WISDOMTRIBES

ALL IN ONE SOLUTION

Python Training Course

Python Training Course Duration

35 Days, Daily 1 Hours

WISDOMTRIBES

Website:www.wisdomtribes.com

Mail:wisdomtribes23@gmail.com

Contact-7667768204/9693167469

Python Training Overview

What are the Python Course Pre-requisites

- Basic Computer Skills. ...
- Understand the Difference Between Front-end and Back-end. ...
- Probability & Statistics. ...
- Installation of Python (on your PC or Mac) ...

Who should do the course

- Python is easy to understand and once you do, you can use those skills to land a wonderful career in the rapidly developing data science industry

Objectives of the Course

- build websites and software, automate tasks, and conduct data analysis

Python Course Content

- Core Python
- Advanced Python
- PANDAS
- NUMPY

- Introduction to Languages
- Introduction to Python
- Python Software’s
- Python Language Fundamentals
- Different Modes of Python
- Python Variables
- Operators
- Input & Output Operators
- Control Statements
- Data Structures or Collections
- Functions
- Loops

Introduction to Languages

- What is Language?
- Types of languages
- Introduction to Translators
  - Compiler
  - Interpreter
- What is Scripting Language?
- Types of Script
- Programming Languages v/s Scripting Languages
- Difference between Scripting and Programming languages
- What is programming paradigm?
- Procedural programming paradigm
- Object Oriented Programming paradigm

Introduction to Python

- What is Python and its History?
- Features – Dynamic, Interpreted, Object oriented, Embeddable, Extensible, Large standard libraries, Free and Open source
- Why Python is General Language?
- Limitations of Python
- What is PSF?
- Python implementations and its applications
- Python versions
- PYTHON IN REALTIME INDUSTRY
- Difference between Python 2.x and 3.x
- Difference between Python 3.7 and 3.8
- Software Development Architectures

Python Software's

- Python Distributions
- Download &Python Installation Process in Windows, Unix, Linux and Mac
- Online Python IDLE
- Python Real-time IDEs like Spyder, Jupyter Note Book, PyCharm, Rodeo, Visual Studio Code, ATOM, PyDevetc

Python Language Fundamentals

- Python Implementation Alternatives/Flavors
- Keywords
- Identifiers
- Constants / Literals
- Data types
- Python VS JAVA
- Python Syntax

Different Modes of Python

- Interactive Mode
- Scripting Mode
- Programming Elements
- Structure of Python program
- First Python Application
- Comments in Python
- Python file extensions
- Setting Path in Windows
- Edit and Run python program without IDE
- Edit and Run python program using IDEs
- INSIDE PYTHON
- Programmers View of Interpreter
- Inside INTERPRETER
- What is Byte Code in PYTHON?
- Python Debugger

Python Variables

- bytes Data Type
- byte array
- String Formatting in Python
- Math, Random, Secrets Modules
- Introduction
- Initialization of variables
- Local variables
- Global variables
- ‘global’ keyword
- Input and Output operations
- Data conversion functions – int(), float(), complex(), str(), chr(), ord()

Operators and Input & Output Operators

> Print
>Input
>Command-line arguments
>Arithmetic Operators
>Comparison Operators
>Logical Operators
>Bitwise Operators
> Shift operators
> Membership Operators
>Identity Operators
>Ternary Operator
>Operator precedence
> Difference between “is” vs “==”
> Python Assignment Operators

Data Structures or Collections

- Strings
- List
- Tuple
- Set
- Dictionary

Control Statements

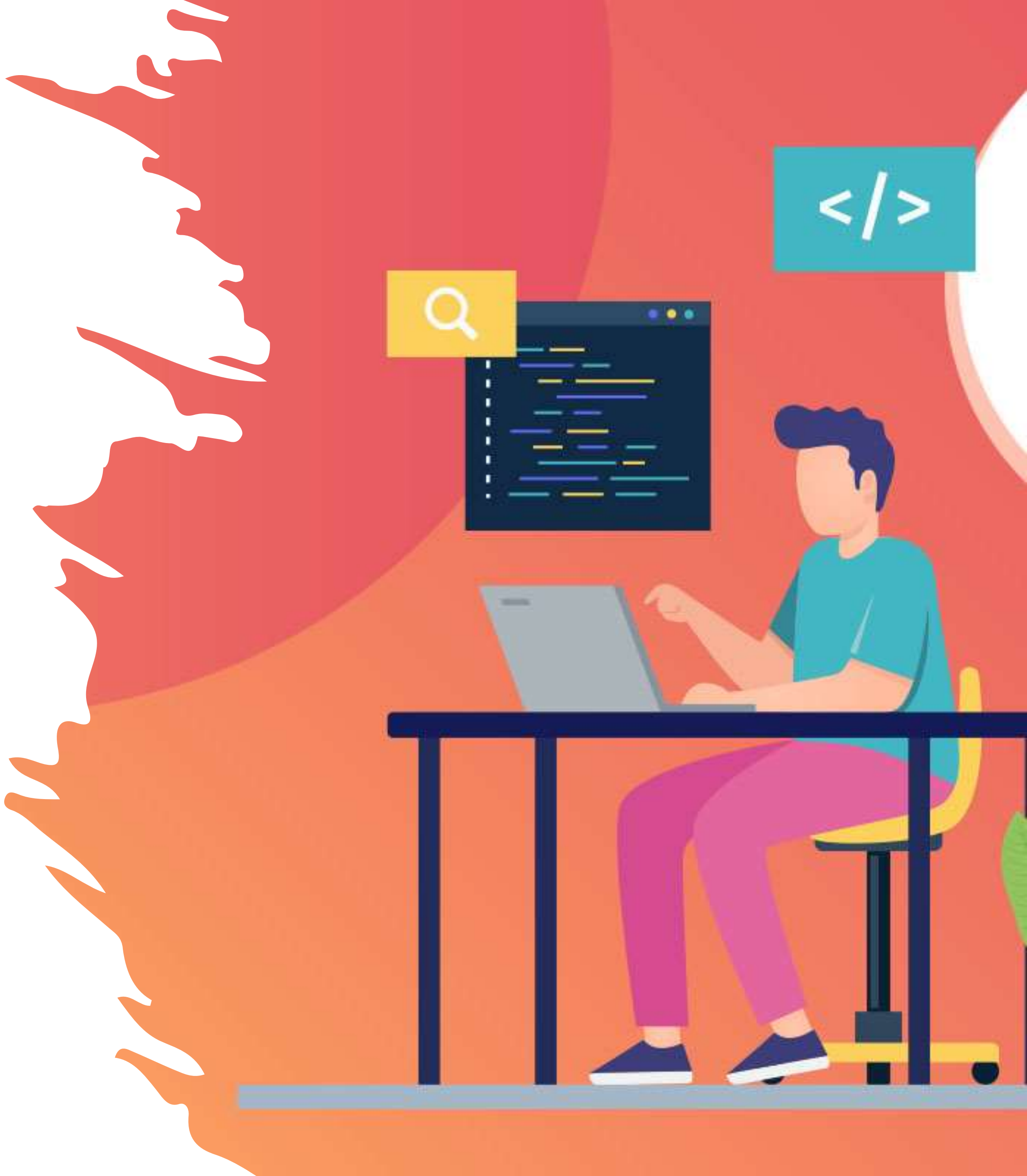
- If
- If-else
- If-elif-else
- Nested-if
- Loop control statements
- For and while
- Nested loops
- Branching statements
- Break and Continue
- Pass and Return

Python Functions

- What is Function?
- Advantages of functions
- Syntax and Writing function
- Calling or Invoking function
- Classification of Functions
  - No arguments and No return values
  - With arguments and No return values
  - With arguments and With return values
  - No arguments and With return values
  - Recursion
- Python argument type functions :
  - Default argument functions
  - Required(Positional) arguments function
  - Keyword arguments function
  - Variable arguments functions
- ‘pass’ keyword in functions
- Lambda functions/Anonymous functions
  - map()
  - filter()
  - reduce()
- Nested functions
- Non local variables, global variables
- Closures
- Decorators
- Generators
- Iterators
- Monkey patching

Advanced Python

- Python Modules
- Packages
- OOPs
  - Exception Handling & Types of Errors
  - **Regular expressions**
  - **File &Directory handling**
  - **Python Logging**
  - **Date & Time module**
  - **OS module**
  - **Multi-threading & Multi Processing**
  - **Garbage collection**
  - **Python Data Base Communications(PDBC)**
  - **Python – Network Programming**
  - **Tkinter & Turtle**
  - **Data analytics modules**



Packages

- Organizing python project into packages
- Types of packages – pre defined, user defined.
- Package v/s Folder
- py file
- Importing package
- **PIP**
- Introduction to PIP
- Installing PIP
- Installing Python packages
- Un installing Python packages

Python Modules

- Importance of modular programming
- What is module
- Types of Modules – Pre defined, User defined.
- User defined modules creation
- Functions based modules
- Class based modules
- Connecting modules
- Import module
- From … import
- Module alias / Renaming module
- Built In properties of module

OOPs

- Principles of OOP – Encapsulation , Abstraction (Data Hiding)
- Classes and Objects
- How to define class in python
- Types of variables – instance variables, class variables.
- Types of methods – instance methods, class method, static method
-
- Object initialization
- 'self' reference variable
- 'cls' reference variable
- Access modifiers – private(__) , protected(_), public
- AT property class
- Property() object
- Creating object properties using setaltr, getaltr functions
- Encapsulation(Data Binding)
- What is polymorphism?
- Overriding

- Class re-usability
- Composition
- Aggregation
- Inheritance – single , multi level, multiple, hierarchical and hybrid inheritance and Diamond inheritance

Exception Handling & Types of Errors

- What is Exception?
- Why exception handling?
- Syntax error v/s Runtime error
- Exception codes – AttributeError, ValueError, IndexError, TypeError…
  - Handling exception – try except block
  - Try with multi except
  - Handling multiple exceptions with single except block
- Finally block
  - Try-except-finally
  - Try with finally
  - Case study of finally block
- Raise keyword
  - Custom exceptions / User defined exceptions
  - Need to Custom exceptions
- Case studies

Garbage collection

- Introduction
- Importance of Manual garbage collection
- Self reference objects garbage collection
- 'gc' module
- Collect() method
- Threshold function
- Case studies

Date & Time module

- How to use Date & Date Time class
- How to use Time Delta object
- Formatting Date and Time
- Calendar module
- Text calendar
- HTML calendar

Regular expressions

- Understanding regular expressions
- String v/s Regular expression string
- "re" module functions
- Match(), Search(), Split(), Findall(), Compile(), Sub(), Subn()
- Expressions using operators and symbols
- Simple character matches
- Special characters
- Character classes
- Mobile number extraction
- Mail extraction
- Different Mail ID patterns
- Data extraction
- Password extraction
- URL extraction
- Vehicle number extraction
- Case study

File &Directory handling

- Introduction to files
- Opening file
- File modes
- Reading data from file
- Writing data into file
- Appending data into file
- Line count in File
- CSV module
- Creating CSV file
- Reading from CSV file
- Writing into CSV file
- Object serialization – pickle module
- XML parsing
- JSON parsing

Multi-threading & Multi Processing

- Introduction
- Multi tasking v/s Multi threading
- Threading module
- Creating thread – inheriting Thread class , Using callable object
- Life cycle of thread
- Single threaded application
- Multi threaded application
- Can we call run() directly?
- Need to start() method
- Sleep()
- Join()
- Synchronization – Lock class – acquire(), release() functions
- Case studies

Data analytics modules

- Numpy
- Introduction
- Scipy
- Introduction
- Arrays
- Datatypes
- Matrices
- N dimension arrays
- Indexing and Slicing
- Pandas
- Introduction
- Data Frames
- Merge , Join, Concat
- MatPlotLib introduction
- Drawing plots
- Introduction to Machine learning
- Types of Machine Learning?
- Introduction to Data science

DJANGO

- Introduction to PYTHON Django
- What is Web framework?
- Why Frameworks?
- Define MVT Design Pattern
- Difference between MVC and MVT

Python Data Base Communications(PDBC)

- Introduction to DBMS applications
- File system v/s DBMS
- Communicating with MySQL
- Python – MySQL connector
- connector module
- connect() method
- Oracle Database
- Install cx_Oracle
- Cursor Object methods
- execute() method and executeMany() method
- fetchone(), fetchmany(), fetchall()
- Static queries v/s Dynamic queries
- Transaction management

Data analytics modules

- Numpy
- Introduction
- Scipy
- Introduction
- Arrays
- Datatypes
- Matrices
- N dimension arrays
- Indexing and Slicing
- Pandas
- Introduction
- Data Frames
- Merge , Join, Concat
- MatPlotLib introduction
- Drawing plots
- Introduction to Machine learning
- Types of Machine Learning?
- Introduction to Data science

Tkinter & Turtle

- Introduction to GUI programming
- Tkinter module and Tk class
- Components / Widgets
- Label , Entry , Button , Combo, Radio
- Types of Layouts
- Handling events
- Widgets properties

Python – Network Programming

- What is Sockets?
- What is Socket Programming?
- The socket Module
- Server Socket Methods
- Connecting to a server
- A simple server-client program
- Server
- Client

OS module

- Shell script commands
- Various OS operations in Python
- Python file system shell methods
- Creating files and directories
- Removing files and directories
- Shutdown and Restart system
- Renaming files and directories
- Executing system commands

Python Logging

- Logging Levels
- implement Logging
- Configure Log File in over writing Mode
- Timestamp in the Log Messages
- Python Program Exceptions to the Log File
- Requirement of Our Own Customized Logger
- Features of Customized Logger