

Lecture 9, 4 September 2025

Madhavan Mukund

<https://www.cmi.ac.in/~madhavan>

Programming and Data Structures with Python

Building collections through iteration

targets = []

for match in matchlist:

targets.append(match[5])

$[m_1, m_2, \dots, m_n]$

↓ [5]

$[t_1, t_2, \dots, t_5]$

List of matches



List of numbers

$[v_1, v_2, \dots, v_n]$ $f:$



$[f(v_1), f(v_2), \dots, f(v_n)]$

Maps each v to $f(v)$

output = []

for v in l :

output.append($f(v)$)

output = map(f, l)

Filtered iteration

cskmatches = []

for match in matchlist:

if match[1] == 'CSK' or
match[2] == 'CSK':

cskmatches.append(match)

filter(p, e)

Condition p.

$[v_1, v_2, \dots, v_n]$

$\downarrow$
P(v₁)

$\downarrow$
P(v₂)

$\dots$ $\downarrow$
P(v_n)

T

F

T

$\downarrow$
 $[v_1$

X

$\downarrow$
 $v_n]$

Combine map & filter

Target when CSK wins toss

map (target, filter (csklosswin, matchlist))

↓
list

Squares of primes



$$\{x^2 \mid x \text{ is prime}\}$$

$$\{x^2 \mid x \in \mathbb{N}_0, \text{prime}(x)\}$$

Set comprehension

$$\{x^2 \mid \underline{x \in \mathbb{N}_0}, \text{prime}(x)\}$$

Build new sets from old sets

$$S = \{x \mid x \notin x\}$$

Russel's paradox

Is $S \in S$?

Set comprehension

Map + Filter

$$\{x^2 \mid x \in \mathbb{N}_0, \text{prime}(x)\}$$

↑
map

↑
iteration

↑
filter

List comprehension

$[m[5] \text{ for } m \text{ in matchlist if } m[3] == \text{'CSK'}]$

apply f iterate over matchlist check p

$[f(x) \text{ for } x \text{ in } l \text{ if } p(x)]$

Dictionaries

$\{k_1:v_1, k_2:v_2, \dots, k_n:v_n\}$

$\{k:v \text{ for } x \text{ in } l \text{ if } p(x)\}$

"generators"

Nested iteration

Multiple generators

$[(i,j) \text{ for } i \text{ in range}(3) \text{ for } j \text{ in range}(4)]$

$i=0 \rightarrow j=0,1,2,3$
for i --

$i=1 \rightarrow j=0,1,2,3$
for j ---

$[(0,0), (0,1), (0,2), (0,3), (1,0) \dots$
 $\dots (2,3)]$

Small Pythagorean tuples

x, y, z

$$x^2 + y^2 = z^2$$

[(x, y, z) for x in range(1, 100)
for y in range(1, 100)
for z in range(1, 100)
if $x*x + y*y = z*z$]

Mutable & immutable values

$x = 5$

$y = x$

$y = y + 3$

x	y
↓	↓
5	8

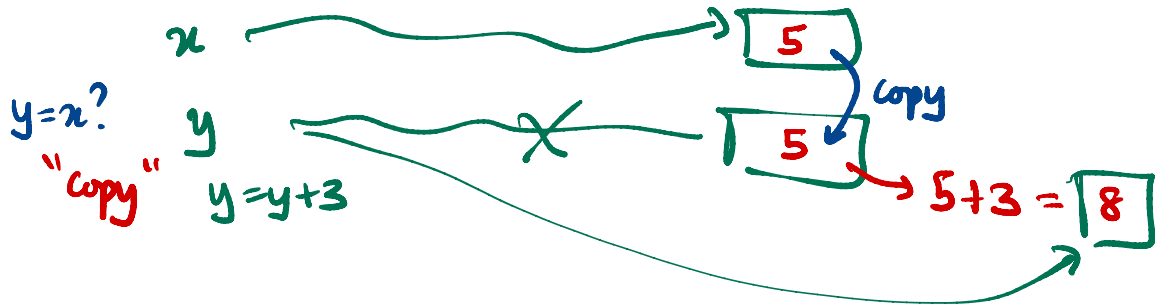
lists → $l1 = [3, 1, 2]$
are
mutable $l2 = l1$
 $l2.sort()$
 $l1.append(4)$

$l1$	$l2$
↓	↘
$[1, 2, 3, 4]$	

Immutable values : Numbers, Booleans, Tuples, Strings

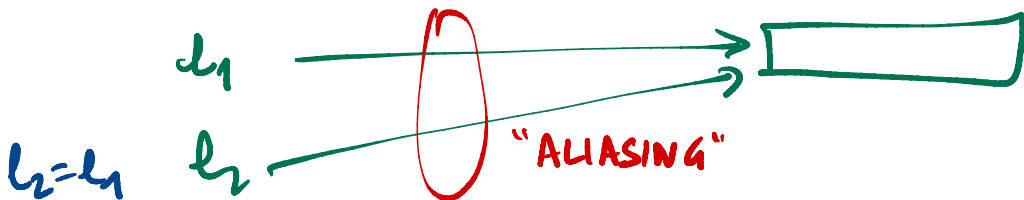
Name

Location



Mutable values

Lists, Dictionaries



"Copying" a list does not depend on
size of list

In place updates affect all aliased names

$l[p] = v$ in place

$l.append(v)$

$l.extend(l2)$

$l.sort()$

$l.reverse()$

$\vdots$

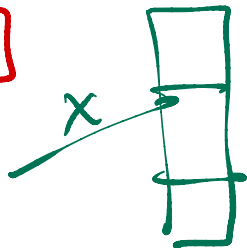
In place

Sorted (l)

new list

$l[i:j]$

New list



$l[:]$ $\rightarrow$ full slice

Copy $l1$ to $l2$

$l2 = l1[:]$

$l2 = [x \text{ for } x \text{ in } l1]$