

1) Java is a 'Statically Typed Language','Dynamically Typed Language' and also 'Strongly Typed Language',

- Statically Typed means every variable and every expression should have a type that is known at compile time.
- Dynamically Typed means type checking will do at the run time.
- Strongly Typed Language means Strong static typing helps detect errors at compile time.

2) Case-sensitive means that the computer program only matches values with the same case (lower/upper).

Case-insensitive means the program ignores case and matches values regardless of their lower or upper case letters.

Java, like most programming languages, is case sensitive. Even the slightest difference in naming indicates different objects (count does not equal Count). In order to be consistent, programmers follow naming conventions. For example, variables are lowercase (car) and classes are uppercase (Car)

3) Identity Conversion means a conversion from a type to that same type is permitted for any type.

First, it is always permitted for an expression to have the desired type to begin with, thus allowing the simply stated rule that every expression is subject to conversion, if only a trivial identity conversion.

Second, It implies that it is permitted for a program to include redundant cast operators for the sake of clarity.

4) When we need to convert from a primitive that is simpler or smaller than the destination type, we don't have to use any special notation for that. During widening conversion, the smaller primitive value is placed over a larger container, which means that all the extra space, on the left of the value, is filled with zeros. This may also be used to go from the integer group to the floating point.

Byte → short → int → long

float → double

5) A compile-time constant is computed at the time the code is compiled, While a run-time constant can only be computed while the application is running.

A compile-time constant will have the same value each time an application runs, while a run-time constant may change each time.

6) Narrowing conversion is needed when you convert from a larger size type to a smaller size. This is for incompatible data types, wherein automatic conversions cannot be done.

But it also can happen

when compiler see the value which is going to assign in the assignment context is constant & within the byte range.

Double -> float -> long -> int -> short →Byte

7) long data type which is 64 bits can be assigned into float data type that's only 32 bits. because float bit structure is designed to store large numbers regardless it's accuracy.

8)