\frac{2}{36}$	$\frac{3}{36}$	$\frac{4}{36}$	$\frac{5}{36}$	$\frac{6}{36}$	$\frac{5}{36}$	$\frac{4}{36}$	$\frac{3}{36}$	$\frac{2}{36}$	$\frac{1}{36}$

chapter 11: more lists

index operator `[]`

`L = ['one', 'two', 'three']`

`L[0] = 'one'`

the expression inside `[]` must be an integer

`L[0.0] = error`

Recall methods for traversing a list

```
• for i in [0, 1, 2]:
    print(L[i])
```

```
#output: one
#         two
#         three
```

or

```
• for i in range(len(L)):
    print(L[i])
```

or

```
• for x in L:
    print(x)
```

Recall in, not in operators

'one' in L # True

'happy' in L # False

'happy' not in L # True

- can concatenate lists (like strings) using +

$$A = [1, 2, 3]$$

$$B = [4, 5, 6]$$

$$C = A + B \quad \# [1, 2, 3, 4, 5, 6]$$

- the * operator repeats lists

$$D = 3 * A \quad \# [1, 2, 3, 1, 2, 3, 1, 2, 3]$$

$$E = A * 3 \quad \# \text{ same}$$

- slices

$$D[3:7] \quad \# [1, 2, 3, 1]$$

$$D[:] \quad \# \text{ copy of } D$$

- recall lists are mutable

$$D[5] = 7 \quad \# [1, 2, 3, 1, 2, 7, 1, 2, 3]$$

- can alter a list using slices

$$D[3:7] = [6, 7, 8, 9]$$

$$\# D \text{ is now } [1, 2, 3, \underline{6, 7, 8, 9}, 2, 3]$$

- can remove elements using slice

$$D[3:7] = []$$

$$\# \text{ now } D \text{ is } [1, 2, 3, 2, 3]$$

• can also use del operator

`del L[1] # ['one', 'three']`

`# E was [1, 2, 3, 1, 2, 3, 1, 2, 3]`

`del E[1:5]`

`# now E is: [1, 3, 2, 3]`

Objects

The is operator determines if two objects are one and the same.

Ex.

$x = 1$

$y = 1$

$x \text{ is } y$
 $x == y$ } both True

$a = \text{'happy'}$

$b = \text{'happy'}$

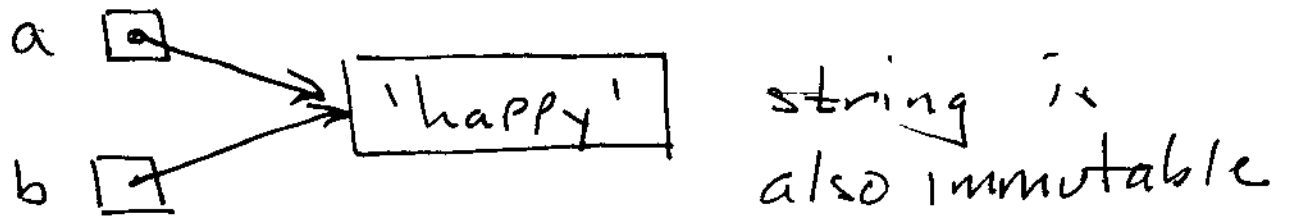
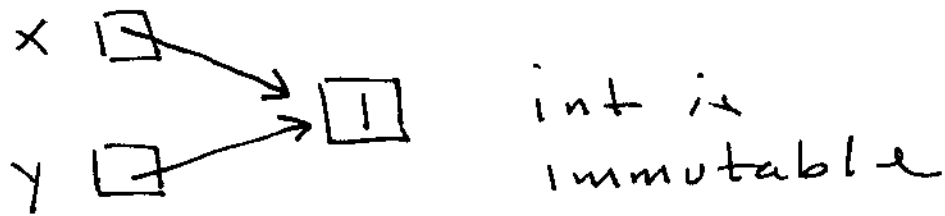
$a \text{ is } b$
 $a == b$ } both True

$A = [1, 2, 3]$
 $B = [1, 2, 3]$ } different list objects

$A \text{ is } B$ # False

$A == B$ # True

Pictures of memory



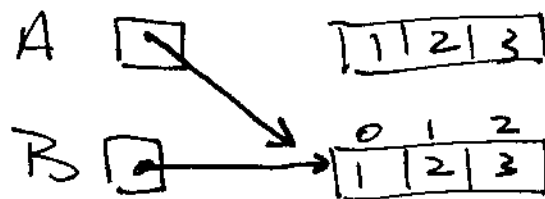
list is a mutable type



For mutable objects (like lists) the assignment operator = creates an alias.

A = B

Picture



A is B $\neq$ True now

So

is : True iff same object
in memory

== : True iff same data

To verify A, R now same object :

$R[0] = 5$ # changes $A[0]$

A # now $[5, 2, 3]$

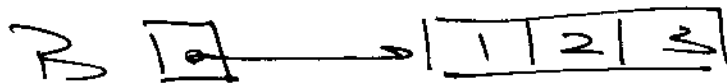
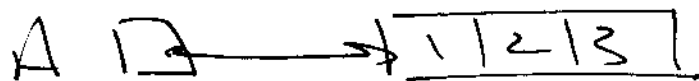
This aliasing by = operator only
happens for mutable objects.

To clone (or copy) a list,
use slice operator

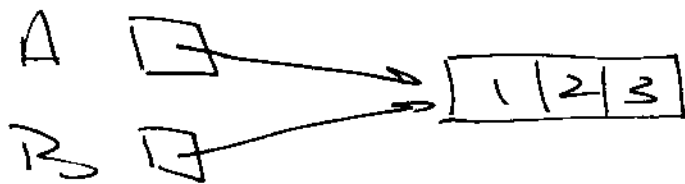
$$A = [1, 2, 3]$$

$$B = A[:]$$

now picture is



$A = B$ gives



|||

Note:

for any variables u and v

if $(u \text{ is } v)$ is True,

then $(u == v)$ is also True.

But converse may not be true,
i.e. we could have

$(u == v)$ True, but

$(u \text{ is } v)$ False