

# How can I format floating point numbers sensibly?

## Question

In java how do I print out a "table" of numbers with all of the decimal points lined up, *i.e.* fixed length fields with a fixed number of digits after the decimal point. I have played around with DecimalFormat but I can't get leading blanks, only leading zeros.

## Answer

Java now includes "printf" like functionality very similar to C. The "g" format seems particularly useful for printing floating point numbers. Use "e" or "f" for fixed point. For example:

### Main.java

```
public class Main
{
    public static void main(String[] args)
    {
        double[] numbers = { 1.2345, 2.5678, 10.334, 100.12, 9.99999, 0.0000000001 };

        System.out.printf("\n\n%10s %10s\n", "Number", "Number");
        for (double number : numbers)
        {
            System.out.printf("%10.5g %10.5g\n", number, number);
        }
        System.out.printf("\n\n%10s %10s\n", "Number", "Number");
        for (double number : numbers)
        {
            System.out.printf("%10.5e %10.5e\n", number, number);
        }
        System.out.printf("\n\n%10s %10s\n", "Number", "Number");
        for (double number : numbers)
        {
            System.out.printf("%10.5f %10.5f\n", number, number);
        }
    }
}
```

produces

Number	Number
1.2345	1.2345
2.5678	2.5678
10.334	10.334
100.12	100.12
10.000	10.000
1.0000e-10	1.0000e-10

Number	Number
1.23450e+00	1.23450e+00
2.56780e+00	2.56780e+00
1.03340e+01	1.03340e+01
1.00120e+02	1.00120e+02
9.99999e+00	9.99999e+00
1.00000e-10	1.00000e-10

Number	Number
1.23450	1.23450
2.56780	2.56780
10.33400	10.33400
100.12000	100.12000
9.99999	9.99999
0.00000	0.00000

More info on formatting is available here:

<http://java.sun.com/javase/6/docs/api/java/util/Formatter.html>