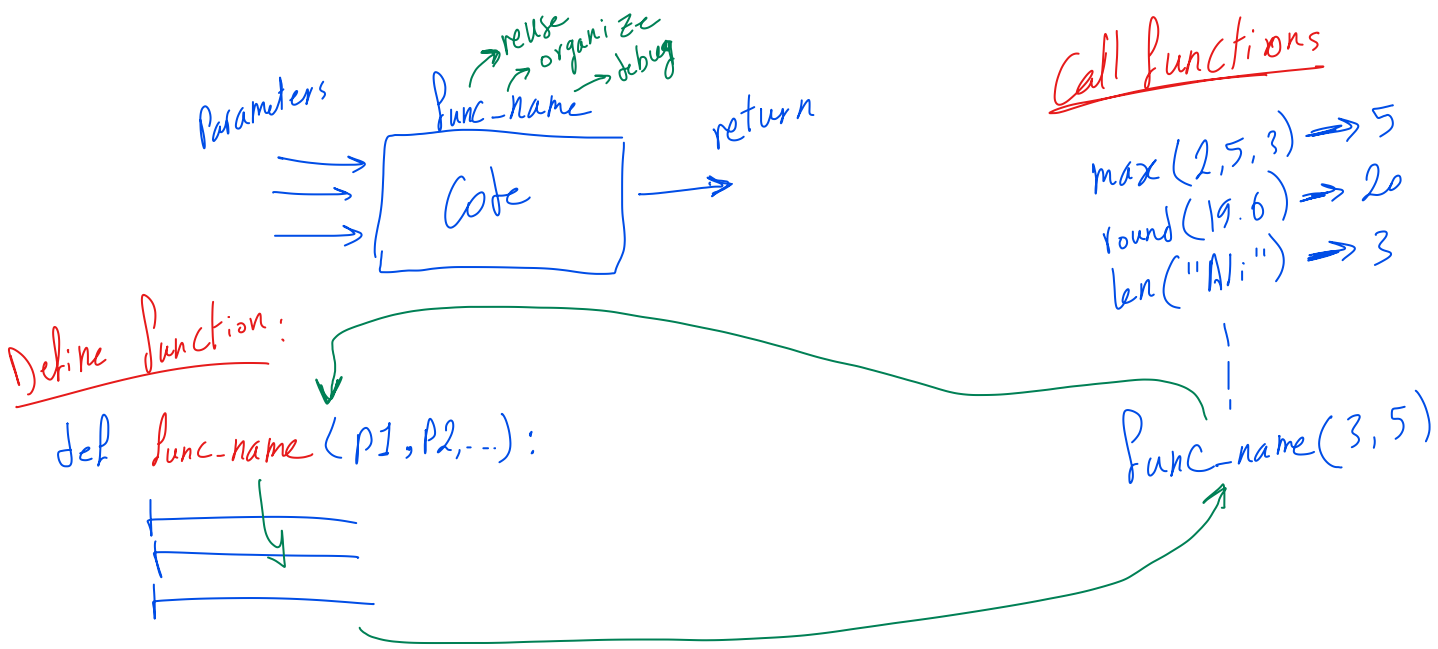
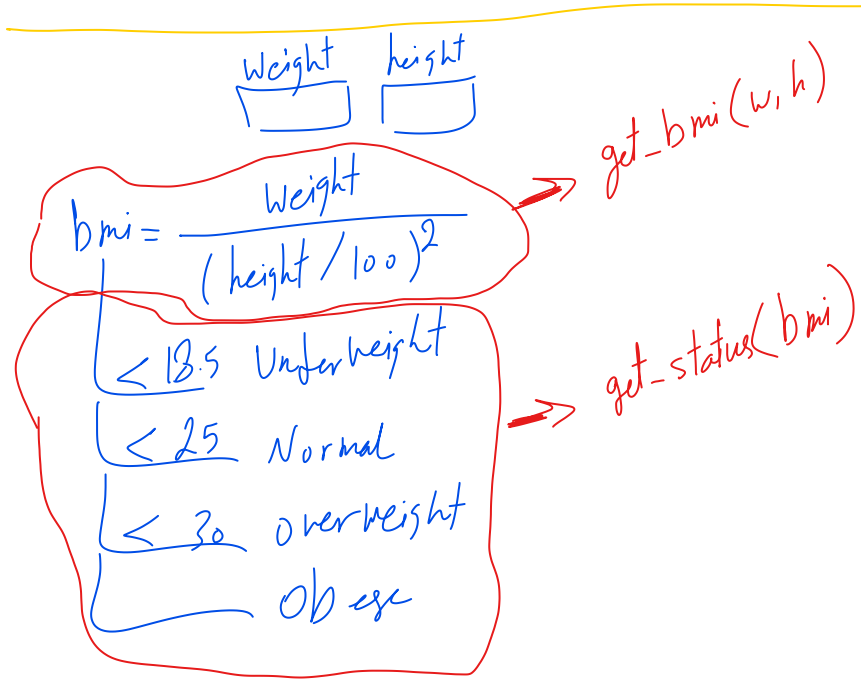
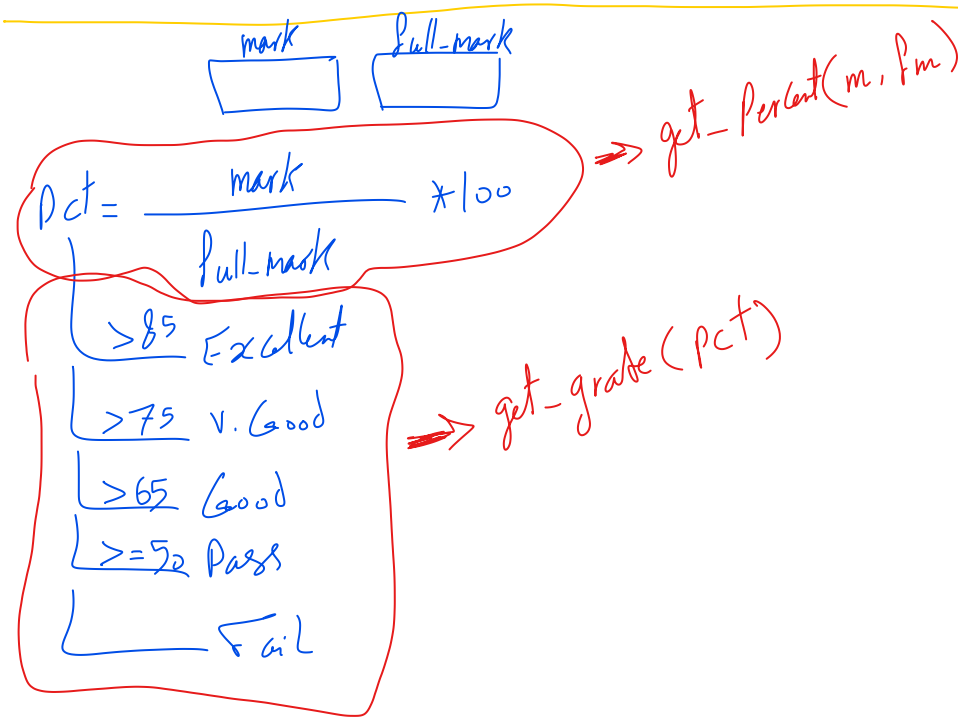
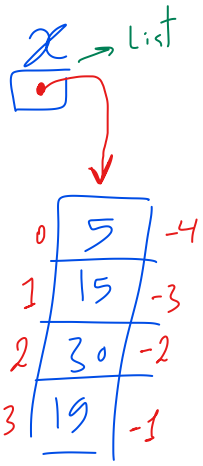

reuse
organize
debug
Parameters
Func_name
Code
return
Call functions
max(2,5,3) → 5
round(19.6) → 20
len("Ali") → 3
Define function:
def Func_name(p1, p2, ...):
Func_name(3, 5)


weight
height
bmi = weight / (height/100)^2
get_bmi(w, h)
< 18.5 Underweight
< 25 Normal
< 30 overweight
Obese
get_status(bmi)


mark
full_mark
pct = mark / full_mark * 100
get_Percent(m, fm)
>85 Excellent
>75 V. Good
>65 Good
>=50 Pass
Fail
get_grade(pct)

x = [5, 15, 30, 19]

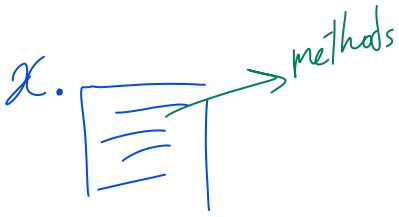


Print(x) ⇒ 5, 15, 30, 19

Print(x[0]) ⇒ 5

Print(x[-1]) ⇒ 19

Print(len(x)) ⇒ 4



Enter num of students: 5

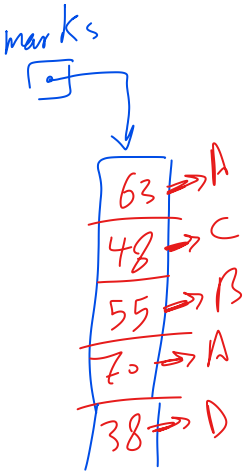
Enter std. mark: 63

~ ~ ~ : 48

~ ~ ~ : 55

~ ~ ~ : 70

~ ~ ~ : 38



A ⇒ m >= best - 10 (best = 60)

B ⇒ m >= best - 20 (best = 50)

C ⇒ m >= best - 30 (best = 40)

D ⇒ m >= best - 40 (best = 30)

else ⇒ else

best = max(...)

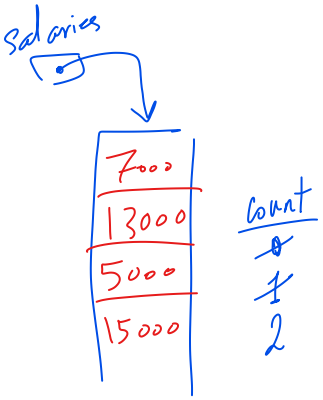
Enter num of emp: 4

Enter emp salary: 7000

~ ~ ~ : 13000

~ ~ ~ : 5000

~ ~ ~ : 15000



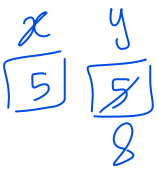
$$avg = \frac{sum(-)}{len(-)}$$ → 10000

---

x = 5

y = x

y += 3



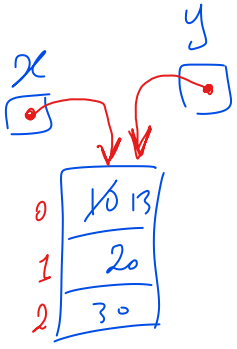
copy value

---

x = [10, 20, 30]

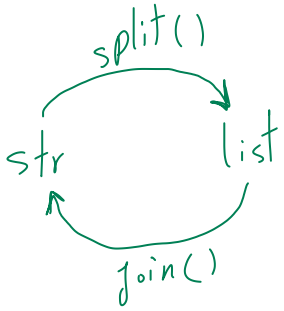
y = x

y[0] += 3



copy Reference

```
dob = "22/6/2002"
dob = dob.split("/")
dob = "-".join(dob)
```



dob

22-6-2002

dob

0	22
1	6
2	2002

Tuple (immutable)

```
x = (23, 13, 19, 25)
print(x[0]) => 23
print(len(x)) => 4
x[0] += 2   X
x.append(-) X
```

Set → no duplication, → not ordered

```
x = {23, 13, 19, 13, 25}
print(x) → {23, 13, 19, 25}
print(len(x)) → 4
print(x[0]) X
```

x

0	Saudi Arabia
1	Arab Emirates
2	Egypt
⋮	

```
print(x[0])
print(y["sa"])
```

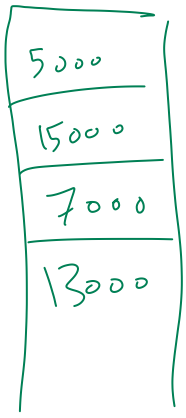
y — Dict

966 sa	Saudi Arabia
971 ae	Arab Emirates
20 eg	Egypt

63
48
70
55
38

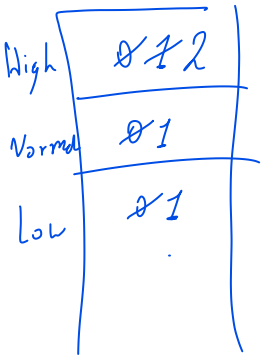
A → n >= best - 10
B → n >= best - 20
C → n >= best - 30
D → n >= best - 40
F → else

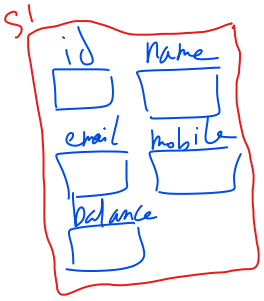
A	~~0~~ 1 2
B	~~0~~ 1
C	~~0~~ 1
D	~~0~~ 1
F	0

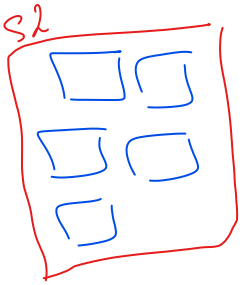


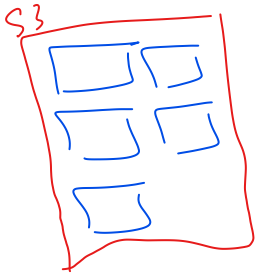
s >= 12000

s >= 6000









Student

Trainer

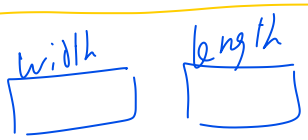
Course

OOP

object => place in memory / special variable

class => code

- How to create object?
- Define operations on object



area = width * length

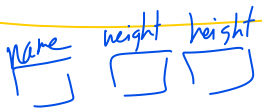
OOP

Rectangle

width
length

__init__(-,-)

get_area()



$$bmi = \frac{weight}{(height/100)^2}$$

- < 18.5 Underweight
- < 25 Normal
- < 30 Overweight
- Obese

OOP

Patient

name
weight
height

__init__(-,-,-)

get_bmi()

get_status()