INDEX

1) OOP's Part – 1 1

2) OOP's Part – 2 30

3) OOP's Part – 3 58

4) OOP's Part – 4 71

5) Exception Handling 83

6) File Handling 104

7) Multi Threading 123

8) Python Database Programming 151

9) Regular Expressions & Web Scraping 166

10) Decorator Functions 180

11) Generator Functions 185

12) Assertions 190

13) Python Logging 193

DETAILED INDEX

- **OOP's Part – 1 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~ 1**
  - What is Class? 2
  - How to define a Class? 2
  - What is Object? 3
  - What is Reference Variable? 3
  - Self Variable 4
  - Constructor Concept 4
  - Differences between Methods and Constructors 6
  - Types of Variables 6
    - Instance Variables (Object Level Variables)
    - Static Variables (Class Level Variables)
    - Local variables (Method Level Variables)
  - Where we can declare Instance Variables 6
    - Inside Constructor by using self variable
    - Inside Instance Method by using self variable
    - Outside of the class by using object reference variable
  - How to Access Instance Variables 8
  - How to delete Instance Variable from the Object 8
  - Static Variables 10
  - Instance Variable vs Static Variable 10
  - Various Places to declare Static Variables 10
  - How to access Static Variables 11
  - Where we can modify the Value of Static Variable 12
  - How to Delete Static Variables of a Class 15
  - Local Variables 18
  - Types of Methods 19
    - Instance Methods
    - Class Methods
    - Static Methods

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 040 – 64 51 27 86, 80 96 96 96 96, 92 46 21 21 43 | www.durgasoft.com

- Setter and Getter Methods 20
- Passing Members of One Class to Another Class 23
- Inner Classes 24
- Garbage Collection 27
- How to enable and disable Garbage Collector in our Program 27
- Destructors 28
- How to find the Number of References of an Object 29

OOP's Part – 2 ~~~~~~~~~~~~~~~~~~~~~~~~ 30

- Inheritance 31
  - By Composition (Has-A Relationship)
  - By Inheritance (IS-A Relationship

- IS-A vs HAS-A Relationship 36
- Composition vs Aggregation 38

- Types of Inheritance 41
  - Single Inheritance
  - Multi Level Inheritance
  - Hierarchical Inheritance
  - Multiple Inheritance
  - Hybrid Inheritance
  - Cyclic Inheritance

- Method Resolution Order (MRO) 46
- Head Element vs Tail Terminology 46
- How to find Merge? 46
- Finding mro(P) by using C3 Algorithm 48
- super() Method 51
- How to Call Method of a Particular Super Class? 53
- Various Important Points about super() 53

OOP's Part – 3 ~~~~~~~~~~~~~~~~~~~~~~~~~~ 58

- Polymorphism 59
- Duck Typing Philosophy of Python 59
- Overloading 62
  - Operator Overloading
  - Method Overloading
  - Constructor Overloading
- Overriding 68
  - Method Overriding
  - Constructor Overriding

OOP's Part – 4 ~~~~~~~~~~~~~~~~~~~~~~~~~~ 71

- Abstract Method 72
- Abstract class 73
- Interface 76
- Concreate class vs Abstract Class vs Inteface 78
- Public,Private and Protected Members 78
- __str__() Method 80
- Difference between str() and repr() functions 80
- Small Banking Application 81

Exception Handling ~~~~~~~~~~~~~~~~~~~~~~~~~~ 83

- Syntax Errors 84
- Runtime Errors 84
- What is Exception 85
- Default Exception Handing in Python 85
- Python's Exception Hierarchy 86
- Customized Exception Handling by using try-except 87
- Control Flow in try-except 87
- How to Print Exception Information 88
- try with Multiple except Blocks 88
- Single except Block that can handle Multiple Exceptions 90
- Default except Block 90
- finally Block 91
- Control Flow in try-except-finally 93
- Nested try-except-finally Blocks 94
- Control Flow in nested try-except-finally 95

- else Block with try-except-finally 96
- Various possible Combinations of try-except-else-finally 97
- Types of Exceptions 101
  - Predefined Exceptions
  - User Definded Exceptions
- How to Define and Raise Customized Exceptions 102

File Handling ~~~~~~~~~~~~~~~~~~~~~~~~ 104

- Types of Files 105
  - Text Files
  - Binary Files
- Opening a File 105
- Closing a File 106
- Various Properties of File Object 106
- Writing Data to Text Files 107
  - write(str)
  - writelines(list of lines)
- Reading Character Data from Text Files 108
  - read() → To Read Total Data from the File
  - read(n) → To Read 'n' Characters from the File
  - readline() → To Read only one Line
  - readlines() → To Read all Lines into a List
- The with Statement 109
- The seek() and tell() Methods 110
- How to check a particular File exists OR not 111
- Handling Binary Data 113
- Handling CSV Files 113
- Writing Data to CSV File 114
- Reading Data from CSV File 114
- Zipping and Unzipping Files 115
- To Create Zip File 115
- Working with Directories 116
- Running Other Programs from Python Program 118
- How to get Information about a File 119
- Pickling and Unpickling of Objects 120

Multi Threading ~~~~~~~~~~~~~~~~~~~~~~~~ 123

- Multi Tasking 124
  - Process based Multi Tasking
  - Thread based Multi Tasking
- The ways of Creating Thread in Python 125
  - Creating a Thread without using any class
  - Creating a Thread by extending Thread class
  - Creating a Thread without extending Thread class
- Setting and Getting Name of a Thread 127
- Thread Identification Number (ident) 128
- enumerate() Function 129
- isAlive() Method 130
- join() Method 130
- Daemon Threads 132
- Default Nature 133
- Synchronization 134
  - Lock
  - RLock
  - Semaphore
- Synchronization By using Lock Concept 135
- Problem with Simple Lock 136
- Demo Program for Synchronization by using RLock 137
- Difference between Lock and RLock 138
- Synchronization by using Semaphore 138
- Bounded Semaphore 140
- Difference between Lock and Semaphore 140
- Inter Thread Communication 141
- Inter Thread Communication by using Event Objects 141
- Methods of Event Class 141
  - set()
  - clear()
  - isSet()
  - wait()|wait(seconds)
- Inter Thread Communication by using Condition Object 143

- Methods of Condition 143
  - acquire()
  - release()
  - wait()|wait(time)
  - notify()
  - notifyAll()
- Case Study 144
- Inter Tread Communication by using Queue 146
- Important Methods of Queue 146
  - put()
  - get()
- Types of Queues 147
  - FIFO Queue
  - LIFO Queue
  - Priority Queue
- Good Programming Practices with usage of Locks 148

Python Database Programming ~~~~~~~~~~~~~~ 151

- Storage Areas 152
  - Temporary Storage Areas
  - Permanent Storage Areas
- File Systems 152
- Databases 152
- Python Database Programming 153
- Working with Oracle Database 155
- Installing cx_Oracle 155
- How to Test Installation 155
- Working with MySQL Database 162
- Commonly used Commands in MySQL 162
- Driver/Connector Information 163
- How to Check Installation 163

Regular Expressions & Web Scraping ~~~~~~~~~ 166

- Character Classes 168
- Pre defined Character Classes 169
- Qunatifiers 169
- Important Functions of Remodule 170
    1) match()
    2) fullmatch()
    3) search()
    4) findall()
    5) finditer()
    6) sub()
    7) subn()
    8) split()
    9) compile()
- Web Scraping by using Regular Expressions 177

Decorator Functions ~~~~~~~~~~~~~~~~~~~ 180

Decorator Chaining 183

Generator Functions ~~~~~~~~~~~~~~~~~~~ 185

- Advantages of Generator Functions 188
- Generators vs Normal Collections wrt Performance 188
- Generators vs Normal Collections wrt Memory Utilization 189

Assertions ~~~~~~~~~~~~~~~~~~~~~~~~~ 190

- Debugging Python Program by using assert Keyword 191
- Types of assert Statements 191
    - Simple Version
    - Augmented Version
- Exception Handling vs Assertions 192

Python Logging ~~~~~~~~~~~~~~~~~~~~~~~~ 193

- Logging Levels 194
- How to implement Logging 194
- How to configure Log File in over writing Mode 196
- How to Format Log Messages 196
- How to add Timestamp in the Log Messages 197
- How to Change Date and Time Format 197
- How to write Python Program Exceptions to the Log File 198
- Problems with Root Logger 199
- Need of Our Own Customized Logger 200
- Advanced logging Module Features: Logger 200
- Logger with Configuration File 203
- Creation of Custom Logger 205
- How to Create seperate Log File based on Caller 206
- Advantages of Customized Logger 208