PCAP*:* Certified Associate in Python Programming **(32 Hours)**

This course is the second in a 2-course series that will prepare you for the *PCAP: Certified Associate in Python Programming* certification exam. The course picks up where *PCEP course* leaves off.

Module 1: Modules and Packages

1.1 – Import and use modules and packages

- import variants: import, from import, import as, import *
- advanced qualifying for nested modules
- the dir() function
- the sys.path variable

1.2 – Perform evaluations using the math module

- functions: ceil(), floor(), trunc(), factorial(), hypot(), sqrt()

1.3 – Generate random values using the random module

- functions: random(), seed(), choice(), sample()

1.4 – Discover host platform properties using the platform module

- functions: platform(), machine(), processor(), system(), version(), python_implementation(), python_version_tuple()

1.5 – Create and use user-defined modules and packages

- idea and rationale;
- the __pycache__ directory
- the __name__ variable
- public and private variables
- the __init__.py file
- searching for/through modules/packages
- nested packages vs. directory trees

Module 2 Exceptions

2.1 – Handle errors using Python-defined exceptions

- except, except:-except, except:-else:, except (e1, e2)
- the hierarchy of exceptions
- raise, raise ex
- assert
- event classes
- except E as e
- the arg property

2.2 – Extend the Python exceptions hierarchy with self-defined exceptions

- self-defined exceptions
- defining and using self-defined exceptions

Module 3 Strings

3.1 – Understand machine representation of characters

- encoding standards: ASCII, UNICODE, UTF-8, code points, escape sequences

3.2 – Operate on strings

- functions: ord(), chr()
- indexing, slicing, immutability
- iterating through strings, concatenating, multiplying, comparing (against strings and numbers)
- operators: in, not in

3.3 – Employ built-in string methods

- methods: .isxxx(), .join(), .split(), .sort(), sorted(), .index(), .find(), .rfind()

Module 4 : Object-Oriented Programming

4.1 – Understand the Object-Oriented approach

- Class
- Object property, method
- Encapsulation
- Inheritance
- Superclass
- Subclass
- identifying class components

4.2 – Employ class and object properties

- instance vs. class variables: declarations and initializations
- the __dict__ property (objects vs. classes)
- private components (instances vs. classes)
- name mangling

4.3 – Equip a class with methods

- declaring and using methods
- the self parameter

4.4 – Discover the class structure

- introspection and the hasattr() function (objects vs classes)
- properties: __name__, __module__ , __bases__

4.5 – Build a class hierarchy using inheritance

- single and multiple inheritance
- the isinstance() function
- overriding

- operators:
- not is
- is
- polymorphism
- overriding the __str__() method
- Diamonds

4.6 – Construct and initialize objects

- declaring and invoking constructors

Module 5 Miscellaneous

5.1 – Build complex lists using list comprehension

- list comprehensions: the if operator, nested comprehensions

5.2 – Embed lambda functions into the code

- lambdas: defining and using lambdas
- self-defined functions taking lambdas as arguments
- functions: map(), filter()

5.3 – Define and use closures

- closures: meaning and rationale
- defining and using closures

5.4 – Understand basic Input/Output terminology

- I/O modes
- predefined streams
- handles vs. streams
- text vs. binary modes

5.5 – Perform Input/Output operations

- the open() function
- the errno variable and its values
- functions: close(), .read(), .write(), .readline(), readlines()
- using bytearray as input/output buffer