

Iteration in Python is a fundamental concept that allows you to process items in a collection (like lists, tuples, dictionaries, or strings) one at a time.

The most common way to iterate in Python is using a for loop:

```
pythonCopyfruits = ["apple", "banana", "cherry"]
for fruit in fruits:
    print(fruit)
```

While Loops

While loops continue executing as long as a condition remains true:

```
pythonCopycount = 0
while count < 5:
    print(count)
    count += 1
```

Comprehensions

List, dictionary, and set comprehensions provide a concise way to iterate and create new collections:

```
pythonCopynumbers = [1, 2, 3, 4, 5]
squared = [x**2 for x in numbers] # [1, 4, 9, 16, 25]
```

Iterator Functions

Python provides built-in functions for iteration:

```
pythonCopy# range() generates a sequence of numbers
for i in range(5): # 0, 1, 2, 3, 4
    print(i)
```

enumerate() provides both index and value

```
for index, value in enumerate(["a", "b", "c"]):
    print(f"Index {index}: {value}")
```

Iterating Over Dictionaries

```
pythonCopyperson = {"name": "John", "age": 30, "city": "New York"}
```

Keys

```
for key in person:
    print(key)
```

Values

```
for value in person.values():
    print(value)
```

Both

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

“Hello World” 100 times

- That’s a problem.
- Should we copy and paste the program 100 times?
- Then what if we are asked to print hello world for 50000 times?

Better approach

- Let’s ask the program to execute the line `print(“Hello World”)` for 100 times by itself.
- Great idea. But how can we do that?

while Statement

- There comes *while*. see the code bellow:

```
count=1
while count<=5:
    print("Hello World")
```

- Here we have asked the program to execute the print command for 100 times.

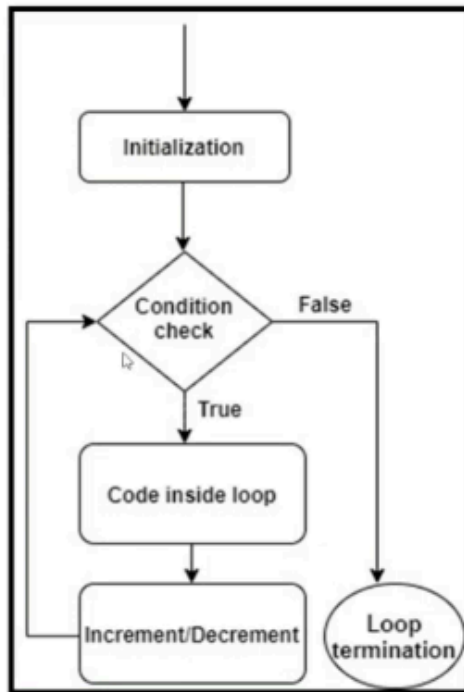
Components of *while* loop

- Initial value/initialization
- Condition
- Block of code
- Increment/decrement

Example:

```
count = 1
while (count<=5):
    print("Hello World")
    count += 1
```

Flowchart



```
count=1
while count<=5:
    print("Hello World")
```

Loop Control Keywords

- break
- continue

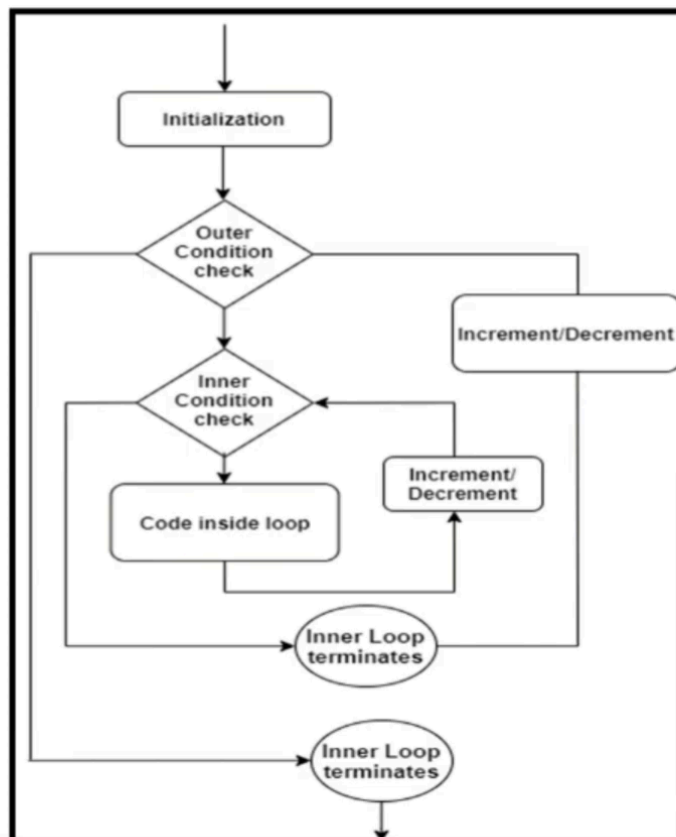
Infinite loop

- The program is going to be executed for an infinite period of time.
- Why?

```
count=5  
while count>=1:  
    print(count)  
    count=count+1
```

- Because for every case the condition remains true.

Nested *while*



Nested While

```
outer = 1
while outer <=2:
    inner = 1
    while inner<=3:
        print(outer,',',inner)
        inner = inner +1
    print('Inner Loop Terminates')
    outer = outer+1
print('Outer Loop Terminates')
```

```
1 , 1
1 , 2
1 , 3
Inner Loop Terminates
2 , 1
2 , 2
2 , 3
Inner Loop Terminates
Outer Loop Terminates
```