

```
--i1; // pre-decrement, decrement by one before returning value

// ----- COMPARISON OPERATOR -----

i1 == i2; // complete equality operator
i1 != i2; // complete inequality operator
i1 > i2; // comparison operator
i1 < i2; // comparison operator
i1 >= i2; // comparison operator
i1 <= i2; // comparison operator

// ----- LOGICAL OPERATOR -----
boolean a = true;
boolean b = false;
a && b; // logical AND operator
a || b; // logical OR operator
```

Control structures

```
// ----- IF ELSE IF ELSE -----

int j = 10;
if (j == 10) {
    Console.WriteLine("I am invincible");
} else if (j > 10) {
    Console.WriteLine("Neato");
} else {
    Console.WriteLine("Pronto");
}

// ----- TERNARY OPERATOR -----
// <condition> ? <run if true> : <run if false>

int toCompare = 17;
string isTrue = toCompare == 17 ? "yes it is 17" : "no it is not 17";

// ----- WHILE LOOP -----
// while and do while loops are both available
// continue skips the current iteration
// break breaks from the current loop completely

int fooWhile = 0;
while (fooWhile < 100) {
    fooWhile++;
}

int fooDowhile = 0;
do {
    fooDowhile++;
} while (fooDowhile < 100);
```