17/9/23 Task 8.1:- Implement Python generator and decorators

AIM :- write a python program to implement python generator and decorator.

ALGORITHM :-

1. Define the function number-sequence.
2. Initialize Current value.
3. Generate Sequence.
4. Get user Input (start, end, step)
5. Create Generator object
6. Print Generated Sequence
7. Stop.

PROGRAM :-

```
    def number_sequence (start, end, step=1):
        Current = Start
      while current <= end :
          Yield Current
          Current += Step
Start = int (input ("Enter the starting number:"))
end = int (input ("Enter the endling number:"))
Step = int (input ("Enter the step value:"))
Sequence_generator = number_sequence (Start, end, step)
 for number in Sequence_generator:
   Print (number)
```

Output :

```
Enter the Starting number : 1
Enter the ending number : 50
Enter the step value : 5
 1
 6
 11
 16
 21
 26
 31
 36
 41
 46
```

17/9/25 Task 8.2 Implement Python Generator and decorator

AIM : To write a program to convert the text to uppercase or lowercase ..

ALGORITHM :

(i) Create Decorators :

- Define uppercase-decorator to convert the result of a function to uppercase.

(ii) Define Functions:

- Define shout function to return the input text. Apply @ uppercase-decorator to this function.

(iii) Define Greet Functions.

(iv) Execute the program.

- Call greet to print in uppercase.
- Call greet(whisper) to print the greetings in lowercase.

PROGRAM :

```
def uppercase-decorator (func):
  def wrapper (text):
      return func(text). upper()
  return wrapper

def lowercase_decorator (func):
    def wrapper (text):
        return func(text).lower()
    return wrapper
@ uppercase_decorator
def shout (text):
    return text

@ lower case decorator
def whisper (text):
    return text

def greet (func):
        greeting = func ("Hi, I am created by a function passed
                                as an argument.")
        print (greeting)
```

Output :-

```
Hi , I AM CREATED BY A FUNCTION PASSED AS
 AN ARGUMENT
hi, i am created by a function passed as an argument
```

```
greet (shout)
greet (whisper)
```

VEL TECH - CSE	
EX NO.	8
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	4
TOTAL (15)	15
SIGN WITH DATE	

RESULT :- Thus the python program to implement python generator and decorator was successfully executed.