

في الدرس دا هنتكلم عن ال **Property decorator**.

لو انا عندي فانكشن اسمها say_hello() مبتاخدش اي parameters ومجرد بس بتعملي وظيفه ثابتة، ولما جيت اطلعها نسيت احط الاقواس كدا print(say_hello)، كدا هيطبعلي الفانكشن دي نوعها ايه وبتنتمي ل class ايه وال location بتاعها في ال memory ومش هيطبعلي خرج الفانكشن.

علشان اقدر اطلعها من غير الاقواس بحولها ل property function يعني بتعامل معاملة ال Attribute، بعمل دا عن طريق اني بضيف قبلها decorator اسمه **@property**، وكدا اقدر استدعيها من غير الاقواس كأنها Attribute وهتديني الخرج بتاعها. لكن لاحظ اني لو حطيت قبلها ال decorator دا وجيت استدعيها وحطيت الاقواس هيديني error لانه مش شايفها فانكشن دلوقتي هو شايفها property (Attribute).

```
# -----
# -- Object Oriented Programming => @Property Decorator --
# -----

class Member:

    def __init__(self, name, age):

        self.name = name

        self.age = age

    def say_hello(self):

        return f"Hello {self.name}"

    @property
    def age_in_days(self):

        return self.age * 365

one = Member("Ahmed", 40)

print(one.name)
print(one.age)
print(one.say_hello())
# print(one.age_in_days())

print(one.age_in_days)
```

CODE

```
Ahmed
40
Hello Ahmed
14600
```

OUTPUT