VLA
ACADEMY

SYLLABUS

PYTHON

www.somosvla.com

Python

General Objective:

To provide students with a solid and practical understanding of the fundamentals of Python programming, as well as intermediate and advanced skills in the language. Through a combination of theoretical and practical lessons, the course aims to equip students to design, develop, debug, execute, and refactor multi-module Python programs. Additionally, the course aims to supply students with skills and knowledge related to real-world problem analysis and modeling using the object-oriented approach in Python. Upon completion of the course and obtaining the PCAP certification, students will be prepared to advance in their professional careers as software developers, data analysts, or in other related roles, as well as to continue their learning in more specialized areas of Python programming.

Content:

- ✓ **Master the fundamental concepts of Python:** By the end of the course, students will be able to understand and apply fundamental Python concepts such as variables, data types, flow control structures (conditionals and loops), and exception handling.
- ✓ **Ability to work with advanced data structures:** Students will acquire skills in handling more complex data structures such as lists, tuples, dictionaries, and sets in Python, as well as understand how to use them effectively in their programs.
- ✓ **Develop object-oriente programming (OOP) skills:** Students will learn the basics of object-oriented programming and be able to apply them in designing and developing Python programs using classes, objects, inheritance, polymorphism, and encapsulation.

- **Create practical Python applications and scripts:** Students will be able to design, develop, and debug multi-module Python programs to solve real-world problems. Additionally, they will be trained to use modules from the Python Standard Library and develop applications with graphical user interfaces using the Tkinter library.

Methodology:

The methodology of the Python - PCAP™ course is based on a comprehensive approach that combines theory and practice to facilitate effective student learning

- **Active Learning:** Students will actively participate in the learning process through a variety of activities such as readings, practical exercises, projects, and class discussions. Exploration and experimentation will be encouraged to reinforce the concepts learned.
- **Practical Focus:** Strong emphasis will be placed on the practical application of learned concepts. Students will work on practical exercises after each lesson to reinforce their understanding and skills in Python. Additionally, larger practical projects will be assigned to allow students to apply what they have learned in real-world situations.
- **Continuous Feedback:** Continuous feedback will be provided to students throughout the course. This will include feedback on practical exercises, projects, and assessments. The aim is to help students identify areas for improvement and reinforce their strengths.
- **Collaborative Learning:** Collaborative learning will be encouraged through group activities, class discussions, and opportunities to work on team projects. This will allow students to share knowledge, solve problems together, and learn from each other.
- **Supplementary Resources:** In addition to in-class lessons, supplementary resources such as additional readings, online tutorials, official Python documentation, and code examples will be provided. These resources will help students delve deeper into topics according to their individual needs.
- **Continuous Assessment:** Student progress will be continuously assessed throughout the course through tests, projects, practical exercises, and exams. This will allow instructors to monitor student progress and adapt teaching as necessary.

Evaluation :

- ✓ **Exam 1:** Lesson 7 (Value: 25%)
- ✓ **Exam 2:** Lesson 14 (Value: 25%)
- ✓ **Exam 3:** Lesson 21 (Valuet: 50%)

Python Syllabus:

| Lesson | Topics | Observation |

- ✓ | **Lesson 1** | History of Python | Print, Time Sleep |
- ✓ | **Lesson 2** | Variables and their types (String, Int, Float, Boolean) Concatenations, When to use comma (,) or plus (+) for concatenation, Implementation of the Type function |
- ✓ | **Lesson 3** | Review of lessons 1 and 2 | Simple Conditional (If-Else) |
- ✓ | **Lesson 4** | Multiple Conditionals (If – Elif – Else) | Nested conditions, Explanation of the term Loop, While Loop |
- ✓ | **Lesson 5** | Lists, Tuples, For Loop | Traversing lists or tuples using For Loop, Creating a counter within For Loop |
- ✓ | **Lesson 6** | Completion of 10 practice exercises |
- ✓ | **Lesson 7** | 6:30-7:30 > Practice session, 7:45-9:30 > Written Part of Exam, 9:30 > Conclusion, Exam 1, Value: 25% |
- ✓ | **Lesson 8** | Completion of 4 practice exercises, Lists, Tuples, While Loop, For Loop, Traversing within an array, Nested loops |
- ✓ | **Lesson 9** | Dictionary: Viewing a dictionary, Adding key-value pairs, Deleting dictionary elements |
- ✓ | **Lesson 10** | To reinforce the topic of dictionaries studied in lesson 9, completion of 6 practice exercises, .split |
- ✓ | **Lesson 11** | Introduction to functions, Built-in functions, Function structure: def name (parameters) |
- ✓ | **Lesson 12** | Completion of Practice4.pdf |
- ✓ | **Lesson 13** | Try-except |
- ✓ | **Lesson 14** | File Handling: Creating a txt file, Modes (w-a-r), Storing Lists, Tuples, Dictionaries, Exam 2, Value: 25% |
- ✓ | **Lesson 15** | Analyzing all methods to create a CRUD using File Handling with lists, tuples, dictionaries |
- ✓ | **Lesson 16** | Studying different modules, Logic part of the Guess The Number game through functions, Import os, os.getcwd(), os.listdir(), Statement is in the folder, student should be taught to test each function |
- ✓ | **Lesson 17** | Using the previous code to finalize functions |
- ✓ | **Lesson 18** | Using the same code as from Lesson 16, the part where the client interacts with the program should be done |
- ✓ | **Lesson 19** | General Practice, Guess Words, Practice User Interface with Interactive Options |

- | **Lesson 20** | Introduction to Tkinter, Creating a window, Menus, Buttons, Labels, Button functionality, Pop-up windows |
- | **Lesson 21** | Text Box, Printing a message inside a text box using For Loop, Creating a txt file, ScrollBar, SpinBox, Reading information within a txt file, Radio Button, CheckBox, Frames, ListBox, Entry |
- | **Lesson 22** | Final Exam begins with a value of 50%. During lesson 22 and Lesson 23, students show progress, and present the finished project in lesson 24 |
- | **Lesson 23** | Final Exam |
- | **Lesson 24** | Final Exam, PCAP Certification |

Follow Us!

VLA
ACADEMY

somosvla

somosvla_