PYTHON

SYLLABUS

Core Python Syllabus

1. INTRODUCTION TO PYTHON
- Features of Python
- Python Virtual Machine (PVM)
- Software Download Install
- Memory management in Python

2. WRITING OUR FIRST PYTHON PROGRAM
- Writing our first Python program
- Executing a Python program
- Getting help in Python
- Reopening the Python program in IDLE
- Grammar

3. INPUT AND OUTPUT
- Output statements
- Various formats of print()
- Input statements
- Constants in Python
- Identifiers and Reserved words
- Naming conventions in Python

4. DATATYPES IN PYTHON
- Comments in Python, Docstrings
- How Python sees variables
- Datatypes in Python
- Sequences in Python
- Sets
- Literals in Python
- Determining the datatype of a variable
- Characters in Python
- User-defined datatypes

5. OPERATORS IN PYTHON
- Arithmetic operators

- Using Python interpreter as calculator
- Assignment operators
- Unary minus operator

- Relational operators
- Logical operators
- Boolean operators
- Membership operators
- Identity operators
- Operator precedence and associativity
- Mathematical functions

6. COMMAND LINE ARGUMENTS
- The built in argv[] list
- Entering various elements from command prompt
- Processing command line arguments

7. CONTROL STATEMENTS
- if statement
- if … else statement
- if … elif … else statement
- while loop
- for loop
- Infinite loops
- Nested loops
- break statement
- continue statement
- pass statement
- assert statement
- return statement

8. FUNCTIONS
- Defining a function
- Calling a function
- Returning results from a function
- Returning multiple values from a function
- Functions are first class objects
- Pass by object reference
- Formal and actual arguments
- Positional arguments
- keyword arguments
- Default arguments
- Variable length arguments
- Local and global variables
- The global keyword
- Passing a group of elements to a function

- Recursive functions
- Function decorators
- Generators
- Structured programming
- Creating our own modules in Python
- The special variable **name**
-

9. ARRAYS USING NUMPY
- Creating an array
- Importing the array module
- Indexing and slicing on arrays
- Types of arrays
- Working with arrays using numpy
- Creating arrays using linspace
- Creating arrays using logspace
- Creating arrays using arange() function
- Creating arrays using zeros() and ones() functions
- Mathematical operations on arrays
- Comparing arrays
- Aliasing the arrays
- Viewing and Copying arrays
- Slicing and indexing in numpy arrays
- Dimensions of arrays
- Attributes of an array
- reshape()
- flatten()
- Working with Multi dimensional arrays
- The array() function
- ones() and zeros() functions
- eye() function
- reshape() function
- Indexing in multi dimensional arrays
- Slicing the multi dimensional arrays
- Matrices in numpy

10. STRINGS AND CHARACTERS
- Creating strings
- Length of a string
- Indexing in strings
- Repeating the strings
- Concatenation of strings

- Checking membership

- Comparing strings
- Removing spaces from a string
- Finding sub strings
- Strings are immutable
- Replacing a string with another string
- Splitting and joining strings
- Changing case of a string
- Checking starting and ending of a string
- String testing methods
- Formatting the strings
- Sorting strings

11. LAMBDAS
- Introduction to Lambdas
- Using lambdas with filter() function
- Using lambdas with map() function
- Using lambdas with reduce() function

12. MODULES AND PACKAGES
- Structured Programming
- Creating our own modules in Python
- The special variable __name__
- Creating our own Package
- Accessing the modules from the package

13. LISTS AND TUPLES
- Creating lists using range() function
- Updating the elements of a list
- Concatenation of two lists
- Repetition of lists
- Membership in lists
- Aliasing and cloning lists
- Methods to process lists
- Nested lists
- List comprehensions
- Tuples
- Creating tuples
- Accessing the tuple elements
- Basic operations on tuples

14. LIST COMPREHENSIONS
- List comprehension examples

15. DICTIONARIES
- Operations on dictionaries
- Dictionary methods
- Using for loop with dictionaries
- Sorting the elements of a dictionary using lambdas
- Converting lists into dictionary
- Converting strings into dictionary

Advanced Python Syllabus

16. INTRODUCTION TO OOPS
- Problems in Procedure Oriented Approach
- Features of Object Oriented Programming System (OOPS)
- Classes and objects
- Encapsulation
- Abstraction
- Inheritance
- Polymorphism

17. CLASSES AND OBJECTS
- self variable
- Constructor
- Types of variables
- Namespaces
- Types of methods: instance, class and static
- Passing members of one class to another class
- Inner classes

18. INHERITANCE
- Constructors in inheritance
- Overriding super class constructors and methods
- super() method
- Types of inheritance
- Method Resolution Order (MRO)
- Duck typing philosophy of Python

19. POLYMORPHISM
- Operator overloading
- Method overloading
- Method overriding
- Constructor overloading

20..ABSTRACT CLASSES AND INTERFACES
- Abstract Method and Abstract Class
- Interfaces in Python
- Abstract Classes vs. Interfaces

21. EXCEPTIONS
- Errors vs Exceptions
- Exception handling
- Types of exceptions
- The except block
- assert statement
- User- defined exceptions
- Logging the exceptions

22. FILES IN PYTHON
- Types of files in Python
- Working with text files with strings
- Knowing whether a file exists or not
- with block
- Working with binary files
- Pickle in Python
- seek() and tell()
- Zipping and Unzipping files
- Running other programs from Python program

23. REGULAR EXPRESSIONS IN PYTHON
- Sequence characters in regular expressions
- Quantifiers in regular expressions
- Special characters in regular expressions
- Using regular expressions on files
- Retrieving information from a HTML file

24. DATE AND TIME
- The epoch

- Date and time now
- Combining date and time
- Formatting dates and times
- Finding durations using timedelta
- Comparing two dates
- Sorting dates
- Stopping execution temporarily
- Knowing the time taken by a program
- Working with Calendar module

25. THREADS
- Difference between process and thread
- Concurrent programming and GIL
- Uses of threads
- Creating threads in Python
- Thread class methods
- Single tasking using a thread
- Multi tasking using multiple threads
- Thread synchronization
- Thread deadlock
- Daemon threads

26. NETWORKING IN PYTHON
- TCP/IP Protocol
- User Datagram Protocol (UDP)
- Sockets
- Knowing IP Address
- Reading the source code of a web page
- Downloading a web page from Internet
- Downloading an image from Internet
- TCP/IP Server and Client
- UDP Server and Client
- File server
- File client
- Sending a simple mail

27. * PYTHON'S DATABASE CONNECTIVITY
- Advantages of a DBMS over files
- Working with MySQL database in Python
- Operations on rows of a table
- Creating database tables through Python
- Working with Oracle database in Python

- Stored procedures

28. * GRAPHICAL USER INTERFACE
- GUI in Python
- the root window
- Fonts and colors
- Working with containers
- Canvas
- Frame
- Widgets
- Button widget
- Label widget
- Message widget
- Text widget
- Scrollbar widget
- Checkbutton widget
- Radiobutton widget
- Entry widget
- Menu widget

29.DataStructure using Python
31.Stack
32.Queue
33.singleLinklist
34.Double Linklist
35.CircularLinklist
36. * DATA ANALYSIS USING PANDAS
- Introduction to data science
- What is data science?
- Data Frame
- Data Analysis
- Data visualization
- Line chart, bar diagram, histogram, pie chart

37. * DATA VISUALIZATION USING MATPLOTLIB
- Introduction to data science
- Data visualization

- Line chart
- Bar diagram
- Histogram
- Pie chart
- Scatter plot
- Box plot