

While loop

- While ~~the~~ loop is an entry control loop because firstly, the condition is checked, then the loop's body is executed.
- The statement of while loop may not be executed at all.
- The while loop terminates when the condition becomes false.
- In a while loop, the test condition variable must be initialized first to check the test condition in the loop.
- In a while loop, at the end of the condition, there is no semicolon.
Syntax:

While (condition)

- While loop is not used for creating menu-driven programs.

do-While loop

- The do-while loop is an exit control loop because in this, first of all, the body of the loop is executed then the condition is checked true or false.
- The statement of the do-while loop must be executed at least once.
- As long as the condition is true, the compiler keeps executing the loop in the do-while loop.
- In a do-while loop, the variable of test condition initialized in the loop also.
- In this, at the end of the condition, there is a semicolon.
Syntax:

While (condition);

- It is mostly used for creating menu-driven programs because at least one time. The loop is executed whether the condⁿ is true or false.

- In a while loop, the number of executions depends on condition defined in the while back block.

- In a do-while loop, irrespective of the condition mentioned, a minimum of 1 execution occurs.