

Lecture 9, 8 September 2026

Madhavan Mukund

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

Programming and Data Structures with Python

Map & filter

map

$[x_0, \dots, x_{n-1}]$



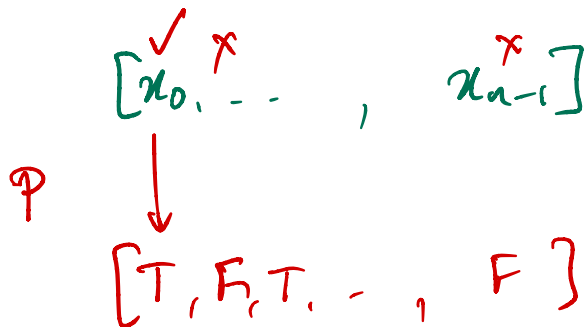
$[f(x_0), \dots, f(x_{n-1})]$

def f

map(f, l)

output sequence

⇒ convert to list



```
def square(x):  
    return x*x
```

```
l = [1, 2, 3, 4, 5, 6]
```

↓ ↓ ↓ ↓ ↓ ↓

```
def even(x):  
    return x%2==0
```

map(square, l)

← range(1,7)

↓

[1, 4, 9, 16, 25, 36]

map(square, filter(even, l))

↓

[2, 4, 6]

↓

[4, 16, 36]

$\{x^2 \mid x \in \mathbb{Z}, \text{ } x \text{ is even, } x > 0\}$ "Set Builder notation"
 map filter
 Given Set $\rightarrow$ New Set "Set Comprehension"

$\{x \mid x \notin x\} \rightsquigarrow Y$ Is $Y \in Y$?
 Russell's Paradox
 What is domain of x

List Comprehension

Assume `square()` & `even()` are defined

`[ square(x) | for x in range(1,7) if even(x) ]`

↓
list

`list(map(square, filter(even, range(1,7))))`

a sequence, convert to a list

Using list comprehension

Create $n \times n$ identity matrix

zeromatrix $n \times n$

$[0 \mid \text{for } i \text{ in range}(n)]$

$[[0 \mid \text{for } i \text{ in range}(n)] \mid \text{for } i \text{ in range}(n)]$
row of n zeros

```
def sety():  
    y = 5  


---

sety()  
print(y)
```



```
def isprime(n):  
    for i in range(2, n):  
        if n % i == 0:  
            return False  
    return True  
  
def primesupto(m):  
    primelist = []  
    for i in range(2, m+1):  
        if isprime(i):  
            primelist.append(i)
```



```
def sety():  
    y=5
```

← outer y
is "hidden"

```
y=0  
sety()  
print(y) → 0
```

When and where is a
name "visible"?



"scope"

"lifetime" – how long
is value
potentially
available

```
def sety()  
    y = 5
```

```
y = 0
```

```
sety()
```

```
print(y)
```

y Lifetime Scope y Lifetime Scope

✓ ✓

✗ ✗

✓ ✗

✓ ✓

✓ ✓

✗ ✗

```
def concat(l1, l2)
    l1.extend(l2)
```

$l1 =$

$l2 =$

$\text{concat}(l1, l2)$

↳ updated

```
def concat(l1, l2)
    l1 = l1 + l2
```

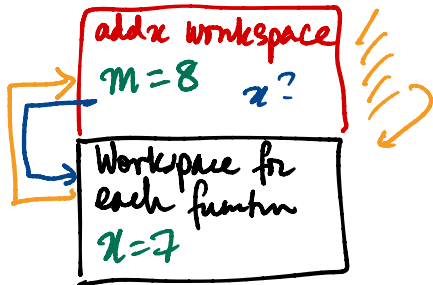
$l1$ is untouched

```
def addx(m)  
    print(x+m)
```

$x = 7$

$\text{addx}(8)$

How functions work



```
def sety():
```

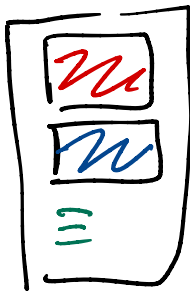
```
    y = x + 5
```

```
def setx():
```

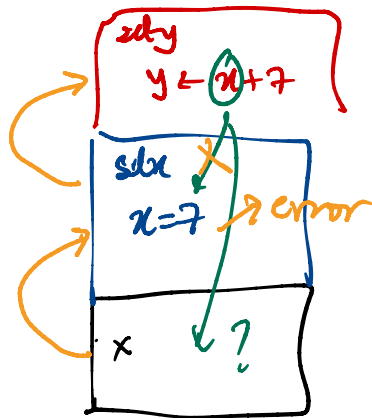
```
    x = 7
```

```
    sety()
```

```
setx()
```



"Static Scope"



```
def sety()  
    y = x + 5
```

```
def setx()  
    x = 7  
    sety()
```

```
x = 15  
setx()
```

