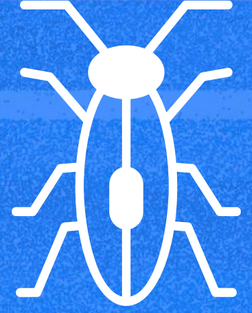
:(

BREAK AND CONTINUE STATEMENT.

LOOP CONTROL STATEMENTS

PRESENTED BY : ADITYA RAJ, ADITYA, BIDYA SAGAR, ABDUL BAASIT, AKASH

:(



WORK? HOW DOES A LOOP WORK? HOW DOES A LOOP

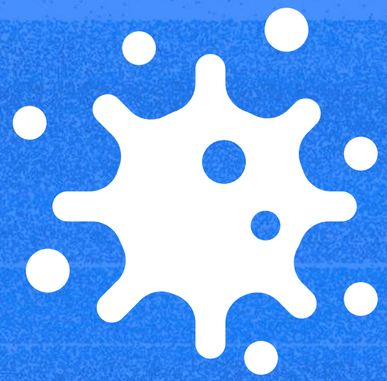
INITIALIZATION

CONDITION CHECK

BODY EXECUTION

VARIABLE UPDATES

End - This cycle repeats until the condition becomes false



The BREAK Statement

The **break** statement is used to **terminate the nearest enclosing loop** immediately, regardless of the loop condition.

Example - To stop a search early

```python
numbers = [10, 20, 30, 40, 50]

target = 30

for n in numbers:
    if n == target:
        print("Found:", n)
        break      # loop stops immediately
    print("Checking:", n)

print("Loop ended.")
```

:(

The CONTINUE Statement

The **continue** statement is used to **skip the remaining statements of the current iteration** and move on to the next iteration of the loop

Unlike break, **continue doesn’t exit the loop**. It only skips one iteration.

Example - To skip unwanted values

```python
numbers = [5, -1, 7, -3, 10]

for n in numbers:
    if n < 0:
        continue   # skip this iteration
    print("Processing:", n)
```

:(

Some use cases:

Use Case	Statement
Stop searching when found	`break`
Skip invalid inputs	`continue`
Exit early from test cases	`break`
Skip processing of unwanted data	`continue`

:(

Analogy?

BREAK – “Teacher says class is dismissed - everyone leaves immediately

CONTINUE – “Teacher says skip this topic – move on to next topic

:(

Your PC ran into a problem and needs to restart. Anyways THANKS FOR LISTENING, if you have any questions ask away.

103% complete

For more information about this issue and possible fixes, visit
http://windows.com/stopcode

If you call a support person, give them this info:
Stop code: LMAO_DED