ICS 110P
M. Bauer

Python Math Module

Python has several pre-built modules which contain functions and constants you can use in your programs. In order to utilize a module, you must import it into Python. You can import it in the Python shell or into source code files. One of the pre-built modules is the math module.

To import the Python math module, type:
import math

Constants

The math module has several defined constants. You can find them at the end of the math module documentation. One of the constants is the value of pi carried out to the maximum available precision allowed in Python. To see the value of pi, type: math.pi you can use the value of pi in calculations by using the expression math.pi in your arithmetic statements.

For example, the circumference of a circle is calculated using the formula $2\pi r$, where r is the radius of the circle. To display the circumference of a circle having a radius 5, type: 2 * math.pi * 5

Functions

The math module has several defined functions. You can find them described at the beginning of the math module documentation. The documentation provides the name of each function, what the function does and the data that you must provide to it. In general, specifics about how python performs function are hidden from you. This is called encapsulation. It is a principle of object-oriented design.

Square Root (sqrt)

One of the functions available in the math module is the sqrt function. It appears in the documentation as math.sqrt(*x*). The *x* in the parenthesis is the data that you have to provide to the function. It is called a parameter. When you provide specific data to the function, it can also be referred to as an argument. The sqrt function displays the square root of the parameter provided to it. To see the square root of 9, type: math.sqrt(9) **or** type: math.sqrt(9.0)

Factorial (factorial)

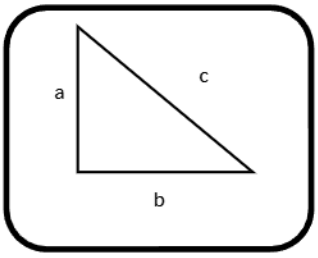
The factorial function displays *x*! for the particular parameter x provided to it. The parameter x must be a positive integer. The value of *x*! is 1 * 2 * … * (*x*-1) * *x*. Note: In mathematics, the expression *n*! is usually used. I substituted *x* for *n* because *x* is name the parameter defined in Python. To display the value of 3!, type: math.factorial(3)

Greatest Common Divisor (gcd)

The gcd function displays the greatest common divisor of two integers. This function has two parameters, so you must provide two pieces of data to use it. Python names the parameters, *a* and *b*. When a function requires multiple parameters, you separate them using one or more comma(s). To display the greatest common division of 56 and 707, type: math.gcd(56, 707)

Power (pow)

The pow function has two parameters, *x* and *y*. It returns x raised to the y power as a real number (float). The function math.pow(*x*,*y*) returns the same value as *x***y*, except *x***y* will return an integer whenever both x and y are integers. To try the pow function, type: math.pow(2,3) then 2**3. Type math.pow(4, 0.5) then 4**0.5.



In mathematics, the Pythagorean Theorem is used to find the lengths of each side of a right triangle, $a^2 + b^2 = c^2$. If a = 3 and b = 4, then $c^2 = 3^2 + 4^2 = 9 + 16 = 25$. Therefore c is the square root of 25 = 5. The parameter to the square root function may be an expression. To find the length of c, given a=3 and b=4, type: math.sqrt(3**2 + 4**2). You can also nest function calls inside of other function calls, type: math.sqrt(math.pow(3,2) + math.pow(4,2))

The sine of an angle is defined as the ratio of the side opposite the angle to the hypotenuse. In the right triangle, the sine of the angle between sides b and c is a/c. The cosine is the ratio of the side adjacent to the angle to the hypotenuse. In the right triangle, the cosine of the angle between sides b and c is b/c. The tangent of an angle is the ration of the side opposite to the angle to the side adjacent to it. In the right triangle, the tangent of the angle between b and c is a/b. Python has the three functions math.sin(*x*), math.cos(*x*) and math.tan(*x*) where *x* is the measure of the angle in radians.

Python also has functions to convert the measurement of an angle between degrees and radians. The function math.degrees(*x*) returns the number of degrees in an angle having the radians measurement *x*. The math.radians(*x*) function returns the number radians in an angle having the degrees measurement *x*.