**Practical 5:**

**Aim: - Write a program to check the syntax of looping statements in Python language.**

**Theory: -**

## **Python For Loops**

A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string).

This is less like the for keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages.

With the for loop we can execute a set of statements, once for each item in a list, tuple, set etc.

Example :-

fruits = ["apple", "banana", "cherry"]  
for x in fruits:  
  print(x)

Example 2:-

for x in range (0,x):  
  print(x)

**Syntax**:-

for counter in iterable:

where,

for and in are keywords

counter can be any variable you used to get the value from iterable

iterable is an object that can be “iterated over”

**Syntax 2:-**

for counter in range (start\_value,end\_value)

where,

for and in and range are keywords

counter can be any variable you used to get the value from iterable

start is interger used to assign the index for looping over an iterable

end is interger used to assign the the end range for looping over an iterable

**Code:-**

import sys

str = input("Enter for loop to check syntax: \n")

striped\_string = str.replace(" ","")

in\_pos = str.find("in")

col\_pos = str.find(":")

range\_exist = [1 if striped\_string.find("inrange") != -1 else 0][0]

#check if for exist

print("Checking for keyword exist.....")

if str[0:3] != "for":

print("What are you trying to do?\n")

sys.exit()

#check if space after for exist

print("Checking space after for .....")

if str[3] != " ":

print("Forgot a space after for...")

sys.exit()

#check for counter variable

print("Checking if counter variable exist .....")

if striped\_string.find("forin") != -1:

print("Forgot to give a counter varible to loop ..")

sys.exit()

#check if space before in exist

print("Checking if space before in exist .....")

if str[in\_pos-1] != " ":

print("Forgot the space before in...")

sys.exit()

#check if in exist

print("Checking if in keyword exist .....")

if in\_pos == -1:

print("Forgot the in...")

sys.exit()

#check if space after for exist

print("Checking if sapace after in exist .....")

if str[in\_pos+2] != " ":

print("Forgot a space after in...")

sys.exit()

#check if colon at the end exist

print("Checking if in : exist at the end of the string.....")

if col\_pos+1 != len(str.strip()):

print("Forgot the : ...",)

sys.exit()

#checking if counter variable exist

print("Checking if counter variable exist.....")

for\_in = str[3:in\_pos].replace(" ","")

if len(for\_in) > 1:

print("Something is wrong")

sys.exit()

#check for loop variable or range variable

print("Checking if loop type is a range based.....")

if range\_exist == 1:

#check if range exist

print("Checking if range is provided or not....")

if striped\_string.find("range:") != -1:

print("Forgot the give a the range")

else:

#check if space after range exist

print("Checking if range is provided after range keyword....")

range\_pos = striped\_string.find("range")

print("Checking range type....")

if striped\_string.find("range(") != -1:

print("Checking if ( exist....")

open\_pos = striped\_string.find("range(")+5

print("Checking if ) exist....")

close\_pos = striped\_string.rindex(")")

print("Checking if ( is before ) exist....")

#if ( is after range

if open\_pos < range\_pos:

print("Where did you even put the ( ?")

#if ) is afrer ( and range

elif close\_pos < range\_pos or close\_pos < open\_pos:

print("Where did you even put the ) ?")

else:

print("Checking if two ranges exist....")

list\_range = striped\_string[open\_pos+1:close\_pos].replace(" ","")

#check if comma seprating is not at start

if list\_range.find(",") == 0:

print("What is the start of the range ?")

#check if comma seprating is not at end

elif list\_range.find(",") == len(list\_range)-1:

print("What is the end of the range ?")

sys.exit()

elif range\_exist == 0:

print("Checking if loop variable is provided.....")

if striped\_string.find("in:") != -1:

print("Forgot to give a iterable..")

sys.exit()

print("No errors")

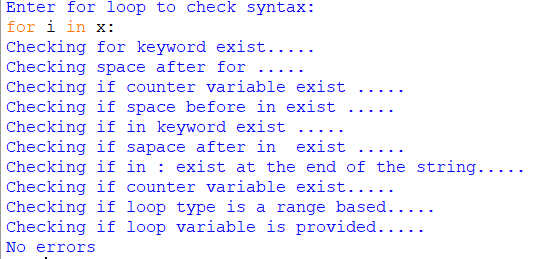
**Output:-**

Input:-

for i in x:

Expected output:-

No error

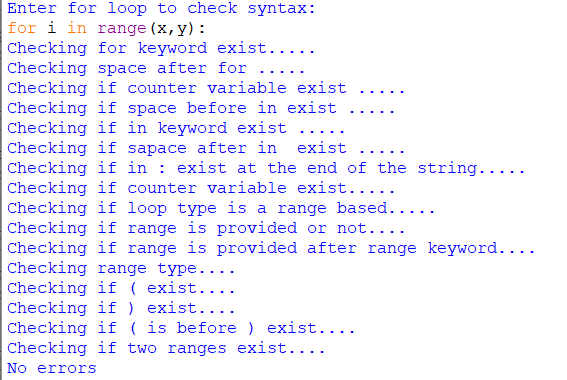


Input:-

for i in range(x,y):

Expected output:-

No error

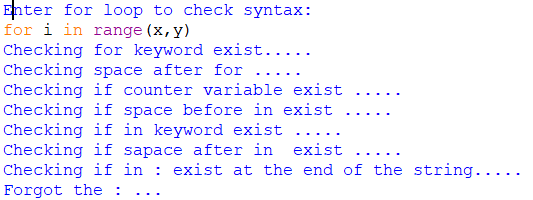


Input:-

for i in range(x,y)

Expected output:-

: missing

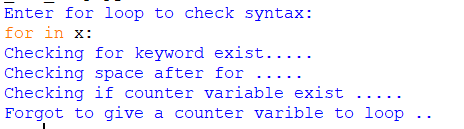


Input:-

for in x:

Expected output:-

Missing counter variable

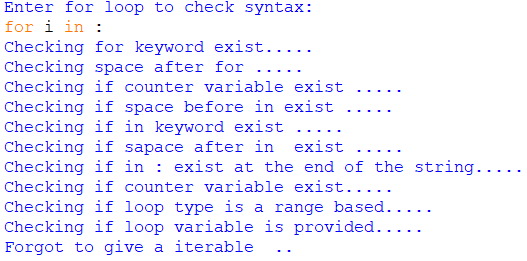


Input:-

For i in :

Expected output:-

Missing iterable

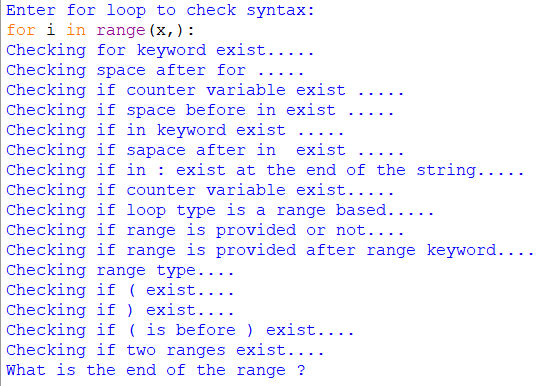


Input:-

For i in (x,): :

Expected output:-

Missing end of the range



**Conclusion:-**

We successfully checked the syntax of for loop in python

**References :-**

https://www.geeksforgeeks.org/c-program-to-check-syntax-of-for-loop/