

## BLUETOOTH TESTING

- 1) Bluetooth Introduction
- 2) Bluetooth Technology
- 3) Bluetooth LE, Bluetooth low energy
- 4) Bluetooth core specification
- 5) Bluetooth profiles
- 6) Bluetooth Networking architecture
- 7) Bluetooth operations
- 8) Bluetooth implementations
- 9) Bluetooth protocol
- 10) Bluetooth architecture
- 11) Bluetooth physical layer
- 12) Physical channel
- 13) Physical links
- 14) Spectrum
- 15) Interference
- 16) Class of Radio
- 17) Power and Range
- 18) Bluetooth packets
- 19) Link manager protocol (IMP)
- 20) Host controller Interface (HCI)
- 21) Logical link control and Adaption Protocol (L2CAP)
- 22) Bluetooth security
- 23) Bluetooth smart
- 24) Bluetooth IOT
- 25) Bluetooth Standards
- 26) Bluetooth Conformance and Compatibility testing
- 27) Bluetooth 5
- 28) Bluetooth 5 and connectionless IOT
- 29) Bluetooth 5 advancing beacon and location based capabilities.

## PYTHON PROGRAMMING COURSE CONTENT

### MODULE 1: BASICS OF PYTHON

- Introduction

### *PYTHON OPERATORS*

- Arithmetic
- Relational

- Logical
- Assignment
- Bitwise Membership
- Identity operators.

## *PYTHON CONDITIONAL STATEMENTS*

- If
- If - else
- If – elif

## *PYTHON LOOPS*

- While
- For
- Range()
- Break and Continue
- Example problems

## *PYTHON NUMBERS*

- Types in numbers
- Type conversions

## *PYTHON STRINGS*

- Built-in functions
- Basic operators
- Slicing
- Example problems

## *PYTHON LISTS*

- Array or list in python
- List slicing techniques
- Built-in functions

## *PYTHON TUPLE*

- Tuple & Immutability
- Built-in functions
- Example problems

## *PYTHON DICTIONARIES*

- Creation of dictionaries
- Built-in functions
- Example problems

## *PYTHON FUNCTIONS*

- Advantages of functions

# Primebit Solution

---

- Function definition using
- def statement.
- Calling function.
- passing parameters
- scope of the variables

## **MODULES IN PYTHON**

- Importing modules
- creation of own modules
- standard modules - os / sys

## **EXCEPTION HANDLING**

- Handling exceptions
- Try-except block
- Example problems

## **FILE HANDLINGS**

- Files and directories
- Text files
- Binary files
- Text file processing
- Binary file processing

## **MODULE 2 : ADVANCED PYTHON**

### **NETWORK PROGRAMMING:**

- In this, we will teach how we will send as well as receive data between devices by using TCP/IP and UDP protocols.
- Python supports many networking protocols through libraries such as a socket, SMTP etc and you can also write network programs directly by using TCP/IP or UDP sockets.

### **TCP VS UDP / CLIENT VS SERVER - PYTHON EXAMPLES:**

- Python socket module makes it easy for you to write our own clients and server programs.

### **MULTIPROCESSING IN PYTHON:**

Multiprocessing refers the ability of a system to support more than one processor at the same time. Applications in multiprocessing system are broken to smaller routines that run independently. The operation system allocates these threads to the processors improving performance of the system.

### **FILE TRANSFER PROTOCOLS:**

In this, we will teach how can we transfer the files as well as receive the files between the devices.

### **EMAILING:**

To transfer the emails, we will use SMTP library which is a built-in library in python. SMTP ( Simple Mail Transfer Protocol) is a protocol which enables you to send emails.

