Python control statements

→ Control statements in Python are used to control the order of execution of the Program based on the value and logic.

→ Python Provides us with the 3 types of control statement :-

1. Break
2. Continue
3. Pass

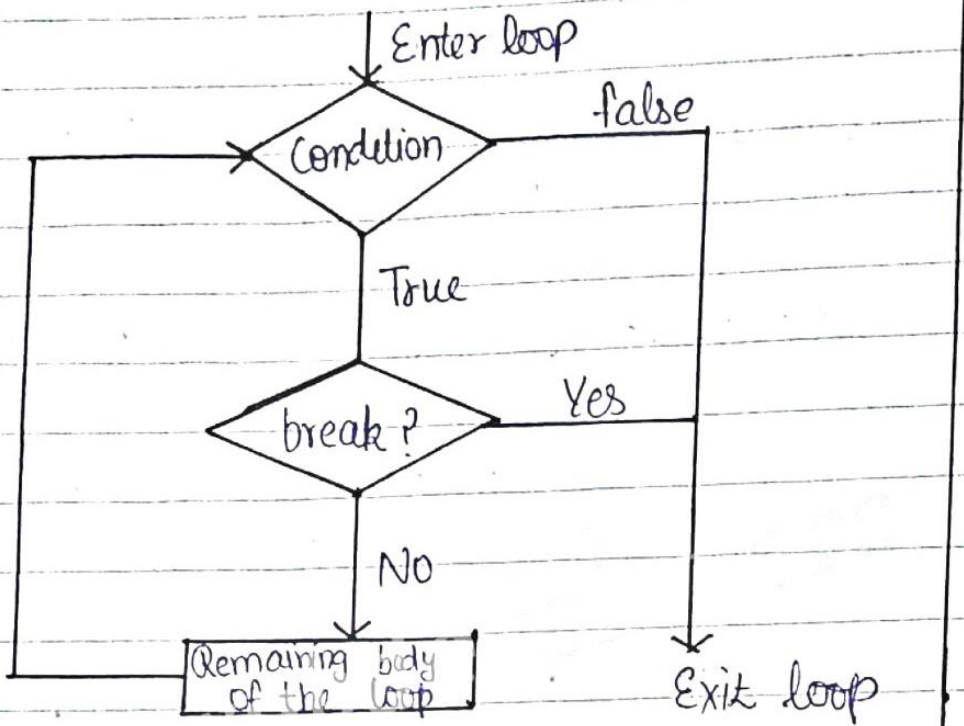
1. Break Statement

- It terminates the looping and transfers execution to the statement next to the loop.
- It brings control out of the loop.

Syntax :-

```
break
```

Flowchart of break



Working of break statement in for loop and while loop :-

```
for var in sequence:
    if condition:
        break
    # code outside for loop
```

```
while test expression:
    if condition:
        break
    # code outside while loop
```

Example:-

```
for ch in "jpwebdevdopers":
    if ch == "d":
        break
    print(ch)
```

OutPut

```
j
p
w
e
b
```

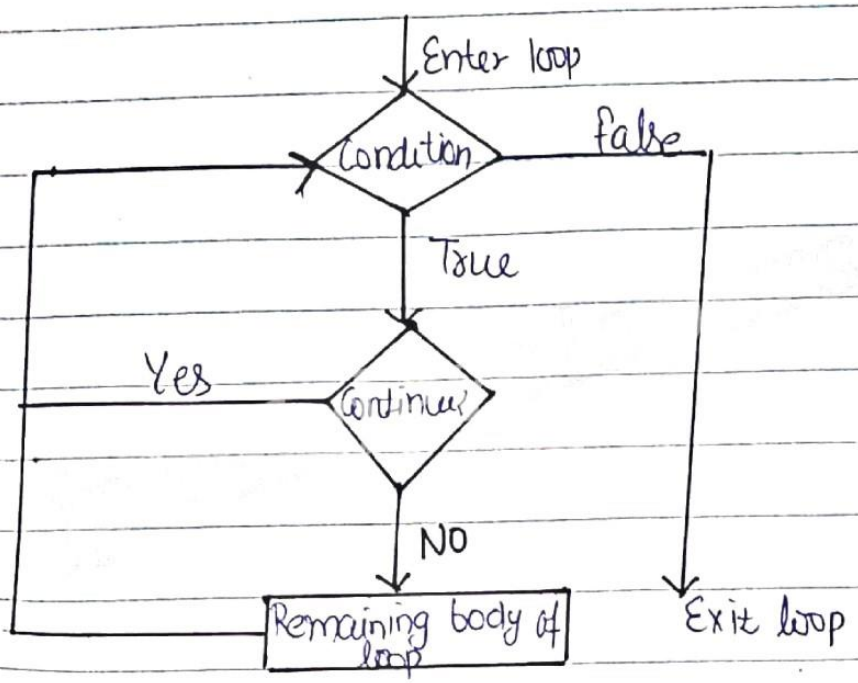
2 Continue Statement

- The `continue` statement is used to skip the rest of the code inside a loop for the current iteration only.
- Continue statement is used in a loop to go back to the begining of the loop.
- loop does not terminate but continues on with the next iteration.

Syntax of Continue

```
continue
```

flowchart of continue



Example

```
for ch in "Jpwebdevelopers":
    if ch == "p":
        continue
    print(ch)
```

Output:

```
j
w
e
b
d
e
v
e
l
```

3. Pass statement

- The **pass** statement is a null operation ; nothing happens when it executes.
- Pass statement is used to do nothing.
- It is a null statement. (Nothing happens when the pass is executed)
- Pass is useful when we need statement syntactically correct but we do not want to do any operation.

Syntax:-

```
pass
```

Example:-

```
a=2
if a==2:
    print("a is equal to 2")
else:
    pass
```

Output: a is equal to 2