



Type Conversion





Type conversion in java

Type conversion in Java, also known as type casting, is the process of changing the data type of a variable's value from one type to another. Java supports two types of type conversions:

Implicit Type Conversion (Widening):

- This type of conversion is performed automatically by the Java compiler when it is safe to do so.
- It typically occurs when you assign a value of a smaller data type to a variable of a larger data type.
- For example, assigning an int to a double or a byte to an int.



```
int intValue = 42;  
double doubleValue = intValue; // Implicit conversion from int to  
double
```

Explicit Type Conversion (Narrowing):

Explicit type conversion, also known as casting, is performed when you want to convert a value from a larger data type to a smaller data type.

This conversion may result in loss of data or precision, so you need to explicitly tell the compiler that you are aware of the potential issues.

You use parentheses and specify the target data type to perform explicit type conversion.



```
double doubleValue = 3.14;  
int intValue = (int) doubleValue; // Explicit conversion (casting)  
from double to int
```

Note: When you perform explicit type conversion from a larger data type to a smaller data type, be cautious about potential loss of information. For example, if you cast a double to an int, the fractional part will be truncated