```python
# Basics of Loops

"""
LOOPS:
    --> In computer programming, a loop is a sequence of instructions that is repeated until a certain condition is reached.
    --> Two Basic Types of Loops:
        --> 1. For Loop
        --> 2. While Loop
"""

"""
RANGES:
    --> The range type represents an immutable sequence of numbers and is commonly used for looping a specific number of times in for loops.

    range (start, stop[, step])

    range() takes three arguments. Out of the three 2 arguments are optional. I.e., start and step are the optional arguments.

        --> A start argument is a starting number of the sequence. i.e., lower limit. By default, it starts with 0 if not specified.

        --> A stop argument is an upper limit. i.e., generate numbers up to this number, The range()  function doesn’t include this number in the result.

        --> The step is a difference between each number in the result. The default value of the step is 1 if not specified.

Examples:
    --> range(7) gives integers from 0 to 6
    --> range(1, 8) gives integers from 1 to 7
    --> range(1, 10, 2) will give odds from 1 to 10
    --> range(7, 0, -1) will give integers from 7 to 1
"""
print(list(range(7)))
print(list(range(1, 8)))
print(list(range(1, 10, 2)))
print(list(range(7, 0, -1)))

"""
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.
"""
names = ['Abhishek', 'Jonas', 'Torrent']
for name in names:
    print(name)

for i in range(5):
    print('LOOPS')


"""
```

```
WHILE LOOPS:
    --> With the while loop we can execute a set of statements as long as a condition is true.
"""
isOpen = True
counter = 0

while isOpen:
    if (counter == 6):
        isOpen = False
    counter += 1
    print('While Loop')


"""
break Keyword:
    --> The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.
    --> If break statement is inside a nested loop (loop inside another loop), break will terminate the innermost loop.
"""

for i in range(1, 11):
    if (i == 6):
        break
    print(i)

"""
continue Keyword:
    --> The continue statement is used to skip the rest of the code inside a loop for the current iteration only. Loop does not terminate but continues on with the next iteration.
"""

for i in range(11, 21):
    if (i == 13):
        print(f"{i} is Unlucky")
        continue
    print(f"Number: {i}")

```