

8) Control Statements (Break, Continue, Pass)

- **Understanding the role of break, continue, and pass in Python loops.**

These three keywords control the flow of loops in Python, each serving a distinct purpose:

1. Break Statement

Purpose: Immediately terminates the entire loop

Common Use :

- Searching for an item (exit once found)
- Preventing infinite loops
- Early termination when a condition is satisfied

```
for i in range(1,10,1):  
    if i==3:  
        break  
    print(i)
```

Output

```
1  
2
```

2. Continue Statement

Purpose: Skips the current iteration and moves to the next one

Common Use :

- Skipping Invalid/empty values
- Filtering specific conditions
- Avoiding nested if-else blocks

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

Output

```
1  
2  
4  
5
```

3. Pass Statement (Do Nothing Placeholder)

A null operation used when syntax requires a statement but no action is needed.

Purpose: A null operation placeholder

Common Use :

- Placeholder for future code
- Empty functions/classes (avoid syntax errors)
- Ignoring specific cases temporarily

```
for i in range(1,5,1):  
    if i == 2:  
        pass # Does nothing  
    print(i)
```

Output

```
1  
2  
3  
4
```

Key Differences:

Statements	Purpose	What it Does
Break	Exit the loop immediately	Stops loop entirely
continue	Skip current iteration	Skips to the next loop round
pass	Placeholder, do nothing	Does nothing at all