

Day_8_Generator_Function(DATA_MINDS)

September 11, 2023

GENERATOR FUNCTIONS - IT WILL GENERATE ONE BY ONE AND GIVES THE DATA TO THE USER , WE USE “YIELD” KEYWORD TO MAKE GENERATOR FUNCTION

IT GENERATES THE OUTPUT CONTINUOUSLY WITHOUT HOLDING INTO A MEMEORY

```
[1]: range(10)
```

```
[1]: range(0, 10)
```

```
[2]: for i in range(10):  
      print(i)
```

```
0  
1  
2  
3  
4  
5  
6  
7  
8  
9
```

GENERATOR - WE CREATE GENERATOR FUNCTION WITH THE HELP OF “YIELD” KEYWORD

FIBONACCI NO - IT IS A KIND OF NO THAT ADD PREVIOUS TWO NUMBERS AS A OUTPUT

```
[5]: def test_fib(n):  
      a,b=0,1  
      for i in range(n):  
          yield a  
          a,b=b,a+b
```

```
[6]: test_fib(10)
```

```
[6]: <generator object test_fib at 0x7fc32392d930>
```

```
[9]: for i in test_fib(10):  
      print(i)
```

```
0  
1  
1  
2  
3  
5  
8  
13  
21  
34
```

```
[12]: def test_fib1():  
      a,b=0,1  
      while True:  
          yield a  
          a,b=b,a+b
```

```
[13]: fib=test_fib1()
```

```
[14]: for i in range(10):  
      print(next(fib))
```

```
0  
1  
1  
2  
3  
5  
8  
13  
21  
34
```

```
[15]: s="Virat"
```

```
[17]: for i in s :  
      print(i)
```

```
V  
i  
r  
a  
t
```

```
[18]: s
```

```
[18]: 'Virat'
```

```
[19]: next(s)
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[19], line 1  
----> 1 next(s)  
  
TypeError: 'str' object is not an iterator
```

```
[20]: v=iter(s)
```

```
[21]: next(v)
```

```
[21]: 'V'
```

```
[22]: next(v)
```

```
[22]: 'i'
```

```
[23]: next(v)
```

```
[23]: 'r'
```

```
[24]: next(v)
```

```
[24]: 'a'
```

```
[25]: next(v)
```

```
[25]: 't'
```

```
[27]: next(45)
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[27], line 1  
----> 1 next(45)  
  
TypeError: 'int' object is not an iterator
```

```
[28]: iter(25)
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[28], line 1  
----> 1 iter(25)  
  
TypeError: 'int' object is not iterable
```

NOTE - ITERATOR IS A KIND OF DATA IN WHICH WE MOVE NEXT NEXT AND NEXT AND FETCH THE DATA AS WELL

NOTE - STRING IS A ITERATOR BUT INT IS NOT ITERATOR

```
[3]: def count_test(n):  
      count=1  
      while count<n:  
          yield count  
          count=count+1
```

```
[4]: c=count_test(5)
```

```
[5]: for i in c:  
      print(i)
```

```
1  
2  
3  
4
```