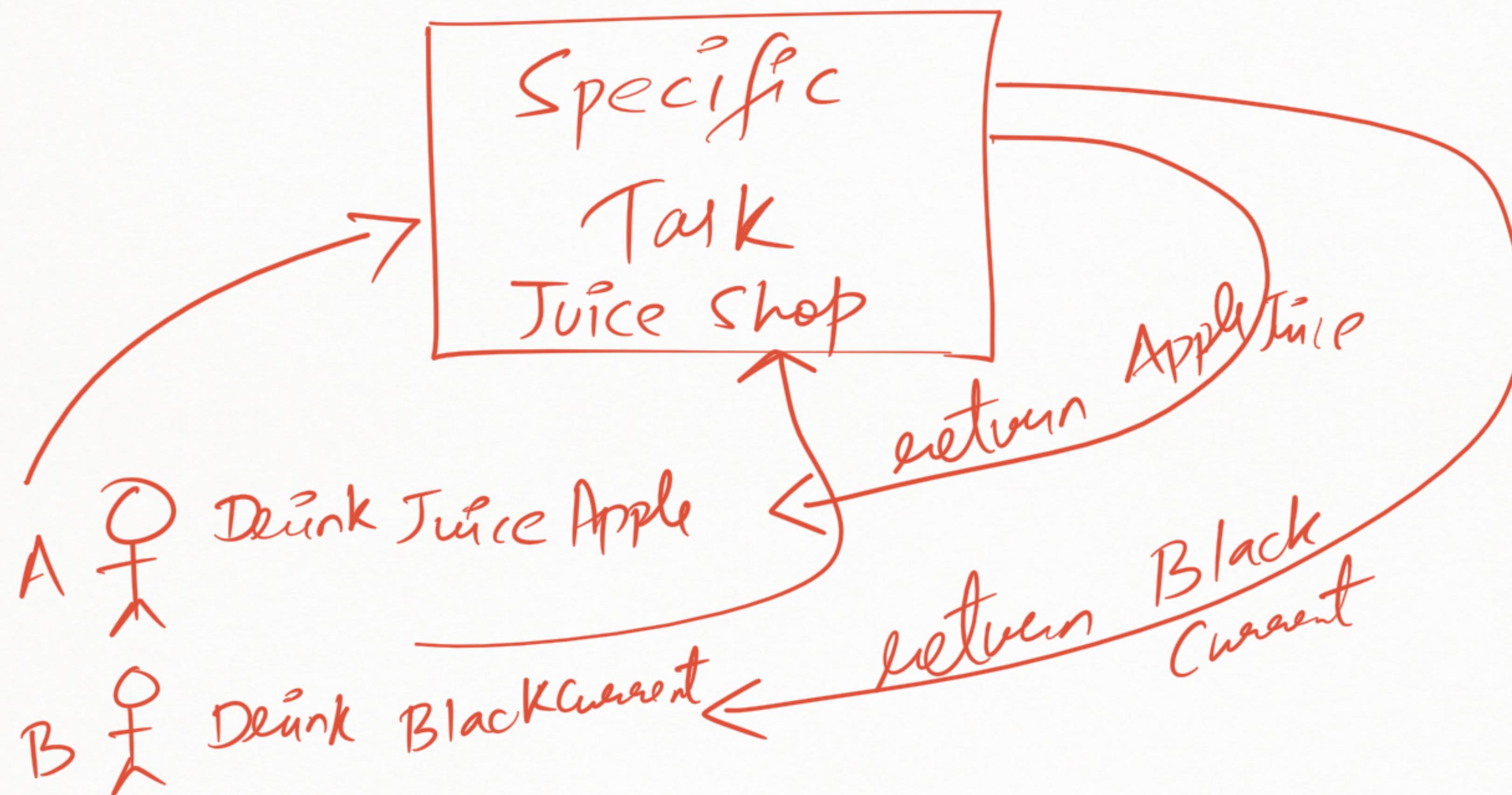
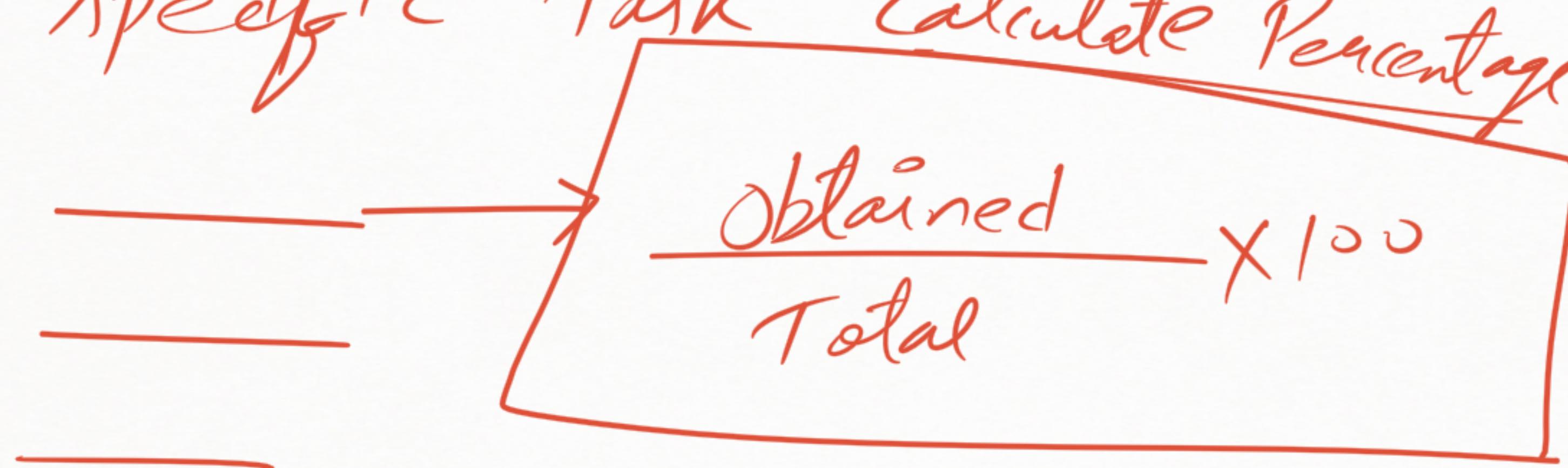
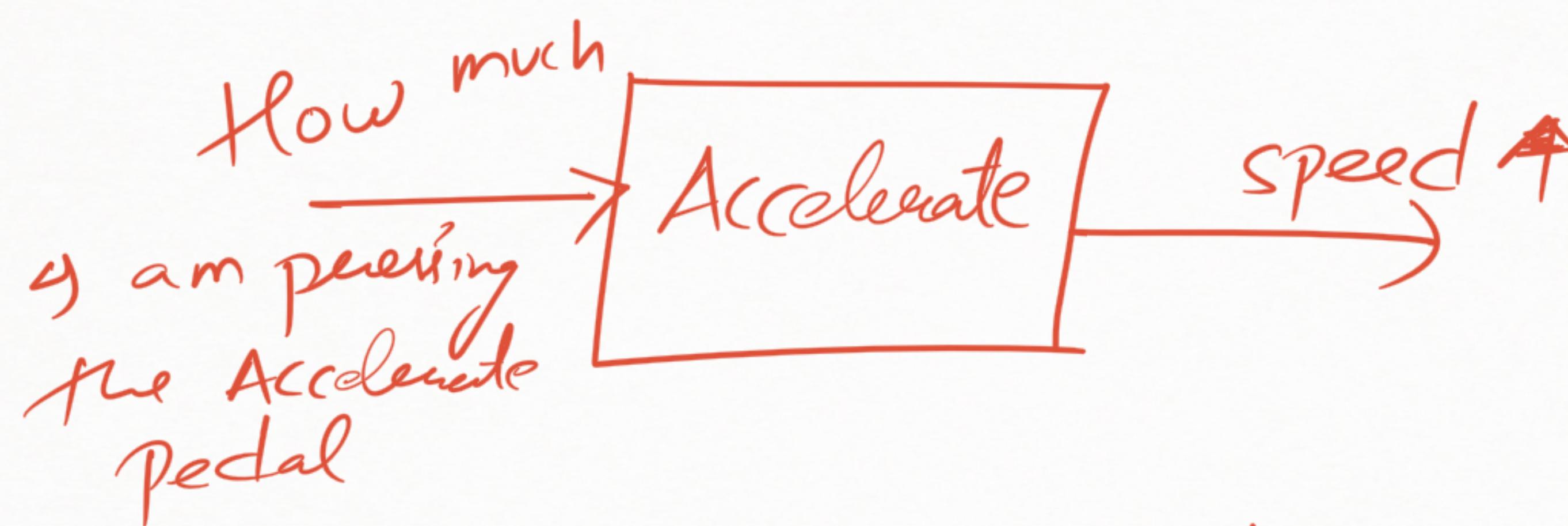


# What are functions



Functions are a block of code  
which is used for performing  
a specific Task





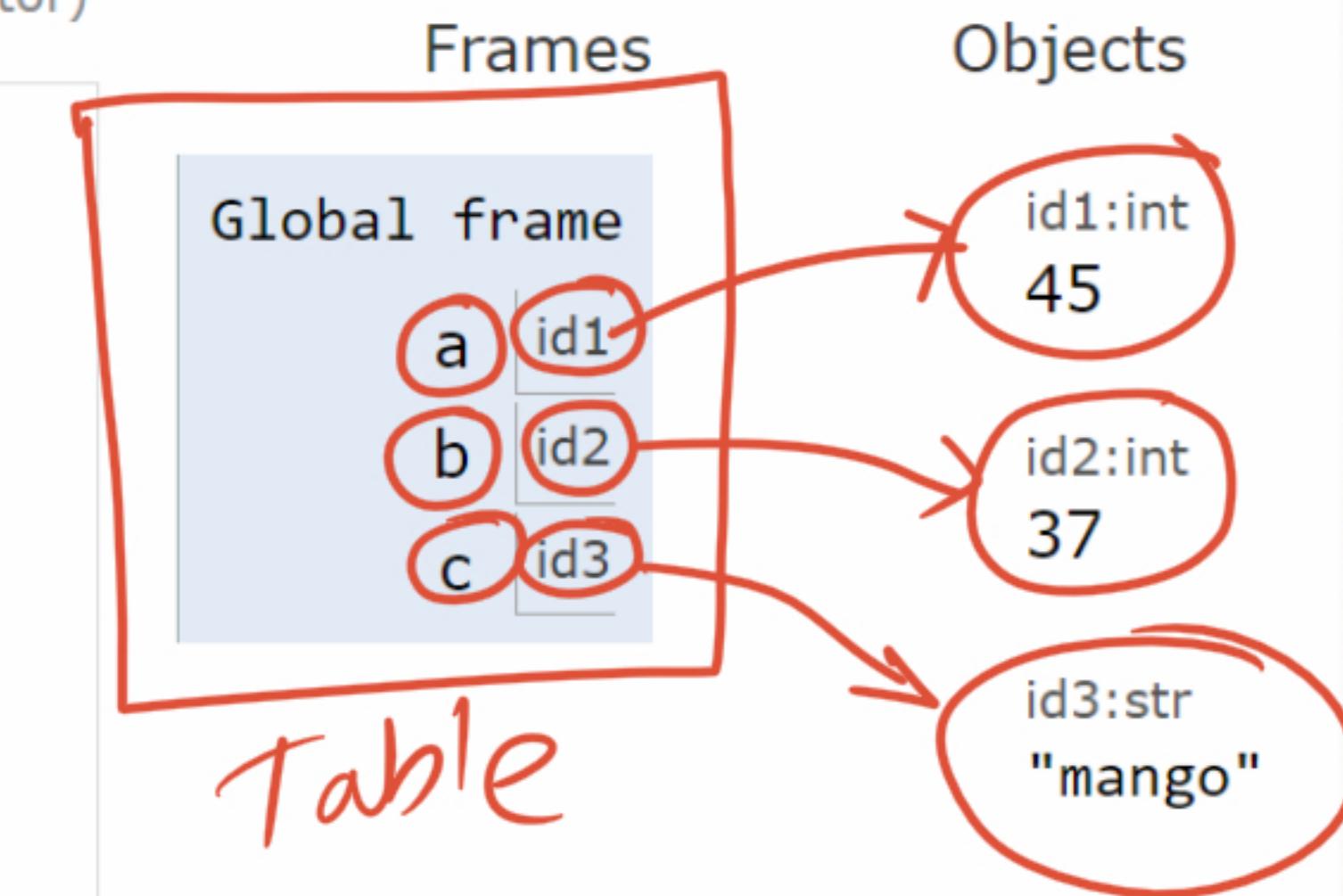
★ We will not be writing a similar code again. We will create a function and we will call it whenever need that kind of task.

code in Python 3.6

```
a = 45  
b = 37  
c = 'mango'
```

(drag lower right corner to resize code editor)

Global Namespace



The diagram illustrates the Global frame as a table. The table has three columns: Name, Address, and Namespace. The Name column contains variable names (a, b, c), the Address column contains object values (45, 37, "mango"), and the Namespace column contains the table itself. Red arrows point from each variable name to its corresponding object value.

Name	Address	Namespace
a	45	Table
b	37	Table
c	"mango"	Table

```

a = 45
def happy(x,y):
    print(x+y)

happy(3,4)
happy(3,400)

```

When I call  
happy again

i.e that just executed

Calling of happy with argument 3 and 4

LOCAL  
NAMESPACE

Local  
Namespace

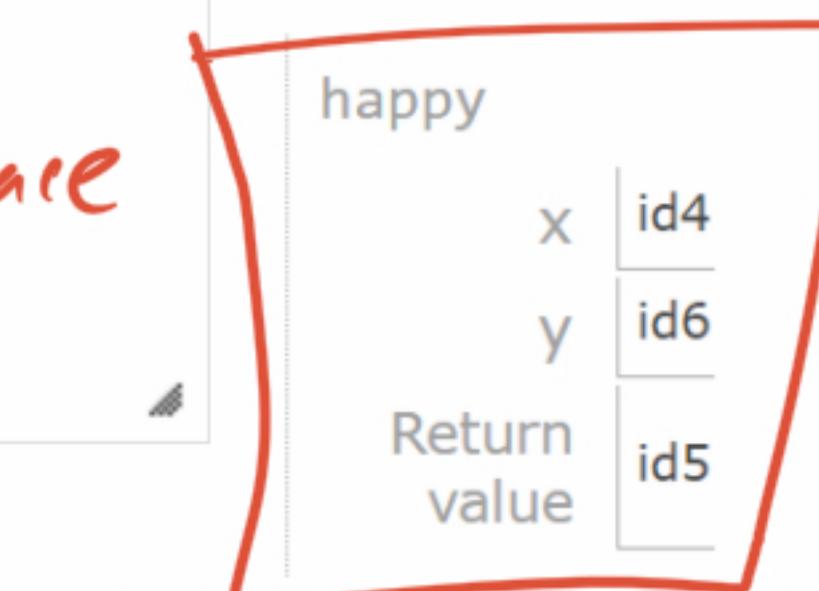
403

Frames

Global frame
a id1
happy id2

Objects

id1:int
45
id2:function
happy(x, y)
id4:int
3
id3:int
4
id5:NoneType
None
id6:int
400



code in Python 3.6

(drag lower right corner to resize code editor)

```
→ a = 45  
def happy(x,y):  
    local  
    b = 'mango'  
    print(x+y)
```

```
happy(3,4)
```

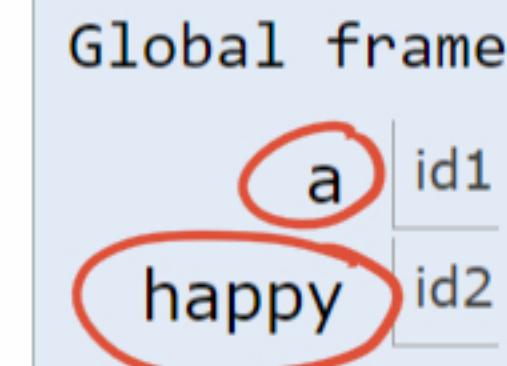
Whatever you write  
inside a particular  
function is considered  
as local variable.

Local Var

Time output (drag lower right corner to resize)

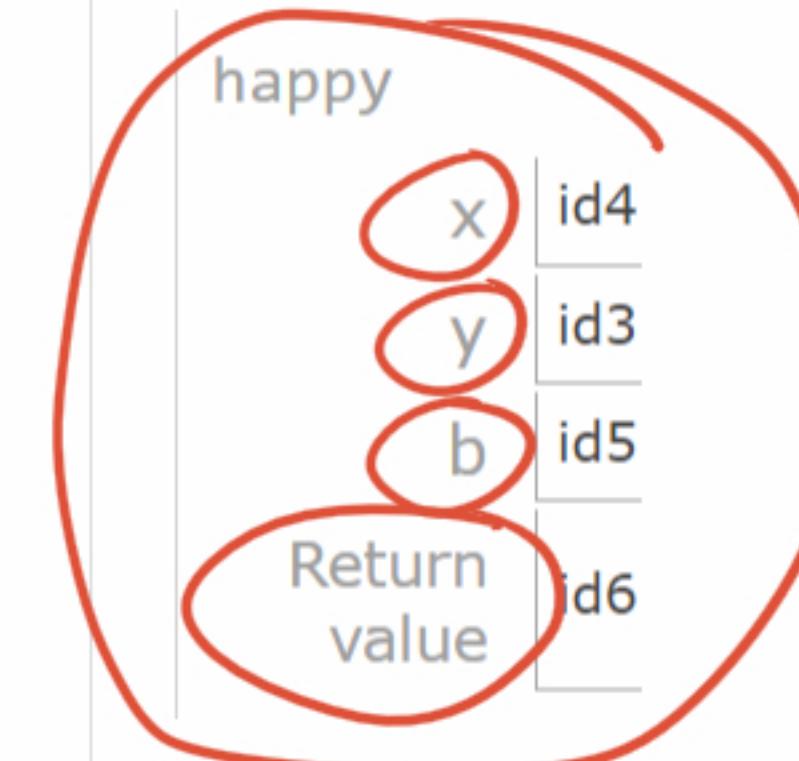
7

Frames



Objects

id1:int	45
id2:function	happy(x, y)
id4:int	3
id3:int	4
id5:str	"mango"
id6:NoneType	None



In [ ]:

```
x = 'mango' ✓  
y = 'kiwi' ✓  
z = [x,y] ✓  
  
def glad():  
    p = 'ice cream'  
    q = 'sweet corn'  
    r = 'lassi'  
    s = 'tea'|
```



Frames

Global frame
x id1
y id2
z id3
glad id4

glad

p id5
q id6
r id7
s id8
Return value id9

Objects

id1:str "mango"
id2:str "kiwi"
id3:list
0 id1   1 id2
id4:function glad()

id5:str "ice cream"
id6:str "sweet corn"
id7:str "lassi"
id8:str "tea"
id9:NoneType None

code in Python 3.6 (drag lower right corner to resize code editor)

```
def find_percentage(obtained_marks, total_marks):
    percentage = (obtained_marks/total_marks)*100
    if percentage <= 70:
        percentage += 5
    return percentage

find_percentage(40,200)
```

local  
Variables  
of  
find-percentage



```

✓ students = [{"name":'priyal jain','marks':[44,50,79,89]},  

    {'name':'krishnakant dubey','marks':[45,53,75,99]},  

    {'name':'Siddharth Kaithwas','marks':[55,67,72,56]}]  

]  

def find_percentage(obtained_marks, total_marks):  

    percentage = (obtained_marks/total_marks)*100  

    if percentage <= 70:  

        percentage += 5  

    return percentage  

def find_grade(percentage):  

    if percentage>=90:  

        return 'A'  

    elif percentage>=80:  

        return 'B'  

    elif percentage>=70:  

        return 'C'  

    elif percentage>=60:  

        return 'D'  

for student in students:  

    obtained_marks = sum(student['marks'])  

    percentage_of_student = find_percentage(obtained_marks, 400)  

    print(student['name'], 'got', find_grade(percentage_of_student), 'grade')

```

Local Namespace

local Namespace

Global frame	
✓ students	id1
✓ find_percentage	id25
✓ find_grade	id26
student	id18
obtained_marks	id35
percentage_of_student	id37

find_percentage	
✓ obtained_marks	id27
✓ total_marks	id28
✓ percentage	id30
Return value	id30

find_grade	
percentage	id30
Return value	id31

If you try to search for a name



LOCAL NAMESPACE

| if not found

GLOBAL NAMESPACE

| if not found.

BUILTIN NAMESPACE

```
|: a = 'mango'  
  
def happy():  
    a = 'kiwi'  
    print(a)  
  
happy()
```

kiwi

```
|: a = 'mango'  
  
def happy():  
    print(a)  
  
happy()
```

mango

When I ask Python what is  
the value of a

LOCAL NAMESPACE

When I ask for a

LOCAL NAMESPACE

Global Namespace