## **Python Programming Course Outline (15 Classes)**

**Class Schedule:**

* **Class 1:** Introduction to Python
  + What is programming?
  + Why Python?
  + Setting up the development environment (IDE/text editor)
  + Input and Output (print easy programmes)
* **Class 2:** Introduction to Python
  + Basic Python syntax (variables, data types, comments)
  + Running Python programs
  + Hands-on exercises: Working with variables, data types, simple calculations.
* **Class 3:** Operators and Expressions
  + Arithmetic operators (+, -, \*, /, //, %, \*\*)
  + Hands-on exercises: Writing expressions involving operators, user input.
* **Class 4:** Operators and Expressions
  + Comparison operators (==, !=, <, >, <=, >=)
  + Logical operators (and, or, not)
  + Hands-on exercises: Writing programme using Comparison operators.
* **Class 5:** Review Class
  + Review on previous topics
  + Hands-on exercises: Writing complex programmes based on previous classes.
* **Class 6:** Conditional Statements
  + Conditional statements (if/else, elif)
  + Using indentation effectively
  + Hands-on exercises: Writing conditional programs.
* **Class 7:** Conditional Statements
  + Nested conditionals
  + Switch statement
  + Hands-on exercises: Writing switch statements.
* **Class 8:** Control Flow Statements
  + Looping statements (for, while)
  + Break and continue statements
  + Hands-on exercises: Building programs with iterative tasks.
* **Class 9:** Review Class
  + Review on previous topics.
  + Hands-on exercises: Writing complex programs based on previous classes.
* **Class 10:** Functions
  + Defining functions with and without arguments
  + Calling functions
  + Hands-on exercises: Programming with functions
* **Class11:** Functions
  + Function arguments and return values
  + Local and global variables
  + Hands-on exercises: Creating reusable functions for calculations and string manipulation.
* **Class 12:** Data Structures
  + Lists: Creating and accessing elements, slicing, concatenation, iteration, mutability
  + Tuples: Creating and accessing elements (immutable lists)
  + Hands-on exercises: Working with lists, tuples, and dictionaries to organize and manipulate data.
* **Class 13:** Data Structures
  + Dictionaries: Creating key-value pairs, accessing elements, common dictionary methods (get, keys, values)
  + Try-except blocks for handling exceptions
  + Hands-on exercises: Working with lists, tuples, and dictionaries to organize and manipulate data.
* **Class 14:** Review Class
  + Review on previous topics.
  + Hands-on exercises: Writing complex programming based on previous classes.
* **Class 15:** Problem Solving and Analysis.
  + Problem on previous topics
  + Showing projects and programming.

**Advance Topics:** (if time possible)

* + Classes and objects
  + Defining attributes and methods within a class
  + Inheritance (basic concepts)

**Resources:**

* Online tutorials and documentation (e.g.<https://www.python.org/doc/>)
* Online tutorials and documentation (e.g.<https://www.w3schools.com/python/default.asp>)
* Online coding platforms (e.g.<https://www.hackerrank.com/domains/python>)