

INSERT Operation.....	293
READ Operation	295
Update Operation	296
DELETE Operation	297
Performing Transactions	298
COMMIT Operation.....	299
ROLLBACK Operation	299
Disconnecting Database	299
Handling Errors	300
22. PYTHON – NETWORK PROGRAMMING.....	302
What is Sockets?	302
The <i>socket</i> Module	303
Server Socket Methods	303
Client Socket Methods	304
General Socket Methods	304
A Simple Server	304
A Simple Client	305
Python Internet modules	306
Further Readings	307
23. PYTHON – SENDING EMAIL.....	308
Sending an HTML e-mail using Python	309
Sending Attachments as an E-mail	310
24. PYTHON – MULTITHREADING.....	313
Starting a New Thread.....	313
The <i>Threading</i> Module	314

Creating Thread Using <i>Threading</i> Module:	315
Synchronizing Threads	317
Multithreaded Priority Queue	319
25. PYTHON – XML PROCESSING	323
What is XML?	323
XML Parser Architectures and APIs:	323
Parsing XML with SAX APIs.....	325
The <i>make_parser</i> Method	325
The <i>parse</i> Method	325
The <i>parseString</i> Method.....	326
Parsing XML with DOM APIs.....	329
26. PYTHON – GUI PROGRAMMING	332
Tkinter Programming	332
Tkinter Widgets.....	333
Button.....	335
Canvas.....	338
Checkbutton.....	340
Entry	344
Frame.....	349
Label	351
Listbox.....	354
MenuButton.....	358
Menu	362
Message	366
Radiobutton	369

Scale.....	373
Scrollbar	378
Text.....	381
TopLevel.....	387
SpinBox	390
PanelWindow.....	393
LabelFrame	396
tkMessageBox.....	398
Standard Attributes.....	400
Dimensions	400
Colors.....	401
Fonts	402
Anchors.....	403
Relief styles	404
Bitmaps.....	406
Cursors	407
Geometry Management	408
Python Tkinter pack() Method	409
Python Tkinter grid() Method.....	410
Python Tkinter place() Method	411
27. PYTHON – FURTHER EXTENSIONS.....	413
Pre-Requisites for Writing Extensions	413
First Look at a Python Extension	413
The Header File <i>Python.h</i>	413
The C Functions	414

The Method Mapping Table	415
The Initialization Function	416
Building and Installing Extensions	418
Importing Extensions	418
Passing Function Parameters	418
The PyArg_ParseTuple Function	420
Returning Values	421
The <i>Py_BuildValue</i> Function	423

1. Python – Overview

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive:** You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented:** Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language:** Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, Unix shell, and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

Python's features include:

- **Easy-to-learn:** Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read:** Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain:** Python's source code is fairly easy-to-maintain.
- **A broad standard library:** Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode:** Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.