

2.B. Tipos de datos.

1. Los tipos de datos.

En los lenguajes fuertemente tipados, como es el caso de Java, a todo dato (constante, variable o expresión) le corresponde un tipo que es conocido antes de que se ejecute el programa.

El tipo limita el valor de la variable o expresión, las operaciones que se pueden hacer sobre esos valores, y el significado de esas operaciones. Esto es así porque **un tipo de dato no es más que una especificación de los valores que son válidos para esa variable, y de las operaciones que se pueden realizar con ellos.**

Debido a que el tipo de dato de una variable se conoce durante la revisión que hace el compilador para detectar errores, o sea en tiempo de compilación, esta labor es mucho más fácil, ya que no hay que esperar a que se ejecute el programa para saber qué valores va a contener esa variable. Esto se consigue con un control muy exhaustivo de los tipos de datos que se utilizan, lo cual tiene sus ventajas e inconvenientes, por ejemplo cuando se intenta asignar un valor de un tipo, a una variable de otro tipo. Sin embargo, en Java, puede haber conversión entre ciertos tipos de datos, como veremos posteriormente.

Ahora no es el momento de entrar en detalle sobre la conversión de tipos, pero sí debemos conocer con exactitud de qué tipos de datos dispone el lenguaje Java. Ya hemos visto que las variables, según la información que contienen, se pueden dividir en variables de tipos primitivos y variables referencia. Pero ¿qué es un tipo de dato primitivo? ¿Y un tipo referencia? Esto es lo que vamos a ver a continuación. **Los tipos de datos en Java se dividen principalmente en dos categorías:**

- **Tipos de datos sencillos o primitivos.** Representan valores simples que vienen **predefinidos en el lenguaje**; contienen valores únicos, como por ejemplo un **carácter o un número.**
- **Tipos de datos referencia.** Se definen con un nombre o **referencia (puntero) que contiene la dirección en memoria de un valor o grupo de valores.** Dentro de este tipo tenemos por ejemplo los vectores o arrays, que son una serie de elementos del mismo tipo, o las clases, que son los modelos o plantillas a partir de los cuales se crean los objetos.

En el siguiente apartado vamos a ver con detalle los diferentes tipos de datos que se engloban dentro de estas dos categorías.

Autoevaluación

El tipado fuerte de datos supone que:

- ☐ A todo dato le corresponde un tipo que es conocido antes de que se ejecute el programa.
- ☐ El lenguaje hace un control muy exhaustivo de los tipos de datos.
- ☐ El compilador puede optimizar mejor el tratamiento de los tipos de datos.
- ☐ **Todas las anteriores son correctas.**