Complete the combination lock challenge below.

Save & Run | 7/9/2024, 1:20:09 PM - 7 of 7 | Download | Show CodeLens | Reformat | Pair?

```
1 //Code by Akshat Garg
2 public class MathChallenge
3 {
4     public static void main(String[] args)
5     {
6         // 1. Use Math.random() to generate 3 integers from 0-40 (not
7         // including 40) and print them out.
8 
9         int a = (int)(40 * Math.random());
10        int b = (int)(40 * Math.random());
11        int c = (int)(40 * Math.random());
12 
13        System.out.println(a);
14        System.out.println(b);
15        System.out.println(c);
16 
17        // 2. Calculate the number of combinations to choose 3 numbers between
18        // 0-40 (not including 40) using Math.pow() and print it out.
19        // For example, Math.pow(10,2) is 10^2 and the number of permutations
20        // to choose 2 numbers between 0-9.
21 
22        int combs = (int)(Math.pow(40,3.0));
23        System.out.println(combs);
24 
25    }
26 }
27 
28 
```

```
2
28
4
64000
```

Result	Expected	Actual	Notes
Pass	2+ lines of output	4 lines of output	Expected output
Pass	true	true	Prints result of 40^3

Pass	Possible answers:
	(int) (Math.random() * 40)
	(int) (40 * Math.random())
		int a = (int)(40 * Math.random());
		int b = (int)(40 * Math.random());
		int c = (int)(40 * Math.random());
			Creates 3 random numbers from 0 to 40 (not inclusive)

Pass	1 or more	1	Calls to Math.pow(...)

You got 4 out of 4 correct. 100.00%

Activity: 2.9.3.1 ActiveCode (challenge2-9-random-math)

Pretty simple exercise, but forgot that pow returns a double value, so had an error I tried to fix.