Learn Python in 2 Hours Basics of Python in Tamil

06 February 2026 10:47 AM

Youtube link [Learn Python in 2 Hours: Basics of Python in Tamil - YouTube]

1. Python Print Statements and Basics

The video begins with an introduction to **Python print statements**, emphasizing that **Python is an interpreted language**. It contrasts Python with low-level languages such as binary (zeros and ones), C++, and assembly language.

The basic usage of the print() function is demonstrated, including:

- Printing simple strings like "hello"
- Printing variables
- Performing arithmetic operations inside print(), such as multiplication (5 * 2 = 10)

It also explains the importance of using correct arithmetic symbols:

- * for multiplication
- / for division

Multiple examples show how Python evaluates expressions and outputs results.

2. Comparison Operators and Boolean Expressions

This section introduces **basic comparison operators** and **boolean expressions**:

- > (greater than), e.g., 10 > 3 → True
- < (less than), e.g., 3 < 10 → True
- Expressions like 3 > 10 evaluate to False

It emphasizes that comparison expressions return **boolean values**:

- True
- False

3. Equality and Relational Operators

This part explains:

- == (equal to)
- >= (greater than or equal to)
- <= (less than or equal to)

Examples such as 10 <= 20 are shown.

It also introduces **augmented assignment operators**, especially:

- += (e.g., a += 2)

These operators simplify updating variable values by combining an operation with assignment.

4. Logical Operators and Conditional Statements

This segment focuses on **logical operators** and **if-else statements**:

- or → returns True if at least one condition is true
- and → returns True only if both conditions are true

Examples demonstrate:

- Writing if and else conditions
- How program flow changes based on whether conditions evaluate to True or False

The structure and indentation of conditional blocks are emphasized.

5. Loops, Range Function, and Control Flow

This section introduces:

- for loops
- The range() function for iterating over numbers

It also explains **loop control statements**:

- continue → skips the current iteration
- break → exits the loop immediately

The **else clause in loops** is introduced, which runs only if the loop finishes normally (without break).

6. Loop Else Clause and List Indexing
Further clarification is given on:
- How the loop else block works
- **List indexing** in Python

Key points:
- Python uses **zero-based indexing**
- Index 0 refers to the first element

Examples show accessing list elements using their index.

7. List Functions and Operations
This part demonstrates common list functions:
- len() → finds the number of elements in a list
- min() → finds the smallest value in a list

Examples use numeric lists to show how these functions work.

8. Sorting Lists and String Operations
This segment covers:
- Sorting lists, including **descending order**
- Accessing elements by index in lists and strings

It also introduces **string operations**, such as:
- Indexing characters
- Slicing strings
- Using split() to divide strings into parts based on a delimiter

9. String Manipulation Functions
This section explains:
- The find() function for locating substrings
- Zero-based indexing in strings
- Using negative indices (e.g., -1) to access characters from the end of a string

Examples demonstrate substring extraction and searching techniques.

10. Dictionaries and Void Functions
This part introduces:
- **Dictionaries** and the .get() method
  - .get() safely retrieves a value or returns None (or a default value) if the key does not exist

It also explains **void functions**:
- Functions without a return statement
- Such functions implicitly return None

11. Functions and Parameter Passing
This section demonstrates:
- Calling functions with arguments
- Passing values into functions
- Returning results from functions

Example:
- Calling add(5, 7) returns and prints 12

It also explains that functions can be called multiple times with different inputs.

12. Default Function Parameters
The final part briefly mentions:
- **Default parameter values**
- How functions can still run even if some arguments are not provided

This allows functions to be more flexible in how they are used.

13. Key Concepts Covered

- Python print statements and syntax
- Arithmetic and comparison operators
- Boolean values (True, False)
- Logical operators (and, or)
- Conditional statements (if, else)
- Looping constructs (for, range())
- Loop control (break, continue, loop else)
- List operations
- String manipulation
- Dictionary usage
- Functions and parameters

14. Summary Table of Operators

Operator	Description	Example	Result
+	Addition	5 + 7	12
-	Subtraction	10 - 3	7
*	Multiplication	5 * 2	10
/	Division	10 / 2	5.0
>	Greater than	10 > 3	True
<	Less than	3 < 10	True
==	Equal to	a == 10	True / False
>=	Greater than or equal to	10 >= 5	True
<=	Less than or equal to	10 <= 20	True
+=	Add and assign	a += 2	Updates a
and	Logical AND	True and False	False
or	Logical OR	True or False	True