



# CORE JAVA – TYPE CASTING

*Lecture Notes*

A COMPLETE GUIDE FOR  
BEGINNERS

**Softech Solutions Inc.**

[www.technosolution.tech](http://www.technosolution.tech)

[saney.alam@techsolutions.tech](mailto:saney.alam@techsolutions.tech)

# Core Java – Type Casting

<i>Table of Contents</i>	<i>Page</i>
<i>Type Casting</i>	<i>02</i>
<i>Implicit Conversions</i>	<i>02</i>
<i>Explicit Conversions</i>	<i>02</i>

## Type Casting

In this lesson we will discuss conversions among different primitive types i.e. char, byte, short, int, long, float, and double.

Booleans cannot be converted to other types. For the other primitive types (char, byte, short, int, long, float, and double), there are two kinds of conversion: **implicit** and **explicit**.

### Implicit Conversions

An implicit conversion means that a value of one type is changed to a value of another type without any special directive from the programmer. A char can be implicitly converted to an int, a long, a float, or a double. For example, the following will compile without error:

```
char c = 'a';  
int k = c;  
long x = c;  
float y = c;  
double d = c;
```

For the other (numeric) primitive types, the basic rule is that implicit conversions can be done from one type to another if the range of values of the first type is a subset of the range of values of the second type. For example, a byte can be converted to a short, int, long or float; a short can be converted to an int, long, float, or double, etc.

### Explicit Conversions:

Explicit conversions are done via **casting**. The name of the type to which you want a value converted is given, in parentheses, in front of the value. For example, the following code casts a value of type double to a value of type int, and a value of type double to a value of type short:

```
double d = 5.6;
```

```
int k = (int)d;  
short s = (short)(d * 2.0);
```

Casting can be used to convert among any of the primitive types except boolean. Please note that casting can lose information; for example, floating-point values are truncated when they are cast to integers (e.g. the value of k in the code fragment given above is 5).