

1.
 - a. Statically typed languages – Programming languages which check the data types at compile-time are statically typed languages
 - b. Dynamically typed languages – Programming languages which check the data types at run-time are dynamically typed languages
 - c. Strongly Typed Languages – Programming languages of which the data types limit the values that a variable can hold or that an expression can produce and limit the operations supported on those values, and determine the meaning of the operations are called strongly typed languages.
 - d. Loosely Typed Languages – Programming languages of which that do not check the data types strictly, or languages that makes it simple to define variables with various data types and that does not hold limits of values that the data types can hold are called loosely typed languages.

2. Case Sensitive – If a programming language has the ability to distinguish between upper and lower case versions of a letter in the language's character set, that is a case sensitive programming language.

Eg:- numberOne and numberone are identified as two different identifiers.

Case Insensitive – If a programming language has the ability to ignore the difference between upper and lower case versions of a letter in the language's character set or if they are identified as the same character, that is a case insensitive programming language.

Eg:- numberOne and numberone are identified as the same identifier.

Case sensitive-insensitive – Programming languages which act as case sensitive in some instances and as case insensitive in some instances

Java is a case sensitive programming language.

3. Identity Conversion is the conversion from a data type to that same data type, which is permitted for any data type.

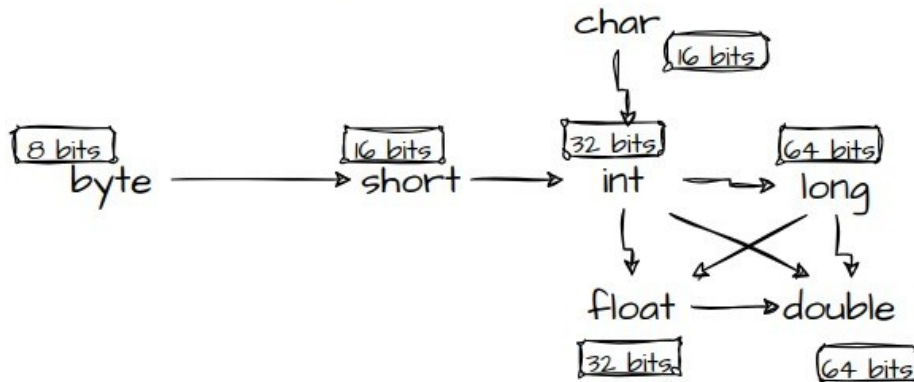
Eg:- int num1 = 10;
 long num2 = 558673932L;

4. Widening Primitive Conversion – Generally, data types with lower bit size will be automatically converted to higher bit sized data types. That is widening primitive conversion.

byte —> short —> int —> long
float —> double

Furthermore, widening primitive conversion will occur from int to float and double, long to float and double also. These conversions do not depend on bit size. Because the bit structures of these converting types are different.

Widening primitive conversion



5. Compile-time constant – Constants which the compiler can identify at compile-time.

Eg:- `final int MY_CONST = 10;`

Run-time constant – Constants which the JVM identifies only at run-time.

Eg:- `final int MY_CONST2 = 10 * (int)Math.random();`

6. Conversion of higher bit sized data type to the lower bit sized data type is called Narrowing Primitive conversions. There are two types of Narrowing Primitive Conversions, explicit and implicit. Explicit conversion is also called casting which do not occur automatically at compile time.

Implicit or automatic Narrowing Primitive Conversion occur automatically if four conditions fulfilled.

- i. Conversion should be in a assignment context
 - ii. It occur only among byte, short, char, int data types from higher bit size to a lower bit size data type.
 - iii. Assigning value should be a compile-time constant
 - iv. The assigning value should be in the range of the type
7. Conversions from long data type to float data type do not depend on bit size. Because the bit structures of these converting types are different. Long type which is a integer type represent the value by 2's complement method while float type which is a floating point type represent the value by 32-bit IEEE 754 method. In IEEE 754 method, the number is taken as fraction and exponent and represent in 31 bits. Therefore in that method more large numbers can be hold in 31 bits than in 64 bit 2's complement method. But the accuracy will be less.

8. Int as the default data type of integers

Character encoding system used by java is UTF-16 which can use maximum of 4 bytes or 32 bits to encode characters. So, from a one write it can accommodate character with maximum of 32 bits, which is equal to the bit size of int data type. Since the long has 64 bits, it treated values as two writes. So, the performance is lower than that of int. So, int has been made the default data type of integers to accommodate most higher values with highest performance.

Double as the default data type of floating point numbers

But in floating numbers, the aim is to store maximum range of values, not the performance or the accuracy. In fact, double which has 64 bit memory space is more accurate than float which has 32 bit memory space and double can accommodate higher range of values. Therefore, double has been made the default data type of floating point numbers.

9. Ranges of byte, short, char, and int are lesser compared to that of long, float or double. Therefore, when implicit narrowing primitive conversion take place among byte, short, char and int, data loss will be minimal and precision can be preserved. When narrowing Primitive Conversion occur with long, float or double, data will be loss and so the precision will be loss. So, the Conversion has to be done explicitly.
10. The Conversion that engage both widening primitive conversion and narrowing primitive conversions is called widening and narrowing primitive conversion. This conversion takes place only from byte to char.

First, the `byte` is converted to an `int` via widening primitive conversion and then the resulting `int` is converted to a `char` by narrowing primitive conversion.

But conversion from short to char is not take place as a widening and narrowing primitive conversion. The short value directly converted to char, without widening conversion to int and then narrowing conversion to char.