

Basic Syntax

Comments

Single line: Use # symbol

Multi-line: Use triple quotes " " " or '''

Inline comments: Add # at end of line

Print Function

Basic output: `print()` function displays text

Multiple values: Separate with commas

String formatting: Use f-strings, `format()`, or % formatting.

Indentation

Critical in Python - Defines code blocks

Use 4 spaces - Standard convention

Consistent indentation - All lines in same block must have same indentation

Data Types

int Whole numbers

float Decimal Numbers

complex Real + imaginary

str Character strings

Immutable Can't change chars

Unicode International support

bool True/False Values

list Ordered, mutable

tuple Ordered, immutable

dict Key-value pairs

set Unique elements only

Data types include numeric, text, boolean, and collections.

Formatting

f-strings `f"Hello {name} "`

`format()` `" Hello {}".format(name)`

% style `" Hello %s" % name`

Variable Declaration

No keyword needed Just assign

Dynamic typing Auto-determined

Case sensitive `myvar ≠ MyVar`

Multiple assignment One line

Naming Rules

Start Letter or underscore

Characters alphanumeric + underscore

No reserved words `if, for, class`

Convention `snake_case`

Strings (Creation Methods)

Single quotes `'Hello'`

Double quotes `" Hel lo"`

Triple quotes Multi Line

Raw strings `r'literal'`

String Operations

Concatenation + operator

Repetition * operator

Length `len()` function

Indexing `[0]` zero-based

Slicing `[start:end:step]`

String Methods

Case `upper(), lower(), title()`

Search `find(), index(), count()`

Check `starts with(), endswith()`

Modify `replace(), strip(), split()`



By **Muhammad Usman Asghar** (musmankkh)

Not published yet.

Last updated 25th July, 2025.

Page 1 of 1.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>