**Illustrative programs**

**1).**Python program to find square root of the number using Newton’s method.

**Newton’s Method Formula :**

0.5\*(approx+n/approx)

**Source code :**

1. “””
2. Square root (Newton Method)
3. Formula:
4. 0.5\*(approx+n/approx)
5. “””
6. #ask a input to Enter a value
7. number = int(input(“Enter a finding number : “))
8. #Creating a function
9. def Newton\_sqrt(n) :
10. #approx value is stored in approx\_value variable
11. approx\_value = 0.5 \* n
12. #correct value is stored in final\_value variable
13. final\_value = 0.5 \* (approx\_value + n / approx\_value)
15. #Creating a while loop
16. #final\_value and approx\_value not satisfy execute a loop
17. while final\_value != approx\_value :
18. approx\_value = final\_value
19. final\_value = 0.5 \* (approx\_value + n / approx\_value)
20. #final\_value and approx\_value is same exit a loop
21. return approx\_value
22. #return a approx\_value
23. print(“Given value is : “, number)
24. #Display a asked value
25. #calling a function in print statement
26. print(“The square is : “,Newton\_sqrt(number))
27. #Display a square root value

**Output :**

**Enter a finding number : 8**

Given value is : 8

The square is : 2.82842712474619

**Algorithm :**

1)start

2)define a function name newton sqrt()

3)initialize approx. As 0.5\*n and the better as 0.5\*(approx+n/approx)

4)use a while loop with a condition better-approx. To perform the following

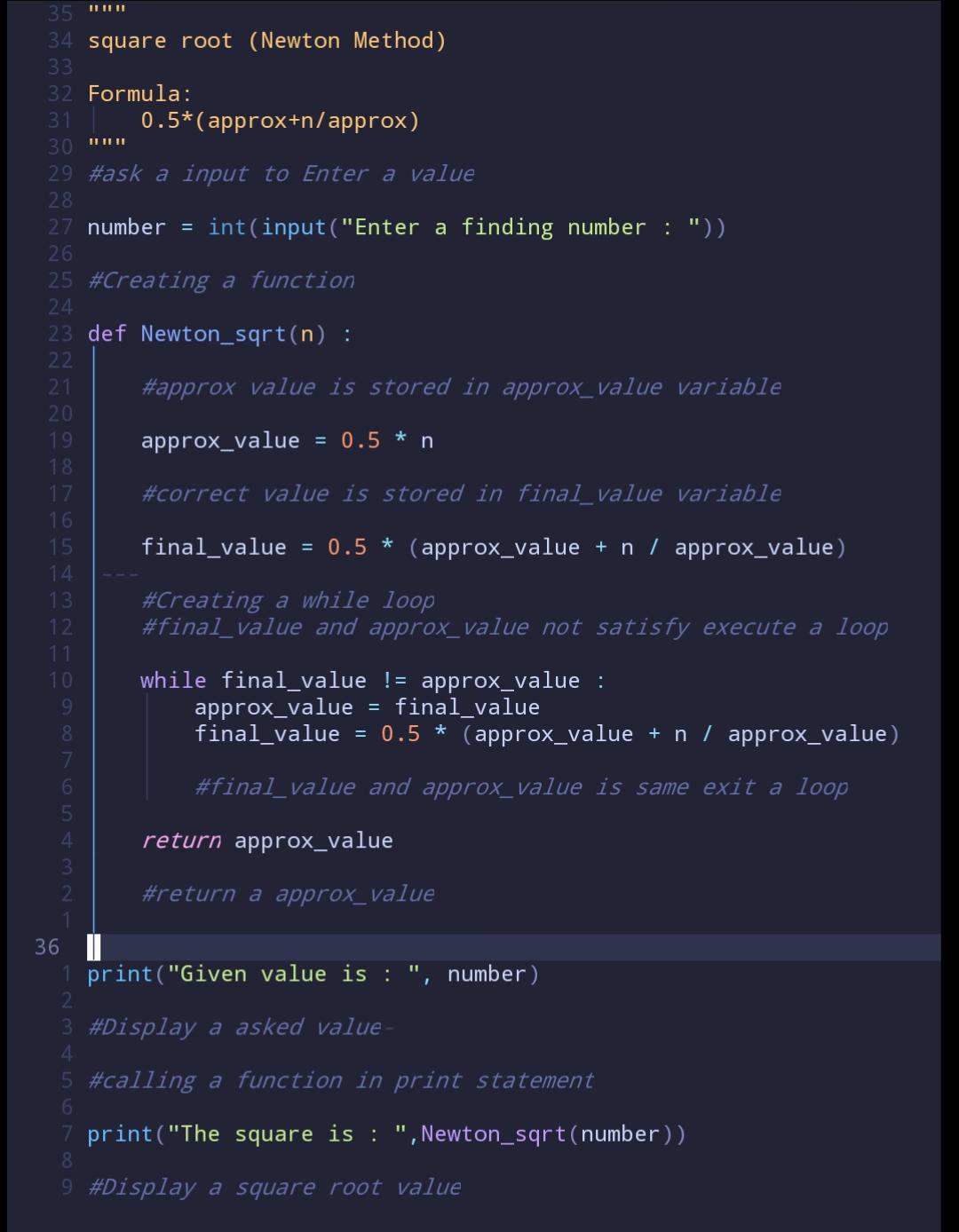
a)set approx-better

b)better=0.5\*(approx+n/approx)

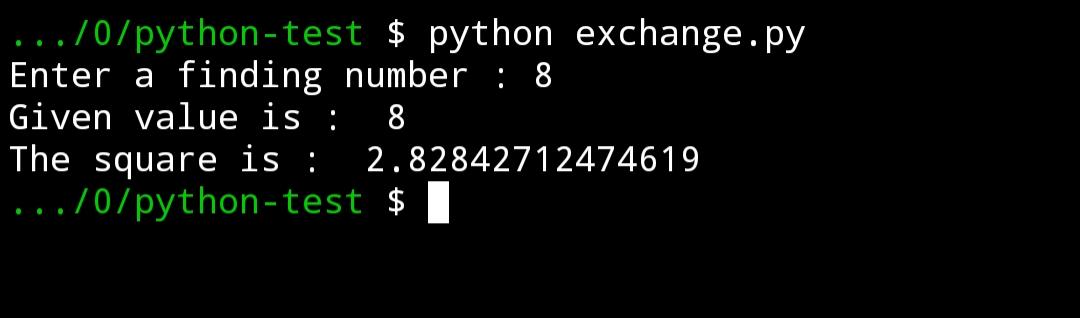
5)print the value of approx.

6)stop

**Proof :**

**Source code:**

**Output:**

****