

VERSION

25

CRASH COURSE

SR. CODE

EAPL/CRASH/CRTC03

COURSE CODE

EACCP

SUB CATEGORY

SOFTWARE DEVELOPMENT



TOTAL DURATION

45
HOURS



THEORY TAKEN

11
HOURS



PRACTICAL TAKEN

34
HOURS

ELYSIUM
ACADEMY
CORE
PYTHON AND
CONCEPTS
**ELYSIUM
ACADEMY
CORE
PYTHON AND
CONCEPTS**

ELYSIUM
ACADEMY
CORE
PYTHON AND
CONCEPTS

COURSE DESCRIPTION



Introduction to programming basics (what it is and how it works), binary computation, problem-solving methods and algorithm development. Includes procedural and data abstractions, program design, debugging, testing, and documentation. Covers data types, control structures, functions, parameter passing, library functions, arrays, inheritance and object oriented design. Laboratory exercises in Python.

COURSE GOALS



- Understand basic principles of computers
- Understand basics of binary computation
- Understand the programming basics (operations, control structures, data types, etc.)
- Readily use the Python programming language
- Apply various data types and control structure
- Understand class inheritance and polymorphism
- Understand the object-oriented program design and development
- Understand and begin to implement code

FUTURE SCOPE



A Python developer has a highly promising future. The entire planet is becoming digital. In terms of these technologies' success, Python has emerged as the language of choice. Let's explore the technologies that rely on Python as a fundamental building block for analysis, creation, and future

Technology advancements like artificial intelligence (AI), machine and deep learning, the Internet of Things (IoT), etc. are made possible by this.

01

CHAPTER

1. Introduction To Script

- 1.1. What is Script, program?
- 1.2. Types of Scripts
- 1.3. Difference between Script and Programming Languages
- 1.4. Features and Limitation of Scripting
- 1.5. Types of programming Language Paradigms

2. Introduction To Python

- 2.1. What is Python?
- 2.2. Why Python?
- 2.3. Who Uses Python?
- 2.4. Characteristics of Python
- 2.5. What is PSF?
- 2.6. History of Python
- 2.7. Python Versions
- 2.8. How to Download and Install Python
- 2.9. Install Python with Diff IDEs
- 2.10. Features and Limitations of Python
- 2.11. Creating Your First Python Program
Python Applications
- 2.12. Printing to the Screen
- 2.13. Reading Keyboard Input
- 2.14. Using Command Prompt and GUI or IDE
- 2.15. Python Distributions



01
HRS



03
HRS

02

CHAPTER

1. Different Modes In Python

- 1.1. Execute the Script
- 1.2. Interactive and Script Mode
- 1.3. Python File Extensions
- 1.4. SETTING PATH IN Windows
- 1.5. Clear screen inside python
- 1.6. Learn Python Main Function
- 1.7. Python Comments
- 1.8. Quit the Python Shell
- 1.9. Shell as a Simple Calculator
- 1.10. Order of operations
- 1.11. Multiline Statements
- 1.12. Quotations in Python
- 1.13. Python Path Testing
- 1.14. Joining two lines
- 1.15. Python Implementation Alternatives
- 1.16. Sub Packages in Python
- 1.17. Uses of Python in Data Science, IoT
- 1.18. Working with Python in Unix/Linux/Windows/Mac/Android..!!

2. Python New IDEs

- 2.1. PyCharm IDE
- 2.2. How to Work on PyCharm PyCharm Components
- 2.3. Debugging process in PyCharm PYTHON Install Anaconda
- 2.4. What is Anaconda? Coding Environments



01
HRS



03
HRS

2.3. Spyder Components General Spyder Features

2.4. Spyder Shortcut Keys

2.5. Jupyter Notebook

2.6. What is Conda? And Conda List?

2.7. Jupyter and Kernels

2.8. What is PIP?

03

CHAPTER

1. Variables in Python

1.1. What is Variable?

1.2. Variables and Constants in Python

1.3. Variable, Variable names and Value

1.4. Mnemonic Variable Names Values and Types

1.5. What Does "Type" Mean?

1.6. Multiple Assignment

1.7. Python different numerical types Standard Data Types

1.8. Operators and Operands

1.9. Order of Operations Swap variables

1.10. Python Mathematics Type Conversion

1.11. Mutable Versus Immutable Objects

2. Python Datatypes

2.1. What is a data type?

2.2. Types of Data types

2.3. Numbers



01
HRS



05
HRS

2.4. List

2.5. Tuple

2.6. Strings

2.7. Dictionary

2.8. Sets

3. List

3.1. Lists are mutable

3.2. Getting to Lists

3.3. List indices

3.4. Traversing a list

3.5. List operations, slices and methods

3.6. Map, filter and reduce

3.7. Deleting elements

3.8. Lists and strings

4. Tuples

4.1. Advantages of Tuple over List

4.2. Packing and Unpacking Comparing tuples

4.3. Creating nested tuple Using tuples as keys in dictionaries

4.4. Deleting Tuples Slicing of Tuple

4.5. Tuple Membership Test Built-in functions with Tuple

04

CHAPTER

1. Dictionary

- 1.1. How to create a dictionary?
- 1.2. PYTHON HASHING? Python Dictionary Methods
- 1.3. Copying dictionary Updating Dictionary
- 1.4. Delete Keys from the dictionary Dictionary items() Method
- 1.5. Sorting the Dictionary Python Dictionary in-built Functions
- 1.6. Dictionary len() Method
- 1.7. Variable Types Python List cmp() Method
- 1.8. Dictionary Str(dict)

2. Set

- 2.1. How to create a set?
- 2.2. Iteration Over Sets Python Set Methods
- 2.3. Python Set Operations Union of sets
- 2.4. Built-in Functions with Set
- 2.5. Python Frozenset

3. Strings

- 3.1. What is string?
- 3.2. String operations and indices Basic String Operations
- 3.3. String Functions, Methods
- 3.4. Delete a string
- 3.5. String Multiplication and concatenation
- 3.6. Python Keywords, Identifiers and Literals
- 3.7. String Formatting Operator



01
HRS



03
HRS

3.8. Structuring with indentation in Python

3.9. Built-in String Methods

05

CHAPTER

1. Python operators

- 1.1. Arithmetic, Relational Operators and Comparison Operators
- 1.2. Python Assignment Operators Short hand Assignment Operators
- 1.3. Logical Operators or Bitwise Operators Membership Operators
- 1.4. Identity Operators Operator precedence
- 1.5. Evaluating Expressions

2. Python Conditional Statements

- 2.1. How to use "if condition" in conditional structures
- 2.2. if statement (One-Way Decisions)
- 2.3. if .. else statement (Two-way Decisions)
- 2.4. How to use "else condition"
- 2.5. if .. elif .. else statement (Multi-way)
- 2.6. When "else condition" does not work
- 2.7. How to use "elif" condition
- 2.8. How to execute conditional statement with minimal code
- 2.9. Nested IF Statement



01
HRS



03
HRS

06

CHAPTER

1. Operators

- 1.1. What is an operator?
- 1.2. Different type of operators
- 1.3. Arithmetic Operators
- 1.4. Assignment operator
- 1.5. Unary minus operator
- 1.6. Relational operators
- 1.7. Logical operators
- 1.8. Membership operators
- 1.9. Identity operators

2. Python LOOPS

- 2.1. How to use "While Loop" and "For Loop"
- 2.2. How to use For Loop for set of other things besides numbers
- 2.3. Break statements, Continue statement, Enumerate function for For Loop
- 2.4. Practical Example How to use for loop to repeat the same statement over and again
- 2.5. Break, continue statements



01
HRS



03
HRS

07

CHAPTER

1. Python Functions What is a function?

- 1.1. How to define and call a function in Python Types of Functions
- 1.2. Significance of Indentation (Space) in Python How Function Return Value?



01
HRS



03
HRS

- 1.3. Types of Arguments in Functions
- 1.4. Default Arguments and Non-Default Arguments
- 1.5. Keyword Argument and Non-keyword Arguments Arbitrary Arguments
- 1.6. Rules to define a function in Python
- 1.7. Various Forms of Function Arguments >
- Scope and Lifetime of variables
- 1.8. Nested Functions
- 1.9. Call By Value, Call by Reference
- 1.10. Anonymous Functions/Lambda functions
- 1.11. Passing functions to function
- 1.12.map(), filter(), reduce() functions
- 1.13. What is a Docstring?

2. Lambda Operator, Filter, Reduce and Map

- 2.1. Lambda function
- 2.2. Filter function
- 2.3. Reduce function
- 2.4. Map function

08

CHAPTER

1. List Comprehension

- 1.1. Introduction
- 1.2. Generator Comprehension
- 1.3. Set Comprehension



01
HRS



03
HRS

2. Modules

- 2.1. Importing module
- 2.2. Math module
- 2.3. Random module
- 2.4. Packages
- 2.5. Composition

3. Input-Output

- 3.1. Printing on screen
- 3.2. Reading data from keyboard
- 3.3. Opening and closing file
- 3.4. Reading and writing files
- 3.5. Functions

09

CHAPTER

1. Exception Handling

- 1.1. Exception
- 1.2. Exception Handling
- 1.3. Except clause
- 1.4. Try??? finally clause
- 1.5. User Defined Exceptions

2. Regular expressions

- 2.1. Match function
- 2.2. Search function
- 2.3. Matching VS Searching
- 2.4. Modifiers
- 2.5. Patterns



1.5
HRS



03
HRS

10

CHAPTER

1. Packages

- 1.1. Predefined Packages
- 1.2. User Defined

2. Packages File Handling

- 2.1. Text Files
- 2.2. Binary Files
- 2.3. Zip and Unzip Files
- 2.4. Pickling
- 2.5. Unpickling
- 2.6. Reading Program from another Program In Command Prompt

3. File Handling

- 3.1. Python File Handling
- 3.2. Python Read Files
- 3.3. Python Write/Create Files
- 3.4. Python Delete Files



1.5
HRS



05
HRS

Placement Assistance

100%

135+ Professional Courses

Practical Sessions

90%

67+ Global Pacts

Corporate Placements

65%

170+ IT Companies Tie-Up

ELYSIUM
GROUP OF
COMPANIES

**ELYSIUM
ACADEMY**

**PRIVATE
LIMITED**

AUTHORIZED INTERNATIONAL

Partners

