

Math on a Sphere

Topic 8: Advanced Math Commands

In this Topic we'll discuss some built-in JavaScript math functions that we can use to make more complicated designs.

One of the most fun built-in commands is the random number generator, which produces random numbers in the same way that rolling a dice does. The random command is an easy way to make more complicated designs. We can generate a random number between 0 and 1 by using

```
Math.random( )
```

This generates a random number with many decimal points, like 0.4302403947804123. So if you're happy with random numbers between 0 and 1, things are pretty simple. But if you desire specific results, like numbers between 4 and 5, or integers between 0 and 10, things get a little more complicate.

When you want to generate a random number, you need tell the computer the set of possible results (outputs). For our dice analogy, this set of results is equivalent to all the different numbers on the faces of a die. As an example, let's use a standard 6-sided dice, numbered from 1 to 6. The default range for the `Math.random` command is 0 to 1, so we need to change this. Our first guess might be that we could just multiply all the randomly-generated numbers by 6. But this would actually produce a range between 0 and 6. ($0*6=0$ and $1*6=6$). So instead, we'd need to do the following:

```
1 + 5*(Math.random( ) )
```

One complicating factor here is the endpoints of the range. (*Note: you don't have to understand the rest of the paragraph, but it explains why the math gets weird when we need to just generate integers.*) The `Math.random()` command produces numbers from the set $[0,1)$, meaning that this is range is inclusive of 0 and exclusive of 1. In plainer English, the random generator will sometimes generate zero exactly, but will never generate 1.0000000. So it could generate 0.999999999, but it'll never generate exactly 1. This starts to be very important when we want to generate integers

So let's look at what we have done so far. We have expanded our range so now we can generate numbers between 1 and 6. But, we're only half-way there. This code would usually produce numbers that aren't integers. So to make only integers we need to also use another built-in math function. There are a few functions we could use, such as `Math.floor`, `Math.ceil`, or `Math.round` (http://www.w3schools.com/jsref/jsref_obj_math.asp). Here, we'll use the `Math.floor` command, which takes a number and rounds it down to the nearest integer. The following is an example of how we could produce random integers between 1 and 6:

```
Math.floor( 1 + 6*(Math.random( ) ) )
```

Last updated on 2013-04-25

Let's look at how this works. When `Math.random()` produces 0, the `Math.floor` command takes this and turns it into the integer 1. When `Math.random()` produces 5.9, the computer calculates `Math.floor(6.9)` which is the integer 6.

What do you want to use the `random` command for? If you need inspiration you can check out the built-in design called "random circles".

In the built-in designs called "flower" and "triangle size" we also use other built-in math functions like `Math.cos` and `Math.sin`. Also, in "flower" you'll see that we can use the term `PI` instead of typing out an abbreviation for π like 3.14159. There are also a lot of other built-in functions we can use. More information on other built-in Java Script math functions can be found on the following websites:

- http://www.w3schools.com/jsref/jsref_obj_math.asp
- <http://www.javascripter.net/faq/mathfunc.htm>