

الدرس دا هنتكلم فيه عن ال **Generators**، هي زى الفانكشنز بالظبط لكن فى الفانكشنز كان عندنا قيمة return واحدة اما فى ال generators عندنا اكر من return وفى الحاله دى بيطلق عليها اسم **yield** مش return.

ال generator بيكون عبارته عن iterator وعلشان ارجع قيمة اول yield بستخدم next() وعلشان اجيب قيمة ال yield اللى بعده بستخدم next() تاني وهكذا، واقدر كمان انى اعمل عليه iteration بال for loop زى ال iterator العادي بالظبط.

لما بعمل call لل generator (استدعيه) مش بينفذ حاجه لأن قيم اللى بيرجعها ال yield اللى جواه بترجع بس باستخدام ال iteration ب next().

لاحظ ان كل next() بتطبع العنصر اللى عليه الدور فى الخروج، علشان كذا فى اول امرين طباعه طبع 1 و 2 ولما كملت بال for طبع من اول م كان واقف اللى هو 3 و 4.

ميزة انى اطبع عنصر عنصر باستخدام ال next() هو انى ممكن احط انفذ كود بين كل امر والتاني على عكس ال for بترجع كل العناصر ورا بعضها.

```
# -----
# -- Generators --
# -----
# [1] Generator is a Function With "yield" Keyword Instead of "return"
# [2] It Support Iteration and Return Generator Iterator By Calling "yield"
# [3] Generator Function Can Have one or More "yield"
# [4] By Using next() It Resume From Where It Called "yield" Not From Beginning
# [5] When Called, Its Not Start Automatically, Its Only Give You The Control
# -----

def myGenerator():
    yield 1
    yield 2
    yield 3
    yield 4

myGen = myGenerator()

print(next(myGen), end=" ")
print("Hello From Python")
print(next(myGen), end=" ")

for number in myGen:
    print(number)
```

CODE

```
ocuments/Python Course/first.py"
1 Hello From Python
2 3
4
```

OUTPUT