

# Popular Modules

This lesson highlights some of the most commonly used Python modules.

## WE'LL COVER THE FOLLOWING ^

- `math`
- `heapq`
- `random`

Here are some useful Python modules for beginners. Keep in mind that this is by no means an exhaustive list. For the full Python library module list, check out their [official documentation](#).

## `math` #

The `math` module offers a wide range of mathematical functions such as factorial, trigonometric operations, etc.

```
import math

fact_of_5 = math.factorial(5) # The factorial of 5
print(fact_of_5)

gcd = math.gcd(300, 90) # Greatest common denominator
print(gcd)

log100 = (math.log(10, 100)) # Logarithm of 10 to the base of 100
print(log100)
```



To get all `math` methods for complex numbers, use the `cmath` module instead.

## `heapq` #

The `heapq` module allows us to create the **heap** data structure. A heap is a

binary tree which always stores a special value at the top (root). A **minheap** stores the smallest value at the top and a **maxheap** stores the largest value at the top.

The **pop** method returns the value at the top of the heap.

Python's **heapq** creates a minheap by default.

```
import heapq

heap = [] # Empty heap

# Inserting elements in the heap
heapq.heappush(heap, 10)
heapq.heappush(heap, 70)
heapq.heappush(heap, 5)
heapq.heappush(heap, 35)
heapq.heappush(heap, 50)

# Popping the smallest value
minimum = heapq.heappop(heap)
print(minimum)
```



## random #

The **random** module is used for generating random numbers in Python. There are several methods which allow us to generate different types of random numbers.

The **random()** method generates a random floating-point number between **0** and **1**, whereas **uniform()** returns a floating-point number within a custom range.

```
import random

rand_num = random.random()
print(rand_num)

rand_num_in_range = random.uniform(30, 50) # A random number between 30 and 50
print(rand_num_in_range)

str_list = ['a', 'b', 'c', 'd', 'e']
random.shuffle(str_list) # Randomly shuffle a list
print(str_list)
```





The library is filled with many crafty modules like the ones mentioned above. Be sure to check it out.

---

Apart from the Standard Library, there are thousands of reliable external packages which can be integrated into Python. We'll talk about them in the next lesson.