🐍 Python Programming Notes

Chapter: Iteration Statements (For Loop & While Loop)

This chapter provides a **deep, conceptual, and examination-oriented explanation** of iteration statements in Python. It includes definitions, syntax, internal working, examples, solved sequence-based questions, and exam practice problems.

🐓 Concept of Iteration

🌧️ What is Iteration?

Iteration means **repeating a set of instructions multiple times** until a condition is satisfied.

In programming, iteration is implemented using **loops**.

🌧️ Why Iteration is Important?

- Avoids repetition of code
- Makes programs efficient
- Reduces code length
- Essential for pattern problems
- Used in searching, sorting, counting, summation, etc.

🐕 Types of Iteration Statements in Python

Python provides two main loops:

1. for loop
2. while loop

🐗 The for Loop (Definite Iteration)

🌧️ Definition

A `for` loop is used when the number of iterations is known in advance. It is generally used to iterate over a sequence (list, string, range, tuple, etc.).

🌧️ General Syntax

```
for variable in sequence:
    statements
```

🌧️ Working Principle

1. The loop variable takes the first value from the sequence.
2. The body executes.
3. The variable takes the next value.
4. Continues until sequence ends.

🐚 The range() Function (Very Important for Exams)

🌧️ Definition

`range()` generates a sequence of numbers.

🌧️ Syntax Forms

1. range(stop)

```
range(5)
```

Output sequence: 0, 1, 2, 3, 4

2. range(start, stop)

```
range(2, 6)
```

Output sequence: 2, 3, 4, 5

3. range(start, stop, step)

```
range(1, 10, 2)
```

Output sequence: 1, 3, 5, 7, 9

⚠ Important Rule: - Stop value is NEVER included.

🐜 Solved Sequence-Based Questions (Exam Oriented)

⛅ Question 1: What will be the output?

```
for i in range(1, 6):
    print(i, end=" ")
```

🦝 Solution:

Range generates: 1 2 3 4 5

Output:

```
1 2 3 4 5
```

⛅ Question 2: Predict the Output

```
for i in range(5, 0, -1):
    print(i)
```

🦝 Understanding:

Start = 5
Stop = 0 (not included)
Step = -1

Output:

```
5
4
3
2
1
```

⛅ Question 3: Find Output

```
for i in range(0, 10, 3):
    print(i)
```

Sequence: 0, 3, 6, 9

Output:

```
0
3
6
9
```

⛅ Question 4: Nested for Loop Output

```
for i in range(1, 4):
    for j in range(1, 3):
        print(i, j)
```

Execution Order: (1,1) (1,2) (2,1) (2,2) (3,1) (3,2)

Output:

```
1 1
1 2
2 1
2 2
3 1
3 2
```

🐝 The while Loop (Indefinite Iteration)

🌧️ Definition

A `while` loop executes a block of code **as long as the condition remains True**.

🌧️ Syntax

```
while condition:
    statements
```

🌧️ Working Principle

1. Condition is checked first.
2. If True → body executes.
3. Control goes back to condition.
4. If False → loop stops.

🐟 Example of while Loop

Example 1: Print 1 to 5

```
i = 1
while i <= 5:
    print(i)
    i += 1
```

Example 2: Sum of First N Numbers

```
n = int(input("Enter number: "))
i = 1
sum = 0

while i <= n:
    sum += i
    i += 1

print("Sum =", sum)
```

🐠 Loop Control Statements

1. break Statement

Definition:

Terminates the loop immediately.

```
for i in range(1, 10):
    if i == 5:
        break
    print(i)
```

Output: 1 2 3 4

When to Use break?

- Searching problems
- Menu driven programs
- Stop when condition satisfied

---

2. continue Statement

Definition:

Skips current iteration and moves to next iteration.

```
for i in range(1, 6):
    if i == 3:
        continue
    print(i)
```

Output: 1 2 4 5

When to Use continue?

- Skip unwanted values
- Ignore specific condition

---

3. pass Statement

Definition:

Does nothing. Used as placeholder.

```
for i in range(5):
    pass
```

🐡 Difference Between for and while Loop (Detailed)

Basis	for Loop	while Loop
Type	Definite iteration	Indefinite iteration
Condition	Implicit	Explicit
Initialization	Automatic	Manual
Update	Automatic	Manual
Infinite Loop Risk	Less	More
Best Used When	Iteration count known	Condition based repetition

---

🌈 Common Mistakes in Exams

❌ Forgetting colon (:)
❌ Wrong indentation
❌ Forgetting to update variable in while loop
❌ Confusing stop value in range()

🐓🐓 Long Answer Exam Questions (Solved)

Q1. Write a program to print factorial of a number.

```
n = int(input("Enter number: "))
fact = 1

for i in range(1, n+1):
    fact *= i

print("Factorial =", fact)
```

Q2. Write a program to check whether a number is prime.

```
n = int(input("Enter number: "))
flag = True

for i in range(2, n):
    if n % i == 0:
```

```
        flag = False
        break

if flag:
    print("Prime Number")
else:
    print("Not Prime")
```

Q3. Write a program to print this pattern:

* **

```
for i in range(1, 5):
    print("*" * i)
```

🐓🐕 Practice Questions (Very Important for Exams)

Short Answer / Output Based

1. Predict output of: range(2, 10, 3)
2. What is difference between break and continue?
3. Can a while loop run infinitely? Explain.
4. Why is stop value not included in range()?

Programming Questions

1. Print first 20 even numbers.
2. Print reverse of a number.
3. Count digits in a number.
4. Print Fibonacci series upto n terms.
5. Print multiplication table from 1 to 10.

🔥 Conclusion

Iteration statements are one of the most important concepts in programming.

✔️ for loop → when iterations known
✔️ while loop → when condition controls repetition
✔️ break → stop loop

✔️ continue → skip iteration
✔️ pass → placeholder

Mastering loops is essential for solving: - Pattern problems - Number problems - Competitive exam questions - Real-world logic building

---

☸️ Final Advice for Students: Practice output-based questions daily. Most examination papers contain at least one question from loop sequences or nested loops.