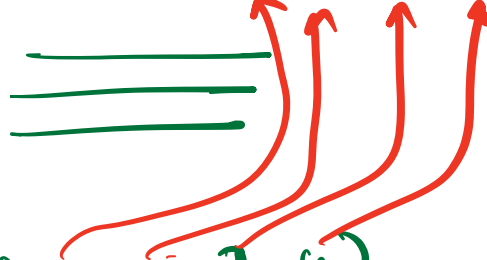


"Hello Everyone!"

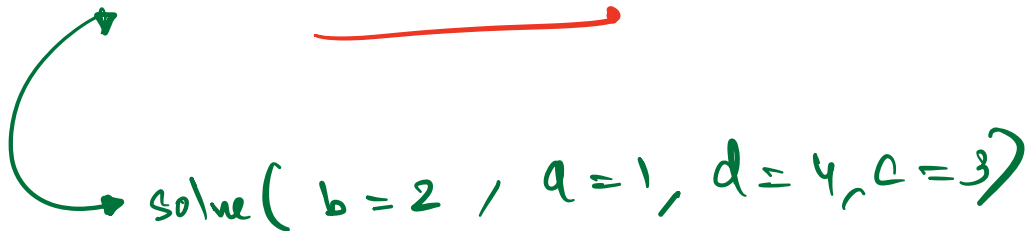
① positional arguments

```
def solve(a, b, c, d):
```



Three horizontal green lines representing a function call are shown. Four red arrows point from the arguments '1', '2', '3', and '4' in the call below to the parameters 'a', 'b', 'c', and 'd' in the function definition above, respectively.

```
solve(1, 2, 3, 4)
```



A red arrow points from the underlined '1' in the function call 'solve(1, 2, 3, 4)' to the parameter 'b' in the function call 'solve(b=2, a=1, d=4, c=3)' below. This indicates that the first positional argument '1' is being mapped to the keyword argument 'b'.

```
solve(b=2, a=1, d=4, c=3)
```

↙ key-worded arguments

★ Mix func

Rule

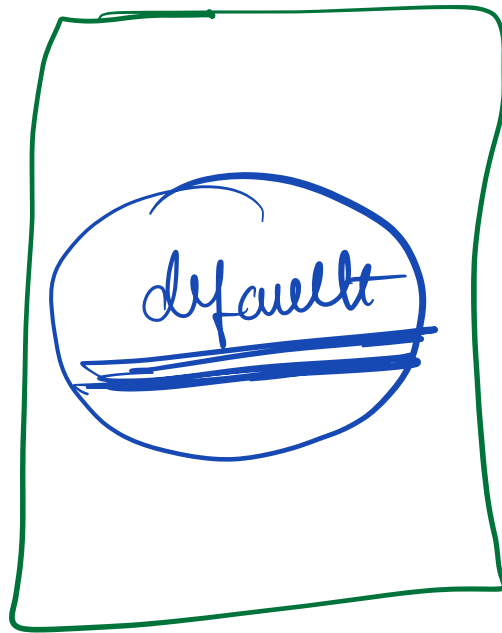
To have all the
keyworded arguments

AFTER

all the positional arguments

positional

keyworded



Rule for default arg

→ in definition

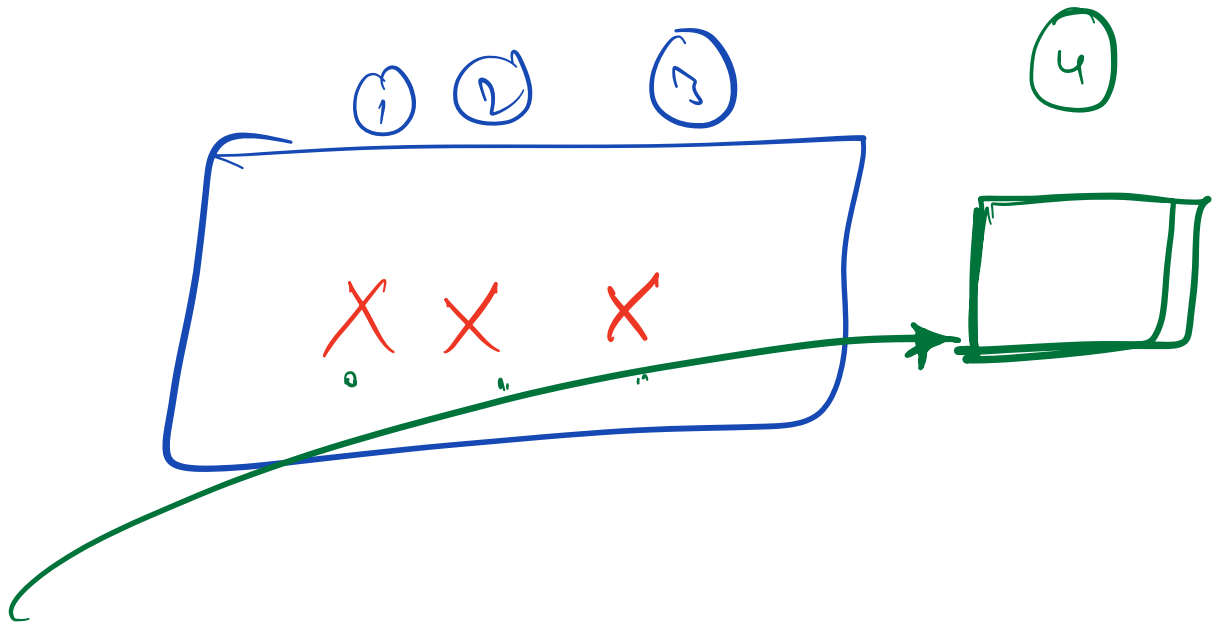
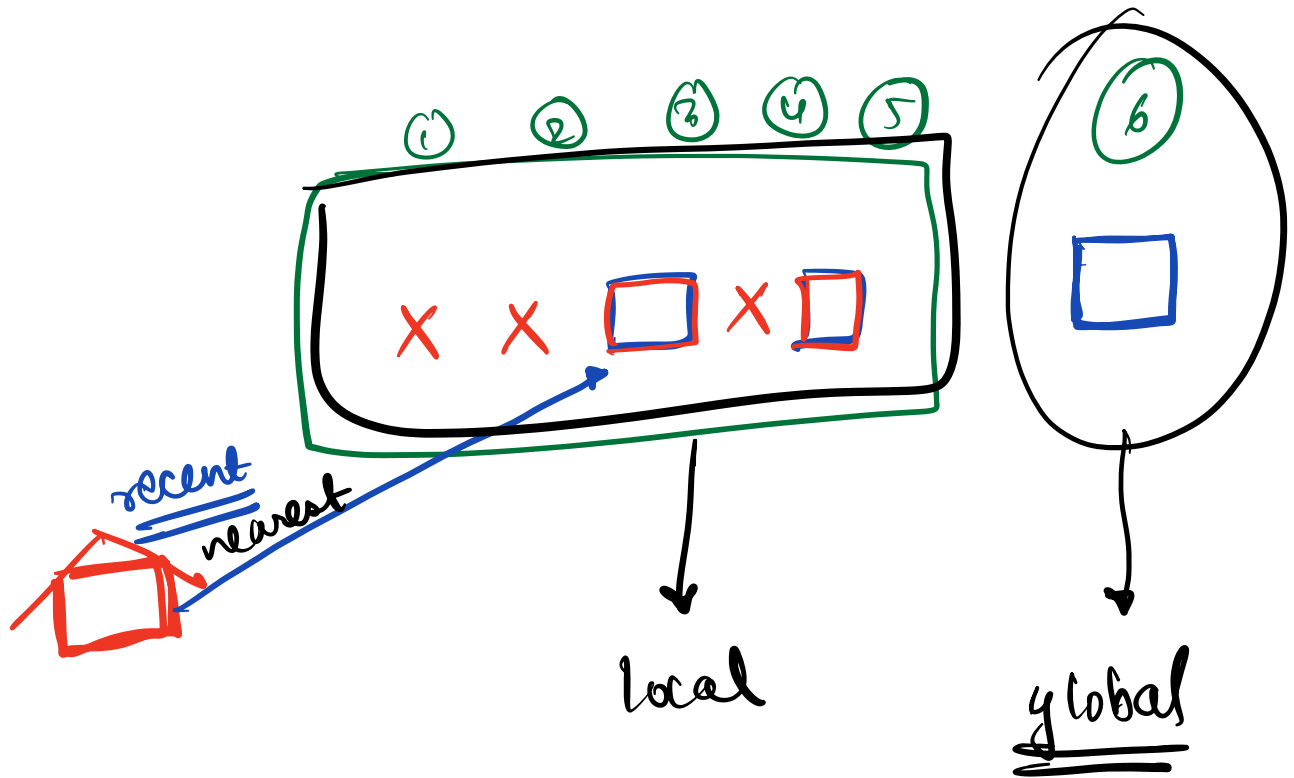
all default arg

AFTER

all positional arg.

10:15 sharp

* Local vs Global variable



within your method

local
variable

outside "

global variable

Cannot be accessed
outside

a = 10 → global

def random(n):

a = 5 → local

print(a) → 5

print(n) → 7

→ random(~~7~~)

print(a) → 10