

# C Language

## Chapter # 12

### Loop Constructs

Lecture: 39

# Today's Lecture

- ▶ do-while Loop and its working with example.

# do-while loop:

- ▶ This loop executes one or more statements while the given condition remains true.
- ▶ In this loop, the conditions comes after the body of loop.
- ▶ This loop is important in a situation where a statement must be executed at least once.
- ▶ **The important point about do-while loop is that the condition ends with a semicolon.**

# *Working of do-while loop*

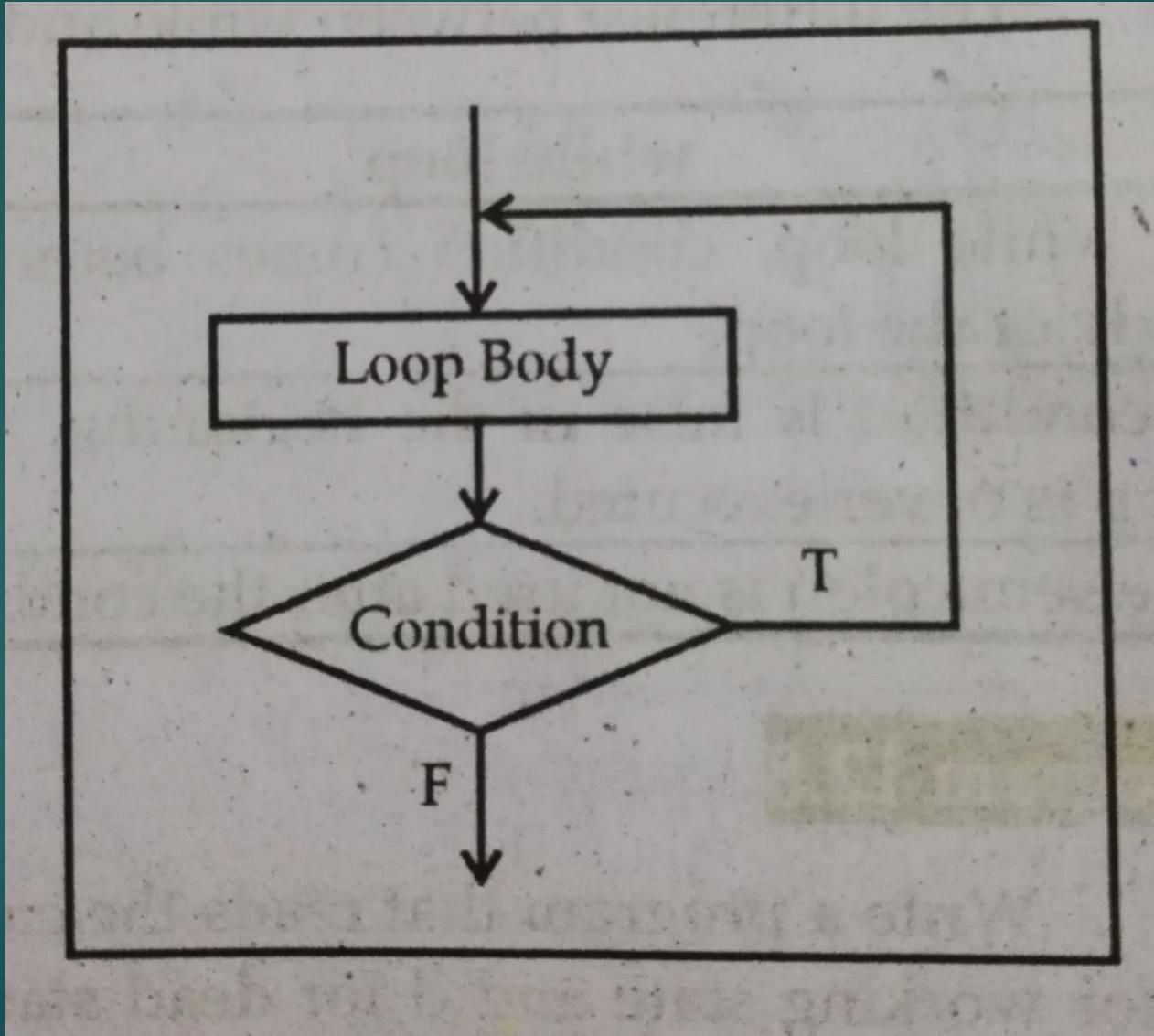
- ▶ First of all the body of loop is executed.
- ▶ After executing all statements in the body, the condition is evaluated.
- ▶ If the condition is true, the control again enters in the body of the loop and executes all statements in the body.
- ▶ This process continues as long as condition remains true.
- ▶ The loop terminate when the condition becomes false.
- ▶ This loop is executed at least once even if the condition false in the beginning.

# *do-while loop*

## **Syntax:**

```
do
{
    Statement 1;
    Statement 2;
    Statement 3;
}
while(condition);
```

# Flowchart (do-while loop)



# Example

**Write a program that displays Pakistan five times using do-while loop.**

```
#include <stdio.h>
#include<conio.h>
main()
{
    int n;
    n = 1;
    do
    {
        printf("Pakistan\n");
        n++;
    }
    while(n <=5);
}
```

# Difference between while and do-while loop

## While loop

1. In while loop, the condition comes before the body of the loop.
2. If condition is false in the beginning, while loop is never executed.
3. The semicolon is not used after the condition.

## do-while loop

1. In do-while loop, the condition comes after the body of the loop.
2. If condition is false in the beginning, do-while loop executed at least once.
3. The semicolon is used after the condition.



# The End

**Read this topic from your books and ask question if any confusion.**

# JAZAKALLAH