import random

# Generate random integer in the interval [4,10]

print(random.randrange(4,11))

# Generate random integer in [4,10] with increment of 3

print(random.randrange(4, 11, 3))

# Generate random even numbers between 2 and 24

print(random.randrange(2, 25, 2))

# Generate random even numbers between 1 and 25

print(random.randrange(1, 26, 2))

Output.

#1. #2. #3.

10 7 4

4 10 4

4 12 8

17 3 23

[**Back to top**](http://www.techbeamers.com/using-python-random/#top)

2. Python Random.Randint(Low, High)  Function.

**Purpose-** The **randint()** function is one of many functions which handle random numbers. It has two parameters low and high and generates an integer between low and high (including both).

**Python randint() example.**

# Generate random integers in range 0 through 9.

import random

iter = 0

while iter < 10:

# Get random number in range 0 through 9.

r = random.randint(0, 9)

print(r)

iter += 1

**Output**

4

3

0 Low value

1

5

7

9 High value

3

0

8

[**Back to top**](http://www.techbeamers.com/using-python-random/#top)

Choose A Random Number From The Sequence.

3. Python Random.Choice(Seq) Function.

**Purpose-** The choice() function arbitrarily determines an element from the given sequence.

**Note-** A sequence in Python is the generic term for an ordered set like a list, tuple etc.

**Python choice() example.**

# How to use Python's choice() function to select a item.

import random

# Generate a random string from the list of strings

print(random.choice( ['Python', 'C++', 'Java'] ))

# Generate a random number from the list [-1, 1, 3.5, 7, 15]

print(random.choice([-1, 1, 3.5, 9, 15]))

# Generate a random number from a uniformly distributed tuple

print(random.choice((1.1, -5, 6, 4, 7)))

# Generate a random char from a string

print(random.choice('Learn Python Programming'))

[**Back To Top**](http://www.techbeamers.com/using-python-random/#top)

4. Python Random.Shuffle(List) Function.

**Purpose-** The shuffle() function rearranges the items of a list in place so that they occur in a random order.

For shuffling, it uses the **Fisher-Yates algorithm** which has O(n) complexity. It starts by iterating the last element in the array to the first entry, then swap each entry with an entry at a random index below it.

**Python shuffle() example.**

# How to randomize a list in Python?

from random import shuffle

mylist = [11,21,31,41,51]

shuffle(mylist)

print(mylist)

Output.

#1. [41, 51, 11, 31, 21]

#2. [11, 31, 41, 51, 21]

[**Back To Top**](http://www.techbeamers.com/using-python-random/#top)

5. Python Random.Sample(Collection, Random List Length) Function.

**Purpose-** The sample() function randomly selects N items from a given collection (list, tuple, string, dictionary, set) and returns them as a list.

It works by sampling the items without replacement. It means a single element from the sequence can appear in the resultant list at most once.

**Python sample() example.**

# How to use sample() in Python?

from random import sample

# Select any three chars from a string

print(sample('Python',3))

# Randomly select a tuple of three elements from a base tuple

print(sample((21, 12, -31, 24, 65, 16.3), 3))

# Randomly select a list of three elements from a base list

print(sample([11, 12, 13, 14, -11, -12, -13, -14], 3))

# Randomly select a subset of size three from a given set of numbers

print(sample({110, 120, 130, 140}, 3))

# Randomly select a subset of size three from a given set of strings

print(sample({'Python', 'C++', 'Java', 'Go'}, 3))

**Output.**

['y', 'o', 't']

[21, 12, 24]

[-14, 14, -13]

[140, 110, 130]

['Python', 'Java', 'C++']

[**Back to top**](http://www.techbeamers.com/using-python-random/#top)

Generating Floating Point Random Numbers.

6. Python Random.Random() Function.

**Purpose-** It selects the next random floating point number from the range [0.0, 1.0]. It is a semi-open range as the random function will always return a decimal no. less than its upper bound. However, it may return 0.

**Python random() example.**

# How to generate a floating-point random number in Python?

import random

# Generate a floating-point pseudo-random number between 0 and 1.

print(random())

**Output.**

#1. 0.4968601882231284

#2. 0.831505293496292

[**Back to top**](http://www.techbeamers.com/using-python-random/#top)

7. Python Random.Uniform(Lower, Upper) Function.

**Purpose-** It is an extension of the random() function. In this, you can specify the lower and upper bounds to generate a random number other than the ones between 0 and 1.

**Python uniform() example-1.**

# How to uniform() method to generate floating-point random numbers?

import random

lower = 111; upper = 999

random\_float = random.uniform(lower, upper)

print(random\_float)

**Output.**

#1. 466.63369297787557

#2. 315.3719206118211

**Python uniform() example-2.**

# Generate a floating-point random number with fixed precision

import random

lower = 1.0; upper = 2.0; fixed\_precision = 2

random\_float = random.uniform(lower, upper)

print(round(random\_float, fixed\_precision))

**Output.**

#1. 1.48

#2. 1.69

#3. 1.57