kod szkolenia: PYTH02 / ENG DL 5d

Advanced programming techniques in Python language

Training recipients

The training is intended for people who want to improve and consolidate, as well as enrich their present knowledge with new functionalities and learn more advanced mechanisms in programming in Python.

Benefits

The training consolidates and significantly extends the knowledge of Python with advanced techniques that will be discussed during the workshops. After completing the training, students will be able to cope with most of the problems that they may encounter while working with the language. Important aspects such as modern practices will be discussed, as well as the popular libraries, asynchronousness and process optimization thanks to the use of available language functionalities.

Training program

1. FUNCTIONAL PROGRAMMING
    - * args & ** kwargs
    - Unpacking arguments and collections
    - Function as parameter (First class citizen)
    - Lambda functions (Anonymous)
    - List comprehensions, dictionary comprehensions, etc … – advanced
    - Nested list and dictionary comprehensions, etc …
    - ITERTOOLS module – overview
    - FUNCTOOLS module – overview
    - Generators and iterators (different ways to define)

- Decorator pattern – create your own decorators (simple decorator)

2. OOP PROGRAMMING – ADVANCED LEVEL
   - Documenting the code
   - Class attributes
   - Multi-inheritance and Method Resolution Order
   - Super method
   - Access to attributes, private attributes
   - Descriptors and defining properties (__get__, __set__, __delete__, __set_name__)
   - __getitem__, __setitem__, __delitem__
   - Implementation of iterability for classes
   - Operator overloading
   - Abstract classes – basic issues
   - Class decorators with arguments
   - Metaclasses – basic issues
3. USEFUL TOOLS
   - Type annotations (hints) – information and application example
   - TIMEIT module – examples (comparison of the execution time of algorithms)
   - LOGGING module – event logging (configuration and application)
   - OS module – revision and additional information
   - SYS module – revision and additional information
   - Interaction with the operating system and file system – SYS and OS modules
4. COLLECTIONS module – an extension of built-in complex types
   - NamedTuple
   - DataClass
   - DefaultDict
   - Deque
   - Counter
5. REGULAR EXPRESSIONS – RE module:
   - Syntax – symbols, structure of regular expressions
   - match & search functions
   - findall & finditer functions
   - Pattern object
   - Match object
   - sub & split functions
   - DOTALL & MULTILINE flags
   - Online tools for creating regular expressions (e.g. regex101 and Pythex)
6. DATA PROCESSING
   - REQUESTS – HTTP protocol module (basic information and requests from Python)
   - BEAUTIFUL SOUP – WebScrapping (XML & HTML) module (example of use in searching hypertext documents)
   - PARAMIKO module – SSH connections (application example)

- JSON, YAML, PICKLE – practical use
- Introduction to Pandas (reading and writing XLS, CSV, etc ...)
- Review of other libraries

7. DATABASE
   - Overview of popular "Connectors" for relational databases – based on connections with MYSQL / PostgreSQL / ORACLE or other selected engine
   - Query handling from Python – basic queries
   - Connection with non-relational databases – on the example of PYMONGO (MongoDB)
   - ORM on the example of SQLAlchemy
8. THREADS AND PROCESSES: THREADING module basic concepts:
   - Launching threads
   - Thread synchronization
   - Rlocks
   - Semaphores
   - GIL (Global Interpreter Lock)
9. MULTIPROCESSING module – basic concepts:
   - Processes, queues, locks
   - Pools
   - Daemons
   - Data exchange between processes
10. ASYNCHRONOUS PYTHON – Introduction and basics
    - ASYNCIO module – overview of the basic issues
    - Coroutines, tasks
    - Streams
    - Subprocesses
    - Queues
    - Exceptions
    - Event Loop
    - Futures
    - Asynchronous (Non-Blocking) HTTP Calls
11. INTRODUCTION TO THE TESTS
    - Unit tests
    -]Introduction to TDD
    - Unittest library
    - Review of other libraries
12. ADDITIONALLY:

- Other libraries selected together with course participants

Expected preparation of the participant

Participation in the training PYTH01 – Basics of Python programming or equivalent knowledge.
Basic knowledge of Linux / Unix / Windows environment.
Knowledge of basic database issues is appreciated.

Czas trwania

5 dni / 35 godzin

Language

- Training: English
- Materials: English