**B. Chuẩn bị:**

**1. Tìm hiểu về các phép tính logic cơ bản (logical operator) của Java**

|  |  |  |
| --- | --- | --- |
| **Toán tử** | **Mô tả** | **Ví dụ**  **int** a = true;  **int** b = false; |
| && | AND (và): Trả về đúng nếu cả 2 vế đều đúng | boolean result = a && b; //result = false |
| || | Trả về đúng nếu ít nhất một vế đúng | boolean result = a || b; //result = true |
| ! | Phủ định | boolean result = !a; //result = false |

**2. Tìm hiểu về các phép tính so sánh của Java**

|  |  |  |
| --- | --- | --- |
| **Toán tử** | **Mô tả** | **Ví dụ**  **int** a = 5;  **int** b = 6; |
| == | So sánh bằng | boolean result = a == b; //result = false |
| != | So sánh khác | boolean result = a != b; //result = true |
| > | Lớn hơn | boolean result = a > b; //result = false |
| >= | Lớn hơn hoặc bằng | boolean result = a >= b; //result = false |
| < | Nhỏ hơn | boolean result = a < b; //result = true |
| <= | Nhỏ hơn hoặc bằng | boolean result = a <= b; //result = true |

**3. Tìm hiểu về khối lệnh if, if-else, switch**

**Khối lệnh if:** được sử dụng để kiểm tra một điều kiện có giá trị là true hay false. Nếu điều kiện là true, khối lệnh bên trong if sẽ được thực thi. Nếu điều kiện là false, khối lệnh bên trong if sẽ bị bỏ qua.

**Cú pháp câu lệnh if**

• Cú pháp:

if (condition) {

// one or more statements;

}

Trong đó:

• condition: là biểu thức trả về giá trị kiểu boolean

• statements: Các câu lệnh sẽ được thực thi nếu điều kiện trả về **true**

**Khối lệnh if-else:** được sử dụng để kiểm tra một điều kiện có giá trị là true hay false. Nếu điều kiện là true, khối lệnh bên trong if sẽ được thực thi. Nếu điều kiện là false, khối lệnh bên trong else sẽ được thực thi.

**Cú pháp if-else**

• Cú pháp:

if (condition) {

// one or more statements;

}

else {

// one or more statements;

}

Trong đó:

• condition: điều kiện để đánh giá. Nếu condition trả về **true** thì khối lệnh bên trong **if** được thực thi. Nếu condition trả về **false** thì khối lệnh trong **else** được thực thi.

**Khối lệnh switch:** được sử dụng để thực hiện các hành động khác nhau dựa trên giá trị của một biến hoặc một biểu thức. Khối lệnh switch có nhiều nhánh case, mỗi nhánh case tương ứng với một giá trị cụ thể. Nếu giá trị của biến hoặc biểu thức khớp với một nhánh case nào đó, khối lệnh bên trong nhánh case đó sẽ được thực thi. Nếu không có nhánh case nào khớp, khối lệnh bên trong nhánh default (nếu có) sẽ được thực thi.

**switch-case: Cú pháp**

**switch** (**switch**-expression) {

**case** value1: statement(s)1;

**break**;

**case** value2: statement(s)2;

**break**;

...

**case** valueN: statement(s)N;

**break**;

**defaut:** statement(s)-**for**-**defaut;**

}

Trong đó:

• *switch-expression*: là biểu thức trả về giá trị thuộc một trong các kiểu: char, byte, short, int hoặc String

• *value1,…valueN* có cùng kiểu dữ liệu so với switch-expression

• *break* là từ khoá để dừng thực thi các câu lệnh ở phía sau. *break* là không bắt buộc.

• *defaultl* à từ khoá để quy định khối lệnh sẽ được thực thi nếu không có trường hợp nào ở các *case* là đúng. *default* là không bắt buộc.