**Operadores relacionais** são simbolos especiais que são capacidades de realizar operações com determinados operandos e gerando resultados.

Existem dois tipos de operadores, os de **similaridade**, que comparam a igualdade ou diferença, e os de **tamanho**, que comparam o valor dos numeros.

Para **igualdade utilizamos “==”**, para **desigualdade utilizamos “!=”**, note que se utilizar apenas o “=” para igualdade, estaremos fazendo uma **atribuição**, gerando erros.

No caso da comparação de tamanho utilizamos os “**<”**, “**>”**, “**<=”**, “**>=”**.

**Conjunção**, com o simbolo **&&**, trata como **true** quando **ambas as afirmações são verdadeiras**.

**Disjunção**, com o simbolo **||**, trata como **false** quando **ambos os operandos são falsos**.

**Dinjunção exclusiva**, com o simbolo ^, trata como **true** quando **as afirmações são opostas**.

**Negação**, com o simbolo !, **nega** inverte o valor real da comparação, **se ela for verdadeira**, **se tornará falsa e vice versa**.

**Estruturas de controle** são estruturas que tem a capacidade de direcionar o código, esses são os **if**s **else**s, **switch** e **operador ternário**.

Note que para as estruturas de controle, existe uma maneira rapida de fazer estruturas de if:

*If condição ? true: comando; isso permite não precisar fazer um if inteiro*

É de boa pratica usar o **switch apenas para valores exatos** e if para valores booleanas, evitar o **uso do valor default do switch** para“casos genéricos”, evitar o **efeito flecha** com ifs escadas de if e sempre usar o comando acima para ifs simples.

Estruturas de **repetição** como o **for**, **while** e **do while** que repetem um determinado trecho do código para formar algo.

Estruturas de **interrupção** como **break**, **continue** e **return** servem para interromper o código ou continua-lo de formas diferentes.

**Blocos** se referem a grupos de 0 ou mais códigos que trabalham em conjunção para executar uma operação. Podem ser dos tipos tipos: blocos basicamente tudo aquilo dentro de um { }.

**Locais**: dentro de métodos.

**Estáticos**: dentro de classes.

**Instância**: dentro de classes (usados em POO).