### 

|  |  |
| --- | --- |
| Trainer(s) Sergii Tishchenko | Target Audience Developers, test automation engineers who wants to know how to program faster with python language |
| Prerequisites To get the most out of this course, you should have good Python language skills, basic Linux command line skills and at least some knowledge of PyCharm IDE  **Required for participants**  • laptop, python 3.7, PyCharm   * Forked and downloaded git repo:   <https://github.com/tishyk/py_gl.git> | Duration 20 hours |
|  |  |

### List of Topics

1. Session 1 (2 h):

* Args and kwargs variables. Packaging and extraction
* Class variables, methods and instance.
* Instance variables and methods
* Built-in variables: \_\_name\_\_, \_\_package\_\_, \_\_file\_\_, \_\_doc\_\_, \_\_class\_\_, \_\_dict\_\_, \_\_mro\_\_, super() method.

2. Session 2 (2 h):

* Magic methods. Classification of magic methods
* Object descriptors
* Built-in decorator functions: property, classmethod, staticmethod,
* \_\_slots\_\_, class instance changings
* Weakrefs explanation.

3. Session 3 (2 h):

* Meta Classes. Creation and scope of usage
* Abstract Class. Abstract methods

4. Session 4 (2 h):

* Creational patterns.

1. Factory Method
2. Abstract Factory
3. Prototype
4. Singleton

5. Session 5 (2 h):

* Structural patterns

1. Adapter
2. Bridge
3. Decorator. Function and class decorators
4. Facade
5. Flyweight
6. Proxy

6. Session 6 (2 h):

* Behavioral patterns

a. Command

b. Observer

c. State

7. Session 7 (2 h):

* Threading.
* GIL
* Lock
* RLock
* Semaphore
* Event
* Condition.

8. Session 8 (2 h):

* Multiprocessing.
* Queue, pipe and the other ways of multiprocessing communication
* Generators as an introduction for asyncio

9. Session 9 (2 h):

* Asyncio

a. Order of execution

b. Concurrency

c. Future states

d. Exceptions handling

10. Session 10 (2 h):

* Network programming

1. Sockets

2. Socket module

3. Servers socket methods

4. Client Socket methods

5. Asyncio and sockets