Floating Point Data Types

Java's Data Types For Floating Point Numbers
float **double** *The double is Java's default type for any decimal or real number*

Default output for numeric data types

In this slide, we show default output, as it would be in JShell or by using System.out.print, for both whole and real numbers.

Whole Number Examples

Literal Value	Default Output
5	5
500_000_000_000L	500000000000

Floating Point Examples

Literal Value	Default Output
5	5.0
5.000000	5.0
5f	5.0
5d	5.0
5e1	50.0
5_000_000.0	5000000.0
50_000_000.0	5.0E7

You can see from this table, that there are more ways to express a decimal real number literal, than a whole number, with the use of the 'F' or 'D' suffix, as well as using Java's scientific notation in the literal value.

Default output for numeric data types

Another interesting difference is that, for whole numbers, the output is never in scientific notation, but for real numbers, it could be, as you can see.

Whole Number Examples

Literal Value	Default Output
5	5
500_000_000_000L	500000000000

Floating Point Examples

Literal Value	Default Output
5	5.0
5.000000	5.0
5f	5.0
5d	5.0
5e1	50.0
5_000_000.0	5000000.0
50_000_000.0	5.0E7

Why is the double a better choice in most circumstances?

Why should we choose double?

First, it's actually faster to process on many modern computers.

That's because computers have, at the chip level, the functionality to actually deal with these double numbers faster than the equivalent float.

Second, the Java libraries that we'll get into later in the course, particularly math functions, are often written to process doubles and not floats, and to return the result as a double.

The creators of Java selected the double because it's more precise, and it can handle a larger range of numbers.

Challenge

The objective of this challenge, is to convert a given number of pounds to kilograms.

STEPS:

1. Create a variable with the appropriate type, to store the number of pounds that we want to convert into kilograms.
2. Calculate kilograms, using the variable above, and store the result in a 2nd appropriately typed variable.
3. Print the result.

Don't forget to use the conversion formula shown here:

1 pound is equal to 0.45359237 of a kilogram.

Floating Point Number Precision Tips

In general, float and double are great for general floating point operations.

But neither should be used when precise calculations are required – this is due to a limitation with how floating point numbers are stored, and not a Java problem as such.

Java has a class called BigDecimal that overcomes this.