

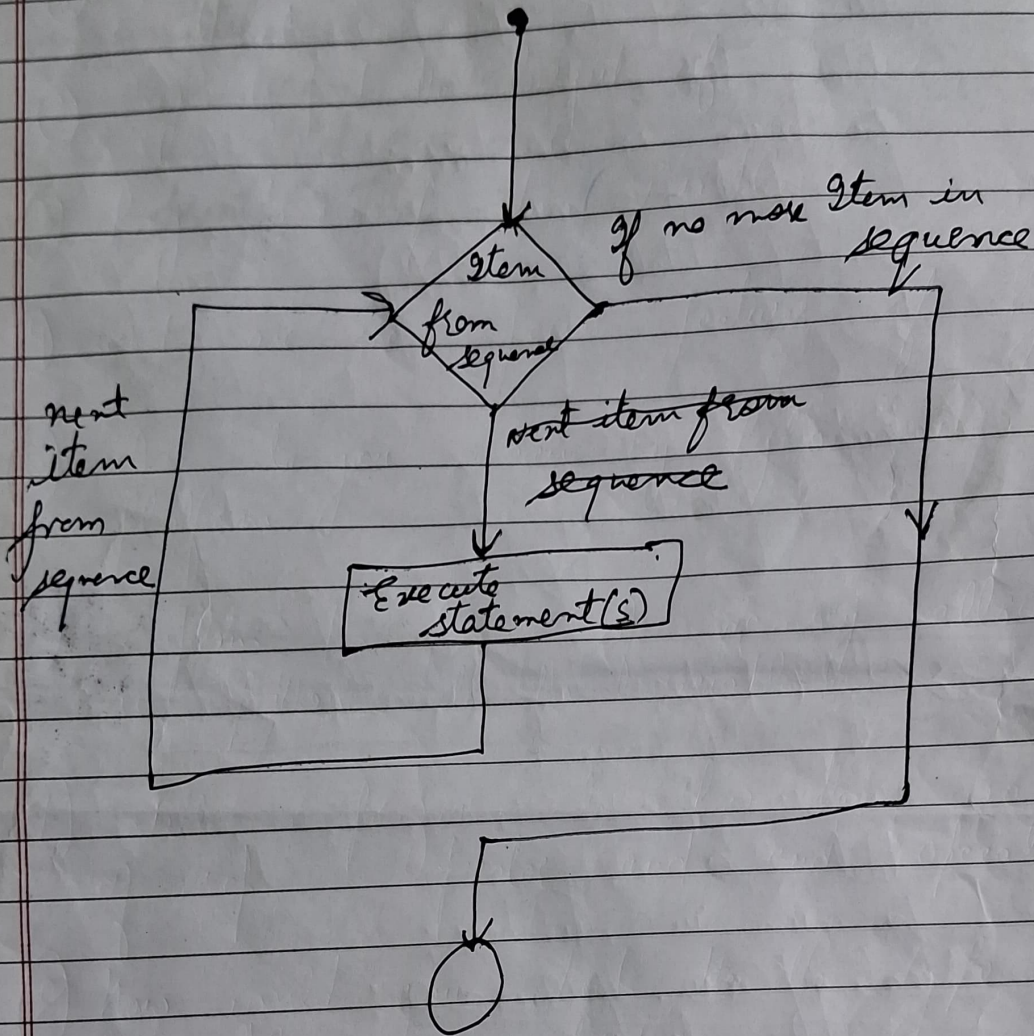
For Loop

①

classmate

Date _____
Page _____

Flowchart of Python for Loop:



eg:
①

zen = " Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated."

```
for char in zen:  
    if char not in 'aeiou':  
        print(char, end='')
```


eg. 2

Python for loop with Tuples

```

numbers = (34, 54, 67, 21, 78, 97, 45, 80, 19)
total = 0
for num in numbers:
    total += num
print("Total =", total)

```

Output:

Total = 539

eg. 3

Python for loop with List

```

numbers = [34, 54, 67, 21, 78, 97, 45, 44, 80, 19]
total = 0
for num in numbers:
    if num % 2 == 0:
        print(num)

```

In the above eg., the for loop traverses a list containing integers and prints only those which are divisible by 2.

Output:

34

54

78

44

80

eg. 4 Python for loop with Range Objects
range() function has the following syntax.

range(start, stop, step)

```
for num in range(5):  
    print(num, end = ' ')
```

```
print()  
for num in range(10, 20):  
    print(num, end = ' ')
```

```
print()  
for num in range(1, 10, 2):  
    print(num, end = ' ')
```

Output:

```
0 1 2 3 4  
10 11 12 13 14 15 16 17 18 19  
1 3 5 7 9
```


eg 5 Tras Use of for loop with dictionaries

```
my-dict = { 'name': 'Alice',  
            'age': 30,  
            'city': 'New York'  
}
```

Iterating Over keys:

```
for key in my-dict:  
    print (key)
```

output:

name
age
city

Iterating Over Values:

```
for value in my-dict.values():  
    print (value)
```

output:

Alice
30
New York

Iterating over key-value pairs:

```
for key, value in my_dict.items():
    print(f"key: {key}, value: {value}")
```

output:

```
key: name, Value: Alice
key: age, Value: 30
key: city, Value: New York.
```

for Else Loop:

Syntax:

```
for variable_name in iterable:
    # starts in the loop
```

```
    :
```

```
else:
```

```
    # starts in else clause
```

```
    :
```

eg:

```
for count in range(5):
    print("Iteration no. {}.".format(count))
else:
    print("for loop over. now in else block")
print("End of for loop")
```


Placeholders `{ }` in the string are replaced by the values provided in the `format ()` method.

Programs :

1. WAP using for loop to print no's between 1 to 10
2. Sum of no's from 1 to 100
3. Print multiplication table of 5
4. find all even no's in a list
5. Python program
5. Try for loop ~~use~~ on all sequence data types
6. Use for loop to access tuple, list, range, dictionary.