Control Flow

For loop | While loop | Do-while loop

Introduction to Loops

A loop is a programming construct that allows repeating a block of code multiple times.

Useful for:

- Repeating a task
- Iterating through lists or data
- Automating repetitive operations

Common types of loops:

- for loop
- while loop
- do-while loop

For Loop – Definition

Used when the number of iterations is **known or predictable**.

Syntax:

```
for (initialization; condition; increment) {
  // code block
}
```

Key points:

- Initialization runs once at the beginning
- Condition is checked before each iteration
- Increment executes after each iteration

For Loop – Example

```
for (int i = 1; i <= 5; i++) {
  print('Iteration $i');
}

/// Output
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
```

- i starts at 1
- Loop runs while i <= 5
- i increments by 1 after each iteration

While Loop – Definition

Used when the number of iterations is not fixed and depends on a condition.

Syntax:

```
while (condition) {
  // code block
}
```

Key points:

- Condition is checked before each iteration
- If the condition is false initially, the loop body may not execute at all

While Loop – Example

```
int i = 1;
while (i <= 5) {
  print('Iteration $i');
  i++;
}
/// Output
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
```

- The condition is evaluated at the start of each iteration
- Increment happens inside the loop

Do-While Loop – Definition

Used when the loop must execute at least once, even if the condition is false.

Syntax:

```
do {
  // code block
} while (condition);
```

Key points:

- Loop body runs first
- Condition is checked after each iteration

Do-While Loop – Example

```
int i = 1;
do {
  print('Iteration $i');
  i++;
} while (i <= 5);
/// Output
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
```

- Code runs once before checking the condition
- If the condition is false, it will not repeat

Real World Use Cases

- **for loop**: Iterating over a list of products, printing numbers, pagination.
- **while loop**: Waiting for user input, monitoring a sensor value.
- **do-while loop**: Showing a menu at least once, password verification loop.

Key Takeaways

- Choose the loop type based on when you want the condition to be checked.
- Ensure the condition will eventually become false to avoid infinite loops.
- Loops make code more efficient and cleaner by avoiding repetition.