

21 Tuesday
17 Jamadi-ul-Awwal

"Understanding self"

parameter

Consider class :

```
class Employee():
```

define method without self

```
def details():
```

pass

do nothing

make object & access method

```
e1 = Employee()
```

```
e1.details()
```

→ you will surely get error

17 Friday
13 Jamadi-ul-Awwal

19 20 21 22 23 24 25
26 27 28 29 30 31

Error: details() take 0 positional arguments but 1 was given.

→ To understand this we have to understand how python handles "e1.details()" in background.

So when we write:
e1.details()
What python do is:

Employee.details(e1)

18 Saturday 14 Jamadi-ul-Awwal

class name

method

obj e1

19 Sunday 15 Jamadi-ul-Awwal

we can check that too

→ so to receive object e1 we will use self parameter.

→ It can be something else but convention is self.

5 Wednesday
11 Jamadi-ul-Awwal

12	13	14	15	16	17	18	19
19	20	21	22	23	24	25	26
26	27	28	29	30	31		

class Employee():

with self parameter

def details(self):

making attribute with self

self.name = "Usman"

making without self
age = 30

↪ life span of this attribute will be just in this method when method running age can not be used in any other method.

6 Thursday
12 Jamadi-ul-Awwal

↪ on the other hand we can use self.name in other methods.