**Cấu Trúc Điều Khiển Trong Java**

**1. Cấu Trúc Điều Khiển if, else if, else**

Cấu trúc điều kiện if-else dùng để thực thi các khối lệnh khác nhau dựa trên kết quả của một biểu thức điều kiện (boolean).

**Cấu Trúc if**

Kiểm tra điều kiện, nếu đúng (true), thì thực hiện các câu lệnh bên trong khối if.

if (condition) {

// Các câu lệnh được thực hiện nếu condition là true

}

**Cấu Trúc else**

Được dùng khi điều kiện của if là false. Khối else sẽ được thực hiện.

if (condition) {

// Các câu lệnh thực thi nếu condition là true

} else {

// Các câu lệnh thực thi nếu condition là false

}

**Cấu Trúc else if**

Khi có nhiều điều kiện cần kiểm tra, bạn có thể sử dụng thêm else if để kiểm tra các điều kiện khác.

if (condition1) {

// Thực thi nếu condition1 là true

} else if (condition2) {

// Thực thi nếu condition1 là false và condition2 là true

} else {

// Thực thi nếu tất cả các điều kiện trên đều false

}

**Ví dụ:**

int score = 75;

if (score >= 90) {

System.out.println("Excellent");

} else if (score >= 75) {

System.out.println("Good");

} else {

System.out.println("Needs Improvement");

}

**2. Cấu Trúc Điều Khiển switch-case**

Cấu trúc switch-case được sử dụng để chọn một trong nhiều nhánh dựa trên giá trị của biến.

**Cấu Trúc:**

switch (expression) {

case value1:

// Các câu lệnh thực thi nếu expression == value1

break;

case value2:

// Các câu lệnh thực thi nếu expression == value2

break;

// ...

default:

// Các câu lệnh thực thi nếu không có case nào khớp

break;

}

* **break**: Dùng để thoát ra khỏi khối switch sau khi một case được thực hiện.
* **default**: Khối này sẽ được thực hiện nếu không có case nào khớp.

**Ví dụ:**

int day = 3;

switch (day) {

case 1:

System.out.println("Sunday");

break;

case 2:

System.out.println("Monday");

break;

case 3:

System.out.println("Tuesday");

break;

default:

System.out.println("Invalid day");

break;

}

**Vòng Lặp Trong Java**

**1. Vòng Lặp for**

Vòng lặp for được sử dụng khi bạn biết trước số lần lặp.

**Cấu Trúc:**

for (initialization; condition; update) {

// Các câu lệnh được thực hiện trong mỗi lần lặp

}

* **initialization**: Khởi tạo biến lặp.
* **condition**: Điều kiện để tiếp tục vòng lặp, nếu false vòng lặp sẽ dừng lại.
* **update**: Cập nhật biến lặp sau mỗi lần lặp.

**Ví dụ:**

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

System.out.println(i); // In ra 0, 1, 2, 3, 4

}

**2. Vòng Lặp while**

Vòng lặp while được sử dụng khi điều kiện lặp là không xác định, chỉ dừng khi điều kiện trở thành false.

**Cấu Trúc:**

while (condition) {

// Các câu lệnh được thực hiện nếu condition là true

}

**Ví dụ:**

int i = 0;

while (i < 5) {

System.out.println(i); // In ra 0, 1, 2, 3, 4

i++;

}

**3. Vòng Lặp do-while**

Vòng lặp do-while đảm bảo rằng khối lệnh bên trong sẽ được thực thi ít nhất một lần trước khi kiểm tra điều kiện.

**Cấu Trúc:**

do {

// Các câu lệnh được thực hiện

} while (condition);

**Ví dụ:**

int i = 0;

do {

System.out.println(i); // In ra 0, 1, 2, 3, 4

i++;

} while (i < 5);

**4. Vòng Lặp for-each Để Duyệt Qua Mảng**

Vòng lặp for-each được sử dụng để duyệt qua các phần tử của một mảng hoặc một tập hợp.

**Cấu Trúc:**

for (type element : array) {

// Các câu lệnh thực hiện với từng phần tử element

}

**Ví dụ:**

int[] numbers = {1, 2, 3, 4, 5};

for (int number : numbers) {

System.out.println(number); // In ra 1, 2, 3, 4, 5

}

Trên đây là các cấu trúc điều khiển và vòng lặp.