WELCOME

WOMEN WHO CODE

Women Who Code Python

Python Libraries 101

OUR MISSION

Inspiring women to excel in technology careers.

WOMEN WHO CODE

OUR VISION

A world where diverse women are better represented as engineers and leaders in technology.

WOMEN WHO CODE

OUR TARGET

Engineers with two or more years of experience looking for support and resources to strengthen their influence and levelup in their careers.

WOMEN WHO CODE

CODE OF CONDUCT

WWCode is an inclusive community, dedicated to providing an empowering experience for everyone who participates in or supports our community, regardless of gender, gender identity and expression, sexual orientation, ability, physical appearance, body size, race, ethnicity, age, religion, socioeconomic status, caste, creed, political affiliation, or preferred programming language(s).

Our events are intended to inspire women to excel in technology careers, and anyone who is there for this purpose is welcome. We do not tolerate harassment of members in any form. Our **Code of Conduct** applies to all WWCode events and online communities.

Read the full version and access our incident report form at **womenwhocode.com/codeofconduct**

WOMEN WHO
CODE

WOMEN WHO CODE PYTHON

Python Libraries 101:

Python Standard Library

Session # 4

Soumya Vemuri
CSE Student

Shermaine Ang
Incoming EIE Freshman at
Imperial College London

Karen Wong
Programmer at R&D
Company

Meet Your Team!

WOMEN WHO CODE PYTHON

SHERMAINE

Incoming EIE Freshman
WWCode Python Volunteer
https://www.linkedin.com/in/shermaine-ang-1348a21ab/

Today's Agenda

1. Recap: What is the Python Standard Library?
2. Modules we'll be exploring today
   a. random
   b. math
   c. statistics
3. Setup and Requirements
4. Hands-on Coding

WOMEN WHO CODE PYTHON

Python Standard Library

WOMEN WHO CODE PYTHON

What is the Python Standard Library?

- Python offers a lot of built-in functionality through its standard library.

- Built-in modules are written in C and integrated with the Python shell.

- The standard library is a huge collection of utilities, ranging from math utilities to debugging to creating graphical user interfaces. Popularly used modules are os, sys, json, re, random, math urllib, tkinter, datetime

random

random Module

- random is a built-in Python module that implements pseudo-random number generators for various distributions
- **Syntax:**

    import random

random Module: Functions

- random.seed(n) initialises the random number generator.
- Seed function is used to save the state of a random function, so that it can generate same random numbers on multiple executions of the code on the same machine or on different machines (for a specific seed value)
- It makes optimization of codes easy where random numbers are used for testing. Seed function is used to generate same random numbers again and again and simplifies algorithm testing process.

random Module: Functions (contd.)

- **random.randrange(start,stop,step)** returns a random number between the given range.
  - start – Start point of the range. This would be included in the range.
  - stop – Stop point of the range. This would be excluded from the range.
  - step - Steps to be added in a number to decide a random number.
- **random.randint(a,b)** returns a random number between the given range, includes starting term and last term.

random Module: Functions (contd.)

- **random.choice(*sequence*)** returns a randomly selected element from the specified sequence
- The sequence can be a string, a range, a list, a tuple or any other kind of sequence.

random Module: Functions (contd.)

- **random.shuffle(*sequence*)** returns the sequence in a random order
- **random.random()** returns a random float number between 0 and 1
- **random.uniform(a,b)** returns a random float number between two given parameters

math

math Module

- Python has a built-in module that you can use for mathematical tasks.

- The math module has a set of methods and constants.

- Syntax:

  import math

-

math Module: Functions

- math.acos(a) returns the arc cosine of a number in radians
- math.asin(a) returns the arc sine of a number in radians
- math.atan(a) returns the arc tangent of a number in radians
- math.cos(a) returns the cosine of a number
- math.cosh(a) returns the hyperbolic cosine of a number
- math.degrees(a) converts an angle from radians to degrees
- math.radians(a)converts a degree value into radians

math Module: Functions (contd.)

- math.ceil(a) rounds a number up to the nearest integer
- math.floor(a) rounds a number down to the nearest integer

math Module: Functions (contd.)

- **math.fmod(x,y)** returns the remainder of x/y
- **math.fsum(list)** returns the sum of all items in any iterable (tuples, arrays, lists, etc.)
- **math.trunc(a)** returns the truncated integer parts of a number
- **math.exp(x)** returns e raised to the power of x
- **math.fabs(a)** returns the absolute value of a number
- **math.factorial(a)** returns the factorial of a number

statistics

statistics Module

- Python has a built-in module that you can use to calculate mathematical statistics of numeric data.

- **Syntax:**

  import statistics

calendar Module: Functions

- **statistics.mean(list)** calculates the mean (average) of the given data
- **statistics.median(list)** calculates the median (middle value) of the given data
- **statistics.median_high(list)** calculates the high median of the given data
  - If the number of data values is odd, it returns the exact middle value. If the number of data values is even, it returns the larger of the two middle values.

statistics Module: Functions (contd.)

- **statistics.median_low(list)** calculates the low median of the given data
- **statistics.mode(list)** calculates the mode (central tendency) of the given numeric or nominal data
- **statistics.stdev(list)** calculates the standard deviation from a sample of data
- **statistics.variance(list)** calculates the variance from a sample of data

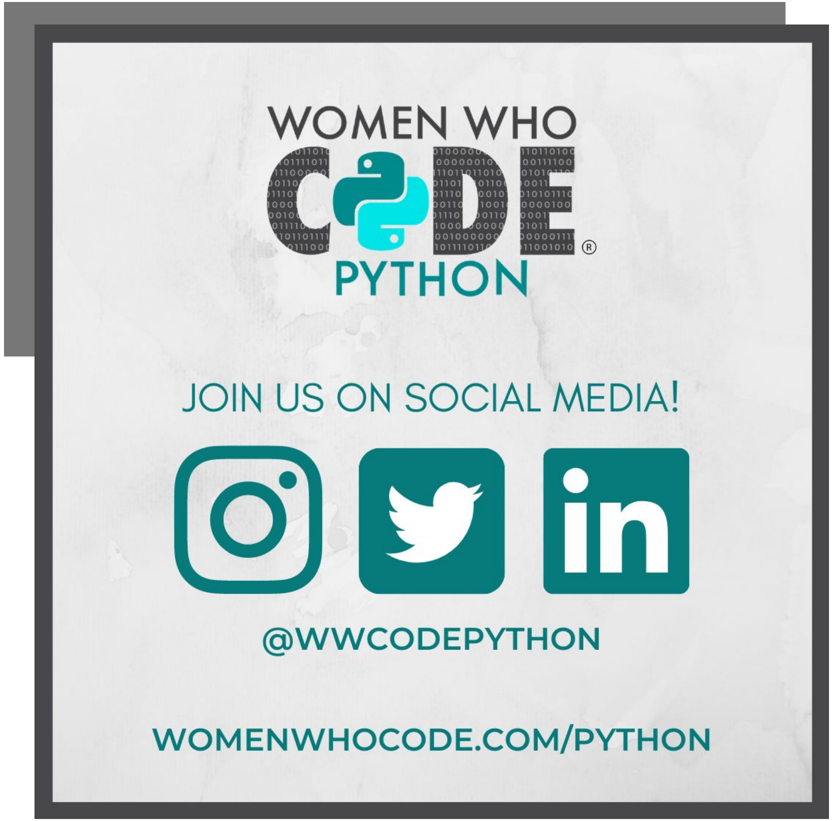
QnA Time!

Let’s Code!

Useful Links

- Link to repository
- Python Documentation
  - random: https://docs.python.org/3/library/random.html
  - math: https://docs.python.org/3/library/math.html
  - statistics: https://docs.python.org/3/library/statistics.html
- Other Resources
  - https://www.w3schools.com/python/module_random.asp
  - https://www.w3schools.com/python/python_math.asp
  - https://www.w3schools.com/python/module_statistics.asp

Stay Connected!



WOMEN WHO CODE PYTHON

JOIN US ON SOCIAL MEDIA!

@WWCODEPYTHON

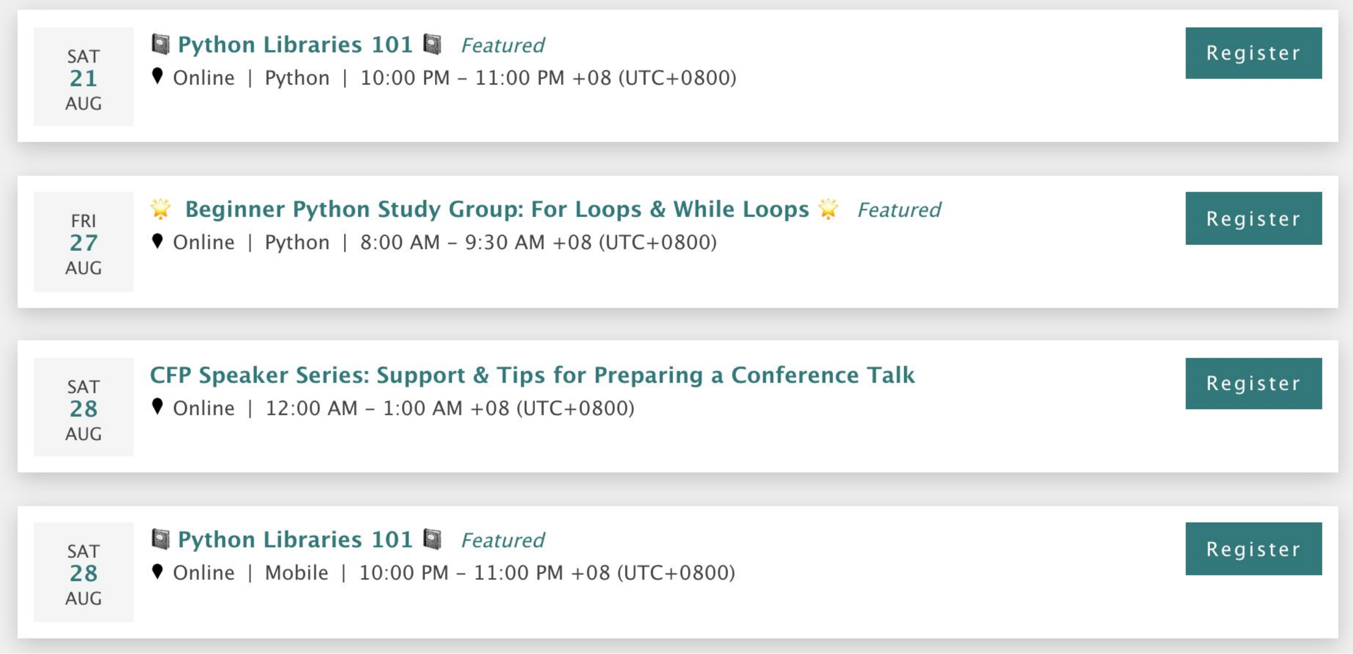
WOMENWHOCODE.COM/PYTHON

Upcoming Events in the Series:

- ➔ Aug. 28 - Recap, QnA, Repeats in Python STL
- ➔ Sept. 04 - Data Visualization with Matplotlib, Seaborn, Plotly

…….and more!

Upcoming Events



SAT
21
AUG

📔 Python Libraries 101 📔 *Featured*

📍 Online | Python | 10:00 PM – 11:00 PM +08 (UTC+0800)

Register

FRI
27
AUG

🌟 Beginner Python Study Group: For Loops & While Loops 🌟 *Featured*

📍 Online | Python | 8:00 AM – 9:30 AM +08 (UTC+0800)

Register

SAT
28
AUG

CFP Speaker Series: Support & Tips for Preparing a Conference Talk

📍 Online | 12:00 AM – 1:00 AM +08 (UTC+0800)

Register

SAT
28
AUG

📔 Python Libraries 101 📔 *Featured*

📍 Online | Mobile | 10:00 PM – 11:00 PM +08 (UTC+0800)

Register

Thank You for Joining!