Transcript: Part 2 -Math Class

>> Now let's continue our discussion of the math class. We've just finished talking about math.random. Now, and of course we've got other methods as well, but since these are static methods, we invoke them with the name of the class rather than an object. So, here's an example. We did this with random, recall we said math.random.

Here's an example of calling Math.cos with 90 as the perimeter. And another example here, we're adding to that ,uh, Math.sqrt (delta). There, there's a, an example in the textbook, called Quadratic, we gonna take a look at that. And essentially, what it does is solve the Quadratic equation here.

And, and it's a secondary polynomial, ax2 + bx + c = 0. So we are solving the values of x that actually make this equation 0. And this is the formula that does that. It's x = -b plus or minus the square root of b square-4ac. And all this divided by 2a.

So we're gonna implement this in the program and see how that works. Recall we're looking for a couple of roots here, since this is a second degree polynomial and the roots are -b + all this stuff and, -b, and then you subtract all this stuff. So we'll, we'll see how that looks in, in just a moment.

Also, don't forget to look at the questions they're in the questions folder. The answers are there as well in, in a separate folder but do look at those. All right, before we actually look at the quadratic, we're gonna do a few of these in interactions, and just give you the flavor of how we do these math methods.

So let's move over to interactions. This is quadratic too. We'll take a look at it in just a moment. But let's move up here, and let's give ourself some room. And let's call a few of these things just, as expression share. So we might say, Math.sqrt, and then we can give it any number in here real, or uh-uh or integer, integer or double, I should say.

So let's take something we know here, the square root of say, 64. We know that ought to come back 8, which it does, that's nice. We might take Math.abs for absolute value. Let's take the absolute value of -123.45. And we know that ought to just come back with that negative sign gone, and it does, especially if you wanna raise something to a power.

That's gonna be Math.pow, P-O-W, so this is exponentiation when you do that. And let's take, let's take 2 to the 6th power. So we say 2,6 and this is gonna give us back 2 to the 6th, which happens to be 64, that's nice. If we took the square root of 64, of course, we will get 8 so that That shows how we do that.

Now let's do, something like, let's find the, math constant pi. There it is, and, if you're using pi to do any arithmetic, you definitely want to use the constant, don't use 3.14, etc. Just use this. This'll be pi, to a, sort of a standard, scale, if you will.

So now, let's, let's, let's compute the, the area of the circle, let's do that. We know that's pi r squared. So, let's say a double, r, that's our radius. Let's let that equal, let's suppose 6, we'll say, 6.0. And now we want area, so let's say double area.

And let's, let that equal, now we need pi r squared. So we know we, pi is, Math.pi, PI there, and then that's times, r squared. So we could say * r *r, that works. Give it a space, not capitalized, though. And that gives us, 113.097 in change there.

Now, if we wanted to do that with, Just using, the pow instead of, as in we wanna raise this to the second power. There we can say Math.PI times, and now we want to do Math.pow, P-O-W, as in power. And now we want to, is pi r squared, so that's r, and it's raised to the second power, OK?

So this should compute area, should be the same answer over there. Let's see what we get. Yeah, well it didn't change. If you want to see it, computed, we'll just get that, rid of that, and take off the semi-colon. Now we got ourselves an expression. And there it is, that way we could also do it just with the r times r, of course.

And we see we get the same answer there. So, ju-just a little bit of practice there using the, the, math methods. And, and again, since these are static, you don't create an instance of the Math class, you just use the name of the class and call the method.

So Math.pi, Math.abs, Math.power, Math.sqrt and so on. All right, so let's look at a little program example. This is Quadradic2, and, again we want to solve that, quadratic equation here, ax square + bx + c = 0. And, just taking a, a quick look back at it. So this is the radical, it's, it's what we call the discriminate.

It can't be zero, or rather, it can't be negative. And so we're gonna actually have an if statement that makes sure that this stuff under the the radical is not negative. If it is negative, it becomes a complex number, and that's OK too, we can handle that. But we're gonna break this up into pieces there in our program so that we can, do those checks.

So first thing we've got is, the user entering values for, a, b, and c, and we're entering them as integers. We could have integer, entered them as, doubles as well, that works of course too. And then here's our discriminate, this is a math.pow(b, 2)- (4 * a * c).

So just think of this as b squared minus 4ac, and this is what's under the radical, as in the square root of. And so we want to make sure that's greater than or equal to zero. If it is, and then we just do the rest of the formula.

So root 1 is going be minus b, plus math, plus the square root of the discriminant divided by 2a. The other root is -b- Math.sqrt(discriminant)/ (2 * a), again taking a quick check back there, that's what we've got. Plus and the other one is minus. And we're dividing by (2 * a) in either case.

And then we've our two routes, we print them out. If it's negative, we have to decide that this got an i on it, as in the square root of -1 is i. And and we compute them that way. So let's just run this and we'll see how this, how this all works out.

I'm just gonna, just flat out run it, I'll end interactions. We're entering a, give ourselves a little room there. And let's enter a 1 down here. Let's enter 1, -4, I'm looking for rational roots here to begin with, and 4. And, that gives us back Root #1 is 2.0 and Root #2 is also 2.0.

So this is actually, x minus 2 times x minus 2. That would be, x squared minus, 4x minus, or plus 4, I should say. Minus 4x plus 4, that's why they would work out. Let's run it again. This time let's give it, we'll just change the order of the user, one, four and minus four.

And there we get also rational roots, but you see they're a little bit messier, we've got that. And if we wanted to see some irrational roots, we could give it, let's just give it 1, 2, 3. That'll make that discriminant negative. And we see what it looks like with the irrational roots.

So play around with this, and, and use, use interactions to, to compute some of these math functions. They're pretty easy to do but, we'll be writing some programs that, that, that use all of this stuff. Just remember that you just use the, the, class name Math.Pi to get the constant, e is also constant.

But if you're calling methods like, power to do exponentiation, it'll be Math.pow.