

Mr G's Java Jive

#12: The Math Class

Even though Java provides us with the five of the standard math operators and an extra one, there's more to math than that. This really short handout shows you how to get to the other stuff.

Who Ya Gonna Call?

You're writing a program that needs the value of π . Sure, you could define it yourself as **3.14159**, but isn't there an easier way? How about needing to get the **absolute value** of a number? Isn't there some way to do this without having to write a little **if** statement? And raising numbers to a power - having to write **num*num** in order to get **num²** isn't all that bad, but what happens when you want to do **num¹²**? Let's not forget all those great trig functions. Isn't there an easy way to get them?

Yes, there is! All you have to do is call the **Math** class.

The Reader's Digest Version

As with **DecimalFormat**, I'm not going to suggest that you take the time right now to check out everything in the **Math** class at <http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Math.html>, instead, I'll give you a quick list of the items there that you're most likely to want to use at this point.

Item	Usage	Notes
E	E=Math.E;	Gives the value of e (roughly 2.71828)
PI	PI=Math.PI;	Gives the value of π (roughly 3.14159)
abs(value)	num=Math.abs(value);	Gives the absolute value of a number
cbrt(value)	num=Math.cbrt(value);	Give the cube root of a number
cos(value)	num=Math.cos(value)	Gives the cosine of an angle in radians
exp(value)	num=Math.exp(value)	Gives e^{value}
log(value)	num=Math.log(value)	Gives the natural log of a number
log10(value)	num=Math.log10(value)	Gives the base 10 log of a number
pow(value1, value2)	num=Math.pow(value1, value2)	Gives value1^{value2}
sin(value)	num=Math.sin(value)	Gives the sine of a value in radians
sqrt(value)	num=Math.sqrt(value)	Gives the square root of a value
tan(value)	num=Math.tan(value)	Gives the tangent of a value in radians
toDegrees(value)	num=Math.toDegrees(value)	Converts an angle in radians to degrees
toRadians(value)	num=Math.toRadians(value)	Converts an angle in degrees to radians

A Few Little Notes

1. You can use **Math.E** and **Math.PI** either **directly** or **indirectly**. If you use them **directly**, you'll have to use the long names of **Math.E** or **Math.PI** every single time. But if you use them **indirectly**, by defining **local final variables E** and **PI** that equal **Math.E** and **Math.PI** respectively, then you'll only have to refer to the simple names of the local variables.
2. A **value**, in all cases, can be either a **single number**, a **variable holding the number**, or an **expression that works out to a number**.
3. All values here are assumed to be **real** (which, as you remember, includes **integers**). This means that to get the **12th root of 2** (for piano tuning), you could say **num=Math.pow(2,0.08333)**.

Promises Kept

I promised you that this would be a one-sheeter, and I kept my promise. Handout #13, on **Choices and Local Methods**, will be a little longer.

This page intentionally left **almost** blank