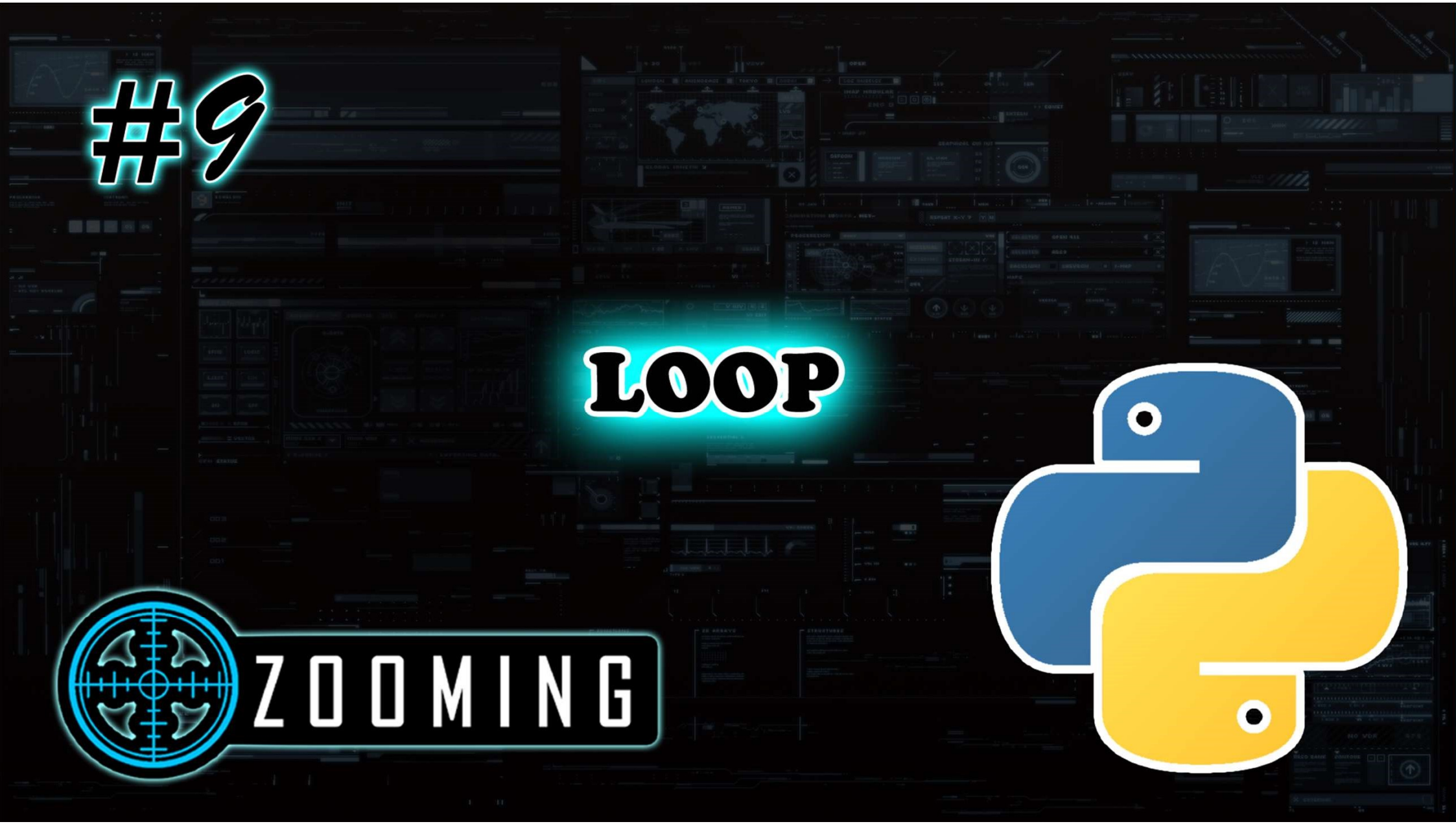
#9
LOOP
ZOOMING

Python Loop

- Python provides three ways for executing the loops.
  - While loop
  - For loop

Python – While Loop

- In python, while loop is used to execute a block of statements repeatedly until a given a condition is satisfied
- When the condition becomes false, the line immediately after the loop in program is executed.

```
count = 0
while (count < 3):
    count = count + 1
    print("Hello Zooming")
```

```
Hello Zooming
Hello Zooming
Hello Zooming
```

Python – While Loop

```
# while like else statement

count = 0
while (count < 3):
    count = count + 1
    print("Hello Zooming")
else:
    print("In Else Block")
```

```
Hello Zooming
Hello Zooming
Hello Zooming
In Else Block
```

```
# Single statement while block
count = 0
while (count == 0): print("Hello Zooming")
```

No result

Python – For Loop

```
# Iterating over a list
print("List Iteration")
l = ["geeks", "for", "geeks"]
for i in l:
    print(i)

# Iterating over a tuple (immutable)
print("\nTuple Iteration")
t = ("geeks", "for", "geeks")
for i in t:
    print(i)
```

```
# Iterating over a String
print("\nString Iteration")
s = "Geeks"
for i in s :
    print(i)

# Iterating over dictionary
print("\nDictionary Iteration")
d = dict()
d['xyz'] = 123
d['abc'] = 345
for i in d :
    print("%s  %d" %(i, d[i]))
```

Python – For Loop with index sequence

- We can also use the index of elements in the sequence to iterate.
- The key idea is to first calculate the length of the list and in iterate over the sequence within the range of this length.

```
list = ["geeks", "for", "geeks"]
for index in range(len(list)):
    print list[index]
else:
    print("In Else Block")
```

```
geeks
for
geeks
Inside Else Block
```

Python – Nested For Loop

- Python programming language allows to use one loop inside another loop.
- A final note on loop nesting is that we can put any type of loop inside of any other type of loop

```
for i in range(1, 5):
    for j in range(i):
        print(i, end=' ')
    print()
```

```
1
2 2
3 3 3
4 4 4 4
```

Python – Loop Control Statement

```
# Continue statement
for letter in 'geeksforgeeks':
    if letter == 'e' or letter == 's':
        continue
    print 'Current Letter :', letter
```

```
Current Letter : g
Current Letter : k
Current Letter : f
Current Letter : o
Current Letter : r
Current Letter : g
Current Letter : k
```

```
# Break statement
for letter in 'geeksforgeeks':
    if letter == 'e' or letter == 's':
        break
print 'Current Letter :', letter
```

```
Current Letter : e
```