

# OpenEpi Random Number Generator

Random.htm passes though to the user in a convenient form the random numbers produced by the JavaScript Math.random() function. Hence a completely separate analysis of their quality is not necessary. They are merely scaled to fit the range requested and placed in the required number of columns.

According to the literature on the Internet, the JavaScript random number generator is adequate for many purposes, and at least one commercial product built around it is offered (<http://www.randomizer.org/about.htm>). It is, however, a pseudorandom number generator, and not suitable for exacting applications such as cryptography. I could not find any systematic summary of tests done on the JavaScript Math.random function in various browsers, which would have been important information.

The numbers produced are allowed to repeat, that is, the numbers are "with replacement". It is possible that the number, 1003, may, by chance, occur more than once in the numbers generated for your purpose, for example. If you wish to have the effect of, say, 2000 UNIQUE numbers, you can do this in at least two ways. Supposing that you are using random numbers to pull hospital records for a study. If you find that one of the numbers points to a chart that has already been pulled, you can simply move on to the next number, essentially discarding the duplicate number. Another possibility is to ask for larger numbers, for example, with six digits, where the possibility of repeating is small (about 1 in a million). Of course, you could sort the numbers and discard duplicates or do a frequency, but these options require other software.

Many articles are available by searching for *...Random Numbers Epidemiology...* in Google. Most do not mention quality of random numbers as being important for ordinary studies, although it becomes so when doing computer simulations with millions of repetitions. Here are some other interesting articles on random numbers.

[http://en.wikipedia.org/wiki/Random\\_number\\_generator\\_attack](http://en.wikipedia.org/wiki/Random_number_generator_attack)

<http://www.fourmilab.ch/hotbits/>

Uner, Eric. Generating random numbers. Embedded Systems Design (05/24/04, 12:00:00 PM EDT)

[http://www.embedded.com/columns/technicalinsights/20900500?\\_requestid=585475](http://www.embedded.com/columns/technicalinsights/20900500?_requestid=585475)