**Control flow statements**

**Control flow statements** in Java are instructions that determine the **order in which code is executed** in a program. They enable the program to make decisions, repeat tasks, or jump to specific parts of the code based on certain conditions.

* Java has **three types** of control flow statements:
  1. **Decision Making Statements** (**Make choices**) e.g., if, switch
  2. **Looping Statements (Repeat actions)** e.g., for, while, do…while, enchanced for loop
  3. **Jump Statements (Control execution)** e.g., break, continue, return

**1. Decision-Making Statements**

These are used to execute code conditionally.

**if Statement**

if (condition) {

// code executes if condition is true

}

**if-else Statement**

if (condition) {

// executes if true

} else {

// executes if false

}

**if-else-if Ladder**

if (condition1) {

// code

} else if (condition2) {

// code

} else {

// default code

}

**switch Statement**

Efficient alternative to many if-else blocks (for discrete values).

switch (expression) {

case value1:

// code

break;

case value2:

// code

break;

default:

// code

}

**2. Looping Statements**

Used to execute a block of code repeatedly.

**for Loop**

for (int i = 0; i < 5; i++) {

System.out.println(i);

}

**while Loop**

int i = 0;

while (i < 5) {

System.out.println(i);

i++;

}

**do-while Loop**

Always runs the loop body at least once.

int i = 0;

do {

System.out.println(i);

i++;

} while (i < 5);

**Enhanced for-each Loop (for arrays/collections)**

int[] numbers = {1, 2, 3};

for (int n : numbers) {

System.out.println(n);

}

**3. Jump Statements**

Used to transfer control to other parts of the code.

**break**

* Exits the current loop or switch.

for (int i = 0; i < 5; i++) {

if (i == 3) break;

System.out.println(i);

}

**continue**

* Skips the current iteration.

for (int i = 0; i < 5; i++) {

if (i == 2) continue;

System.out.println(i);

}

**return**

* Exits from the method.

public int add(int a, int b) {

return a + b; }