Python

Fundamentals of Python Programming

Get Started

VEDICSKILL.

Topics

[Optional]

- Jupyter Notebook
- Basics of Python Programming
- Operating over Sequence
- Writing Object-Oriented Programming

[Refresher]

- Type Hinting and Type Checking
- Asynchronous I/O
- Managing Pydantic Data Models

Next |  Jupyter Notebook

[Optional: Skip This Lesson if needed]

Jupyter Notebook

Ju → Julia

Py → Python

r → R

→ Support Over 40 Programming Language

Settings

Autocomplete | Code Mirror

Python Basics

print

Output values to console or standard output.

Print the values to standard output or file, converting them to strings as needed.

Syntax:

```
>>> print(<expr>)
>>> print(<expr>, <expr>, <expr>, …, sep=“ ”, end=“\n”, file=None, flush=False)
```

Example:

```
>>> print(“Hello World!”)
Hello World!
```

Variables

Variables hold data, assignment puts data inside them

A variable is a name used to store a value in memory

Syntax:

```
>>> x = 10
x  → variable name
=  → assignment operator
10 → value
```

Rule:

1. Must start with a letter or underscore
2. Letters, numbers, underscore are the allowed characters
3. Automatically data type is assigned to variable

Escape Characters

Special Sequence characters starting with backslash (\)

An escape characters is the backslash (\) followed by another characters to represent non-printable or special symbols in the strings.

Escape Sequence	Example	Output
\n	print("Hello\nWorld")	Hello ⏎ World
\t	print("Hello\tWorld")	Hello World
\\	print("C:\\Path\\To\\File")	C:\Path\To\File
\'	print(“It\'s fine”)	It's fine
\"	print("He said, \"Hi\"")	He said, "Hi"

Example:

```
>>> print(“Hello\nWorld!”)
Hello
World!
```

Format Specifiers

printf-style string formatting

In Python, format specifiers are used to insert or format a value inside a string

Specifier	Example	Output	Meaning
%s	print(“Hello %s”%“John”)	Hello John	Inserts string value
%d	print(“Age = %d”%15)	Age = 15	Inserts integer value
%f	print(“Score = %f”%16.9)	Score = 16.9	Inserts float value
%r	print(“Debug %r”%“error”)	Debug “error”	Insert raw representation of value
{}	print(“Hello {}r”.format(“John”))	Hello John	Insert any values without {} in sequence

Example:

```
>>> print(“%s is %d years old”%(“John”,35))
John is 35 years old
```

f-string

Formatted String Literal

An f-string is a modern way to insert variables and expression directly inside a string.
Recommended approach as of today

Syntax:

```
>>> print(f“string {variable}”)
```

Example:

```
>>> name = “John”
>>> print(f“Hello {name}”)
Hello John
```

User Defined Functions

def