

PYTHON



PYTHON

Python has a simple syntax that makes it suitable for learning programming as a first language. The learning curve is smoother than other languages such as Java, which quickly requires learning about Object Oriented Programming or C/C++ that requires to understand pointers.

WHAT IS PYTHON USED FOR?

- Web Development, using the frameworks Django, Flask, Pylons
- Data Science and Visualization using Numpy, Pandas and Matplotlib
- Machine learning with Tensorflow and Scikit-learn
- Desktop applications with PyQt, Gtk, wxWidgets and many more
- Mobile applications using Kivy or BeeWare
- Education: Python is a great language to learn programming!

GETTING STARTED WITH PYTHON PROGRAMMING

- Overview
- A Brief History of Python
- How python is differ from other languages
- Python Versions
- Installing Python and Environment Setup
- How to execute Python program
- Writing your first Python program
- How to work on different Popular IDE's
[Pycharm , Jupyter Notebook]

MODULE 1: VARIABLES, KEYWORDS AND OPERATORS

- Variables
- Memory mapping of variables
- Keywords in Python
- Comments in python
- Operators
- Basics I/O and Type casting
- Getting user input



100% JOB GUARANTEE

MODULE 2: DATA TYPES IN PYTHON

- Numbers
- Strings
- Lists
- Tuples
- Dictionary
- Sets

MODULE 5: DICTIONARY AND SETS

- Introduction to Python 'dictionary' data type
- Creating a dictionary
- Dictionary built-in functions
- Introduction to Python 'set' data type
- Set and set properties
- Set built-in functions

MODULE 3: NUMBERS AND STRINGS

- Introduction to Python 'Number' & 'string' data types
- Properties of a string
- String built-in functions
- Programming with strings
- String formatting

MODULE 6 :DECISION MAKING & LOOPS

- Introduction of Decision Making
- Control Flow and Syntax
- The if Statement
- The if..else Statement
- The if...elif...else Statement
- Nested if...else Statement
- The while Loop
- break and continue Statement
- The for Loop
- Pass statement

MODULE 4: LISTS AND TUPLES

- Introduction to Python 'list' data type
- Properties of a list
- List built-in functions
- Programming with lists
- List comprehension
- Introduction to Python 'tuple' data type
- Tuples as Read only lists

MODULE 7: USER DEFINED FUNCTIONS

- Introduction of functions
- Function definition and return
- Function call and reuse

[CONTINUE]

MODULE 7: USER DEFINED FUNCTIONS

- Function parameters
- Function recipe and docstring
- Built in functions
- Scope of variables
- Recursive functions
- Lambda Functions / Anonymous Functions
- Iterators
- Generators
- Zip function
- Closures
- Decorators
- Map , Filter & Reduce functions
- *args and **kwargs

MODULE 8: MODULES AND PACKAGES

- Modules
- Importing module
- Standard Module – sys
- Standard Module – OS
- The dir Function
- Packages

MODULE 9: EXCEPTION HANDLING IN PYTHON

- Understanding exception

[CONTINUE]

MODULE 9: EXCEPTION HANDLING IN PYTHON

- Run Time Errors
- Handling I/O Exceptions
- Try, except, else and finally statement
- Raising exceptions with: raise, assert

MODULE 10: FILE HANDLING IN PYTHON

- Working with files
- File objects and Modes of file operations
- Reading, writing and use of 'with' keyword
- read(), readline(), readlines(), seek(), tell() methods
- Handling comma separated value files (CSV file handling)
- CSV reading and writing with DictWriter
- Pickling (Pickle file handling)

MODULE 11: EMAIL SENDING AUTOMATION

- Understanding SMTP
- Sending email with sendmail() function
- Email sending with attachment and MIME

MODULE 12: REGULAR EXPRESSION

- Pattern matching
- Meta characters for making patterns

[CONTINUE]

MODULE 12: REGULAR EXPRESSION

- Re flags
- Use of match() , sub() , findall(), search(), split() methods

MODULE 13 :OBJECT ORIENTED PROGRAMMING WITH PYTHON

- OOPs concepts: Classes and objects
- Making of a class and module namespace
- Static and instance variables
- Deep understanding of self and init ()
- Inheritance and Overriding
- Overloading functions
- Operator overloading
- Encapsulation: Hiding attributes
- Understanding threads,multithreading

MODULE 14:DATABASE CONNECTIVITY WITH PYTHON

- Working with MySQL database
- Working with Sqlite3 database

MODULE 15: SOCKET PROGRAMMING & TKINTER GUI APPLICATION

- Introduction to Tkinter module
- Using root window
- Creating frames

MODULE 15: SOCKET PROGRAMMING & TKINTER GUI APPLICATION

- Using Labels and Buttons
- Using Text and Entry widgets
- KM to M converter application
- Project: Calculator, Notepad etc.
- How to create setup file of project.

MODULE 16: INTRODUCTION TO DJANGO

- What is Django?
- Django and Python
- Django's take on MVC: Model, View and Template
- DRY programming: Don't Repeat Yourself
- How to get and install Django

MODULE 17: GETTING STARTED WITH DJANGO

- About the 3 Core Files:
 - models.py
 - urls.py
 - views.py
- Setting up database connections
- Managing Users & the Django admin tool
- Installing and using 'out of the box' Django features
- Django URL Patterns and Views**
 - Designing a good URL scheme
 - Generic Views

DJANGO FORMS

- Form classes
- Validation
- Authentication
- Advanced Forms processing techniques

DJANGO & REST APIS

- Django REST framework
- Django-piston

UNIT TESTING WITH DJANGO

- Overview / Refresher on Unit Testing and why it's good
- Using Python's unittest2 library
- Test
- Test Databases
- Doctests
- Debugging Best Practices

TRAINING INCLUDES THE FOLLOWING SERVICES

- Theoretical sessions
- Practical session
- Doubt solving session
- Analysis & Evaluation
- Assignments
- 1 mini project
- 1 live project
- Special soft skill training sessions
- Interview preparation
- Interviews
- Experience Certificate