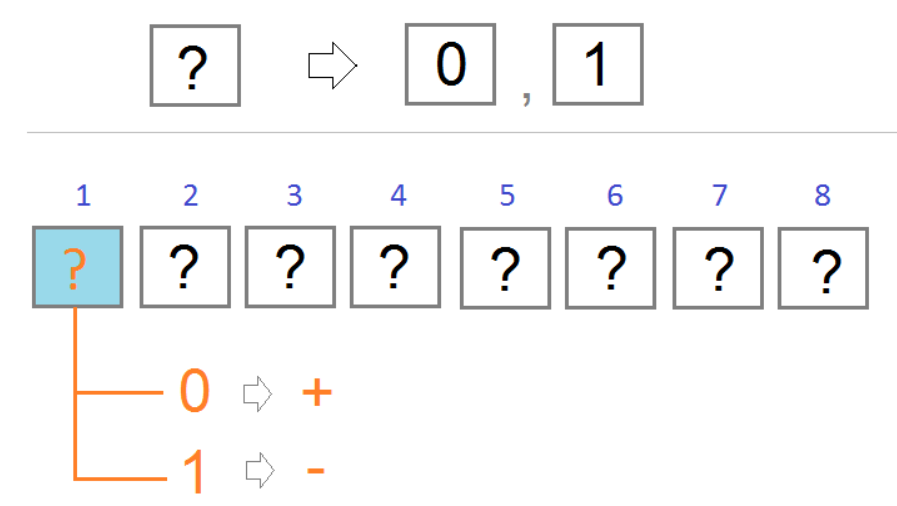
2 – Byte

* 1 byte là 8 bit nằm kề nhau trên vùng bộ nhớ của máy tính. Mỗi bit là một số nhị phân 0 hoặc 1. Java lấy tên "byte" để đặt cho một kiểu số nguyên (integer) có phạm vi nhỏ (Kích thước 1 byte).



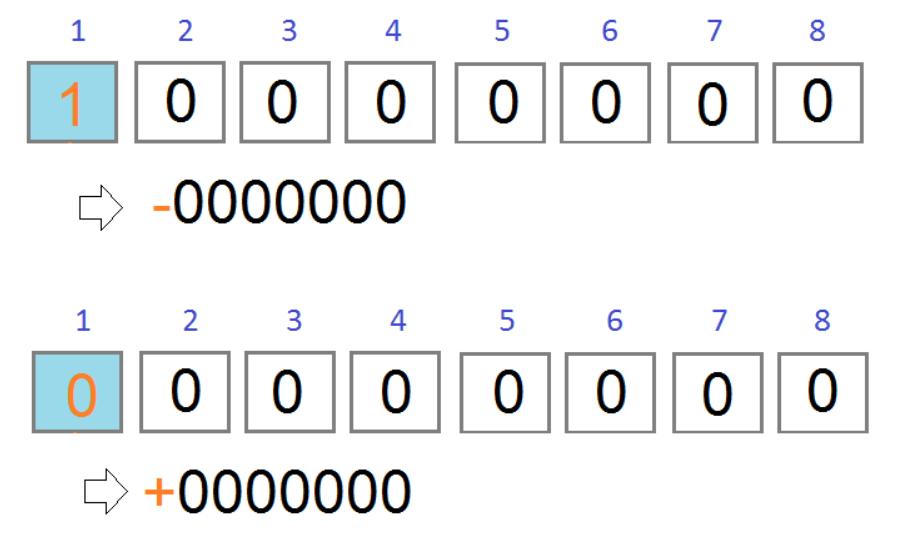
bit đầu tiên trong dẫy 8 bit có giá trị là 0 hoặc 1.

* Nếu là 0 Java coi nó là dấu + (Đại diện cho số dương)
* Nếu là 1 Java coi nó là dấu - (Đại diện cho số âm)

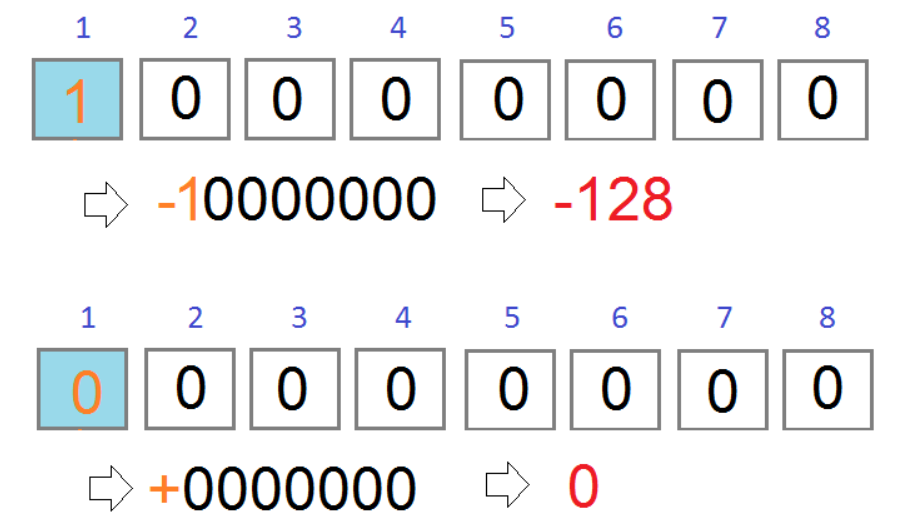
Với 7 bit phía sau bạn có thể biểu diễn được một số từ 0 tới 127. Từ đây bạn suy ra kiểu byte trong Java có phạm vi [-127, 127].  
*Nhưng khoan đã nó phải là [-128, 127], tại sao vậy?*

Tại sao số nhỏ nhất của kiểu byte trong Java là -128?

Nếu theo quy tắc **bit**đầu tiên có giá trị 0 tương đương dấu +, giá trị 1 tương đương dấu -, thì chúng ta có tới 2 cách để biểu diễn số 0 (Xem hình minh họa).



Vì vậy dẫy nhị phân "1 0 0 0 0 0 0" nên được coi là số nhỏ nhất của kiểu dữ liệu byte trong Java, nó biểu diễn số -128.



## 3 – Boolean

boolean là kiểu dữ liệu đơn giản nhất, nó có kích thước 1 bit. Nó có 2 giá trị là true và false.

Giá trị mặc định cho kiểu boolean là false.

// Khai báo một biến có tên 'valid', kiểu boolean.

// Nó sẽ có giá trị là false (Theo mặc định).

boolean valid;

// Khai báo một biến có tên 'active', kiểu boolean.

// Giá trị là true.

## boolean active = true;

## 4- char

Mặc dù "char" là 4 ký tự đầu tiên của từ "Character" (Ký tự), nhưng kiểu char trong Java sử dụng để lưu trữ các số nguyên không âm (Non-negative integers) có kích thước 2 byte. Nó cũng được sử dụng để đại diện cho một ký tự Unicode, vì bản chất mỗi ký đều tương ứng với một con số cụ thể (Con số này được hiểu như là mã của ký tự).

Vì char là kiểu số nguyên không âm, kích thước 2 byte, nên phạm vi của nó là [0, 2\*16-1]  ( [0, 65535] ).

Khi char được hiểu như một ký tự Unicode, ký tự nhỏ nhất là '\u0000' (Mã 0), và ký tự lớn nhất là '\uffff' (Mã 65535).

Ký tự Unicode là gì?

* TODO

## 5- short

short là kiểu dữ liệu để biểu diễn một số nguyên 2 byte (16 bit), bao gồm cả số nguyên âm.

* Giá trị nhỏ nhất là -32,768 (-2^15) và giá trị lớn nhất là 32,767 (2^15 -1).
* Giá trị mặc định là 0.

\*\* Chú ý: Xem thêm giải thích về quy tắc để xác định số dương hay âm tại mục kiểu dữ liệu byte.

## 6- int

Kiểu dữ liệu int được sử dụng để biểu diễn một số nguyên có kích thước 4 byte (32 bit).

* Giá trị nhỏ nhất: -2,147,483,648 (-2^31)
* Giá trị lớn nhất: 2,147,483,647 (2^31 -1)
* Giá trị mặc định 0.

Trong Java, kiểu dữ liệu int được coi là kiểu dữ liệu mặc định cho các số nguyên. Như vậy nếu bạn viết số 100, Java sẽ tạo ra một vùng bộ nhớ 4 byte để lưu trữ. Còn nếu bạn muốn Java tạo ra một vùng bộ nhớ 8 byte để lưu trữ số 100 bạn phải ghi là 100L. (long là kiểu số nguyên 8 byte, được giới thiệu ở mục bên dưới).

int a = 100;

## 7- long

Kiểu dữ liệu long sử dụng để biểu diễn các số nguyên có kích thước lên đến 8 byte (64 bit).

* Giá trị nhỏ nhất là -9,223,372,036,854,775,808.(-2^63)
* Giá trị lớn nhất là 9,223,372,036,854,775,807. (2^63 -1)
* Kiểu này được sử dụng khi cần một dải giá trị rộng hơn int.
* Giá trị mặc định là 0L.

// Bạn cần thêm ký tự 'L' ngay sau số 100

// để Java tạo ra một vùng bộ nhớ 8 byte để lưu trữ số 100

long a = 100L;

## 8- float

Kiểu dữ liệu float được sử dụng để biểu diễn một số thực (real number) có kích thước 4 byte (32 bit).

* Giá trị nhỏ nhất: -3.4028235 x 10^38
* Giá trị lớn nhất: 3.4028235 x 10^38
* Giá trị mặc định 0.0f

Ví dụ:

float value = 2.5f;

## 9- double

Kiểu dữ liệu double được sử dụng để biểu diễn một số thực (real number) có kích thước 8 byte (64 bit). Nó là kiểu mặc định cho các kiểu số thực.

* Giá trị nhỏ nhất: -1.7976931348623157 x 10^308
* Giá trị lớn nhất:  1.7976931348623157 x 10^308
* Giá trị mặc định 0.0d

Ví dụ:

double a = 2.5d;

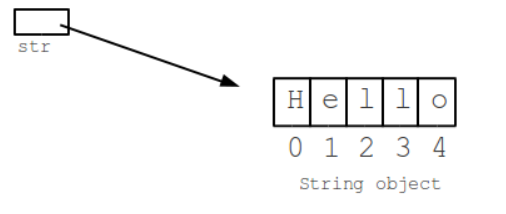
// Vì double là kiểu mặc định cho kiểu số thực,

// Bạn có thể viết ngắn gọn hơn:

double b = 2.5;

## 10- Kiểu tham chiếu (Reference Type)

Trong **Java**một kiểu dữ liệu được tạo ra bởi sự kết hợp các kiểu nguyên thủy với nhau được gọi là kiểu tham chiếu (Reference type). Kiểu tham chiếu thường được sử dụng nhất đó là **String**, nó là sự kết hợp của các ký tự.



Các kiểu dữ liệu tham chiếu được tạo ra dựa trên một lớp. Lớp (class) giống như một bản thiết kế (blueprint) để định nghĩa một kiểu tham chiếu.

**class** **Address** {

String address;

String cityName;

}

**class** **Student** {

String fullName;

int age;

Address address;

}