

Math Module

- `math.ceil(x)` Rounds a number *up* to the nearest integer
- `math.cos(x)` Returns the cosine of a number
- `math.degrees(x)` Converts an angle from radians to degrees
- `math.floor(x)` Rounds a number *down* to the nearest integer
- `math.log(x)` Returns the natural logarithm of a number, or the logarithm of number to base.
- `math.pow(a, b)` Returns the value of *a* to the power of *b*
- `math.radians(x)` Converts a degree into radians
- `math.sin(x)` Returns the sine of a number
- `math.tan(x)` Returns the tangent of a number

Math Constants

- `math.e`
- `math.inf`
- `math.pi`

Random Module

- `seed()`
- `random.randint(a, b)`
- `random.random()`

String Methods

- `word.capitalize()` Converts the first character of word into upper case
- `word.center(n, c)` Centers the string word in n spaces with the optional argument of padding the word with character/string c
- `word.count(str)` Counts the number of times the substring str appears in the string word
- `word.find(str)` Finds the first index of the substring str in the string word. If str is not in word, then it returns -1
- `word.isalnum()` Checks if the string word is alpha-numeric, meaning it only contains letters and numbers
- `word.isalpha()` Check that the string is *only* alphabetic, meaning it only has letters
- `word.isdecimal()` Checks if a string is a decimal
- `word.lower()` Converts the string word into lower case
- `word.upper()` Converts the string word into upper case
- `word.strip()` Removes the white space before and after the word
- `word.replace('a', 'b')` Replaces the letter 'a' in the string word with the letter 'b'

Turtle Methods

- `bob = turtle.Turtle()` Creates a turtle instance and stores it in variable bob
- `bob.forward(distance)` Moves the turtle forward by distance
- `bob.right(angle)` Turn turtle right by angle units
- `bob.left(angle)` Turn turtle left by angle
- `bob.setheading(angle)` Set the orientation of the turtle to angle
 - heading 0 => turtle points right
 - heading 90 => turtle points up
 - heading 180 => turtle points left
 - heading 270 => turtle points down
- `bob.home()` Move the turtle to the origin – the coordinates (0, 0)
- `bob.circle(radius)` Draw a circle with the given radius
- `bob.distance(x, y)` Returns the distance from the turtle to the point (x, y)
- `bob.penup()` Put the pen up, there won't be anything drawn when we move the turtle
- `bob.pendown()` Put the pen down, drawing resumes
- `bob.pencolor(color)` sets the pen color to the string color
- `turtle.exitonclick()` Shut down the turtlegraphics window on mouse click
- `bob.goto(x, y)` Move the turtle to the coordinates (x, y)

List Methods

- `my_list.append(element)` adds an element at the end of the list
- `my_list.copy()` Returns a copy of the list
- `my_list.insert(pos, element)` Adds an element at the specified position pos
- `my_list.pop(pos)` Removes the element at the specified position pos, the default value is -1 (the end of the list)
- `my_list.remove(element)` Removes the first occurrence of the element with the specified value