Task 8: Implement Python generator and delorators

AIM: write a Python Program to implement Python generator and decorators.

8.1 write a Python Program that includes a generator function to Produce a sequence of numbers. The generator should be able to:

a) Produces a sequence of numbers when Provided with Start lend, and Step values

b) freduce a default sequence of numbers starting from 0, ending at 10, and with a step of 1 if no values are Provided.

Produce a sequence of numbers when Provided with start, end and step values.

Algorithm:

- 1) Define Generator function:
- · Define the function number Sequence (Start, end, Sterg =1)

 a) Intialize (urrent value:
- . Set current to the value of Start.
- 3) benerate Sequence:
- · while current is less than or equal to end.
 - · Yield the current value of current
- · increment (urrent by step.
- w) het user input:
- · Read the starting number (start) from user input
- · Read the ending number (end) from user input · Read the Step value (step) from the user input
- 5) Create henerator object;
- · Create generator object by coulting number-sequence (stort, and, stell) with two Provided Values.
- 6) Print Generated Sequence:
- · Iterate over the value Produced by the generator offect
- · Print each value.

```
8.1 Program:
 def number_sequence (start, end, step=1):
 Current = Start
 while current <= end:
 Yield current
  arrent+=step
 Start = int (influt ("Enter the Starting number:"))
 end = int (input (" Enter the ending number:"))
   SteP = int (input (" Enter the SteP value:"))
 # Create the gonerator
   sequence _generator = number_ sequence (start, end, stel)
  # Print the generated sequence of numbers
 for number in sequence —generator.
     Mint (number)
 Produce a default sequence of numbers starting from a ending at
 10, and with a step of 1 if no values are provided.
 Algorithm:
 1) start function:
· Define the function my - generator(n) that takes a Parameter 17.
2) Intialize Counter:
· set value to o.
3) Generate Values:
 · while values is less than n:
 · Yield the current value.
 increment value by 1.
4) Create generator object:
· (all my_generator(11) to create a generator object.
5) iterate and Print Values:
· For each value Produced by the generator object.
      Print Value.
```

```
Outlut:
            Enter the starting number: 1
        Ther the ending number : 50
           6 more a partie that increases a series of the factors that is
                                       numbers The Strengton
Hadrice a servence of rumbers when thousand was served.
ald ald break
    soquence of numbers starting from a complete
                                                                                                                  p Ation and with a
                     Step of 1 if no values are provided
    HOTOTO DE PROPER DE PROMENS DE PROPER DE STORMENTE DE STO
                                                                                        Metine Generator function:
                 nefine the function number - somence ( start, end. stell) =1)
                                                                                               I midize (urrent value:
                                                             tests so BUIDY SH & granes of Start.
                                                                                                              destrote Sequence:
                              on of loves than are end to end
                                                       the (nitent veine of Plais.
                                                                                                                                      inchangent .
                                                                                   (wrent 68 see.
                                                                                                            I het user input:
                           · REDS the Starting number (start) them uses influt
                            taring most (big) bottom governs get body.
                                                                                                                                    941- 6004 1
                         turni 1020 off most (90+2) 51910V 15+2)
                                                                                             1 Cheste menerator objet.
recite generator office to calling sumbor-sequence (sent.ord. stell)
                                                                                              · 20 00 0 606400 - 1020 - 164
                                                                                       Thirt Generated Scauence:
                                                                                                                   thinks over the
           dayle release sat us becaused some
                                                                                                                    SHIPV 4109 WILL
```

8-1(b) frogram: do me def my - Benerator (m): 1 10 m Location 4 ya corna) MAI LIN #intialize counter and intermed and intermediate counter Value = 0 # 100P until counter is less than n while votue Kn: # produce the current value of the counter yield value # Provide the turrent value o. # increment the counter Value +=1 # iterate over the generator object Produced by my-generator for value in my _9enerator (3): # Print each value Produced by generator Print (value)

8.2 Imagine you are working on a messaging application that needs to format message differently based on the user's preference

users can choose to have their message automotically converted to uffer (ase (for emphasis) or to lower case (for a softer tone). You a Provided with two decorators: upper case _ decorator and lower case _ de These decorators modify the behaviour of the functions they decorat

by converting the text to upper case or lowercase, respectively-write. Program to implement it.

Algorithm: 1) create Decorators:

· Define uffer case - decorator to convert the result of a function uller case. · Define lower case decorator to convert the result of a function

lower case. 2) Define Functions:

· Define Should function to return the influt text. APPly a upper case - decorator to this function.

· Define whister function to return the influt text. APPLY @ were case - decorator to this function,

HI, I AM CREATED BY A FUNCTION PASSED AS AT AUGUMENT his i am created by a function based as an argument. is not say at cortain tipou logg total still go suld trained off mining 10 00100 400400 944 36400 15 totano on the monoron is 1 = 12 50150 STREET EN BY PRINCIPLE SHE AND THREET A blut each value begunes by severator tot colessifice enfections to no grixion see not enferme to अर्थे के प्रतिकात के प्रतिकार POSTERIOR AND SERVICE WITH BACH OF 25004) UD) SER 40 WHEN (ase (for emphasis) or to lower (ase (for a softer tone). You are est-sea decional from decional - 920) 19991 :2 totoral 96 out that keepling the decidence modify the behaviour of the functions they decide ש ניחוצרווים וויפ לפגל דם עוףש נפגף סד נמטפרפגב, דפקבנהייפוץ- עודים ב : गार्थाभवश्व liveste Decorators: of the territor of contrary of topology of a fraction of a fraction to (920) 19/11 of nothing to be those of the result of a function in . 920) DUM Locking trunctions: Elder that the miles of without from authority nothing title of demonst - sen will be eight ups tules out the second to the select test and " Charge co co _ secondor to the fire

3) Define Greet function: · Define Great function that · Alcerts a function (func) as inflat. · call this function the text " Hi, I am created by a function . Print the result. the contract to see is not allowed u) Frecuse the Program: greet (should) to Print the greeting in uppercase. · Call greet (whisper) to Print the greeting in lower case. Program :der upper case - decorate (func): def wrapper (text): return func (text)_upper () return wrapper def lower (ase - decorator (func): def wrapper (text): return fine (text). lower() return wrapper @ uller case_ decorator def shout (text): return text @ lower case - decorator det stout (text): return bext def greet (func): greating = func ("Hi, lam created by a function possed as an ambenen

greet (shout) greet (whister)

GON	
VEL TECH - CSE	
LX NO.	8
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	2
TIVA VOCE (5)	5
RECORD (5)	
1 56 (20)	
3 ALTH H DATE	115

Results. Thus the Python Arostain to And implement Python se nerose and personators was successfully executed and the ordinat was verified.