

In Python, ``break`` and ``continue`` are control flow statements used within loops to modify their behaviour. Here's how they work:

### **``break`` Statement**

The ``break`` statement is used to exit a loop prematurely. When ``break`` is encountered, the loop terminates immediately, and the control moves to the next statement after the loop.

Example:

```
python
# Using break in a loop
for i in range(1, 6):
    if i == 3:
        break # Exit the loop when i is 3
    print(i)
# Output:
# 1
# 2
```

In this example, the loop prints numbers 1 and 2. When ``i`` becomes 3, the ``break`` statement is executed, and the loop exits.

### **``continue`` Statement**

The ``continue`` statement is used to skip the current iteration of the loop and move to the next iteration. When ``continue`` is encountered, the remaining code in the current iteration is skipped, and the loop proceeds with the next iteration.

Example:

```
```python
# Using continue in a loop
for i in range(1, 6):
    if i == 3:
        continue # Skip the rest of the loop when i is 3
    print(i)
```

# Output:

# 1

# 2

# 4

# 5

```

In this example, the loop skips printing the number 3 because of the `continue` statement and continues with the next iteration.

### Usage in `while` Loops

Both `break` and `continue` can also be used in `while` loops.

Example with `break` in a `while` loop:

```
```python
i = 1
while i <= 5:
```

```
if i == 4:
    break # Exit the loop when i is 4
print(i)
i += 1
```

Output:

```
# 1
# 2
# 3
...
```

Example with `continue` in a `while` loop:

```
```python
i = 1
while i <= 5:
    i += 1
    if i == 4:
        continue # Skip the rest of the loop when i is 4
    print(i)
```

Output:

```
# 2
# 3
# 5
# 6
```

...

Summary:

- **`break`**: Terminates the loop entirely and moves control to the code after the loop.
- **`continue`**: Skips the rest of the current iteration and moves to the next iteration of the loop.

Both **`break`** and **`continue`** can be powerful tools for controlling the flow of your loops, enabling you to fine-tune how iterations are executed.