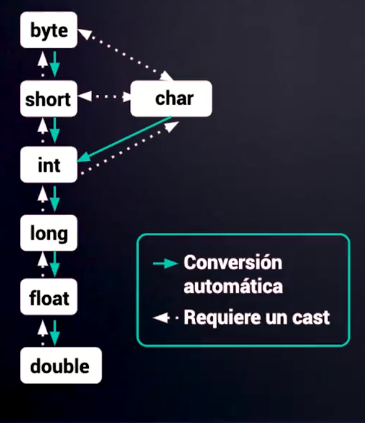
**Casteo entre tipos de datos**



**Casteo entre tipos de datos**

Java nos ayuda a realizar casteo automático de los tipos de datos más chicos a otros más grandes.

Sin embargo, en algunos casos vamos a necesitar realizar un cast manualmente, así como aprendimos en la clase anterior ((dataType) variableOperación).

Por ejemplo: supongamos que declaramos dos variables a y b de tipo int y una variable c de tipo double que es igual a la división de las primeras dos variables.

En este caso, aunque definimos que el tipo de dato de c es double, Java automáticamente convertirá el resultado de la división a tipo int, ya que este es el tipo de datos de las dos variables que dividimos, pero seguirá respetando que la variable c es de tipo double y añadirá un decimal al final (.0).

Esto significa que muchas de nuestras operaciones pueden verse afectadas. Por ejemplo:

**int** a = 30;

**int** b = 12;

**double** c = a / b;

System.out.println(c); // 2.0 (??)

En este caso, ya que Java convierte nuestras variables automáticamente, debemos indicarle a nuestra variable c (de tipo double) que debe hacer cast de su valor para que Java no altere los valores de las variables y el resultado de la operación sea correcto:

**int** a = 30;

**int** b = 12;

**double** c = (**double**) a / b;

System.out.println(c); // 2.5

Es decir, como a y b son de tipo int, el resultado de una operación entre ambas variables será de tipo int, por lo que no tendrá decimales, pero si guardamos el resultado de esta división en una variable de tipo double añadiremos un .0.

Esto podemos solucionarlo si indicamos que además de que la variable c es de tipo double, el valor de esta variable también debe ser de tipo double. Esto significa que Java ejecutará la división entre a y b como si fueran de tipo double, por lo que tendrán decimales a pesar de haber sido definidas inicialmente como números enteros.