

OOP Python 6/2

To write interesting programs we need:

- 1) lists, so we can store multiple things easily
- 2) functions, so we separate logically coherent tasks + not have to think about the whole program! (CODE REUSE)

Make a list

items = [] → empty list

for i in range(10): #

items.append(3*i) # add to end of list

0	1	2	3	4	5	6	7	8	9
0	3	6	9	12	15	18	21	24	27

-10 -9 -8 -7 -6 -5 -4 -3 -2 -1

for i in range(len(items)): → length of list

print(items[i])

for mine in items:

print(mine)

SLICE items[a:b] → list starting at index a ending at index b-1.

Lists of lists

$[[x_1, y_1, w_1, h_1], [x_2, y_2, w_2, h_2], \dots]$

`boxes = []`

```
for i in range(numBoxes):  
    # ENTER STUFF  
    box = [L, W, H]  
    boxes.append(box)
```

`boxes[i] → LIST`

`boxes = [ [2, 5, 3], [6, 1, 2], [4, 8, 12], [6, 6, 7] ]`

Diagram: A box around the last element `[6, 6, 7]` with the label `boxes[3]` above it. Three arrows point from the bottom of the box to the elements 6, 6, and 7.

`boxes[2] = [4, 8, 12]`

`boxes[0] = [2, 5, 3]`

`boxes[3][2] = 7`

`boxes[1][0] = 6`

Function Parameter Passing Lists

main

boxes $\rightarrow$ [[2, 5, 3], [6, 1, 4], [3, 4, 5], [2, 2, 2]]

fitsInside (boxes[2], boxes[3])

fitsInside (box1, box2)

box1

box2

box1[0] += 2

def add(a, b):

tot = a + b

a += 5

return tot

add

a [3] tot [12]
b [9]

def main():

x = 3

y = 9

z = add(x, y)

print(x, y, z)

x [3] 3, 9, 12

y [9]

z [12]