

# python Day - 1 (Beginner)

Topic:- printing, Debugging, String Manipulation  $\rightarrow$  variables

## (1) Introduction to python Output - The print() function

The print() function is the most basic and most frequently used function in python.

It allows the computer to display information on the screen.

### Syntax

```
print("your message here")
```

Important points:-

- (\*) Anything inside quotes is treated as text (called a string).
- (\*) you can use single (' ') or double (" ") quotes.
- (\*) python always requires parentheses with print().

### Examples:-

```
print("Hello world")
```

```
print("python is fun")
```

## (2) String Manipulation

String manipulation means modifying text or changing how it appears.

## (2.1) New line character (\n)

\n tells python to move to a new line

Example :-

```
print ("Hello\nworld")
```

Output :-

Hello  
world

Meaning :-

- Python prints "Hello"
- Then moves to new line
- Then prints "world"

## (2.2) String Concatenation

Concatenation means joining two or more strings together

Example

```
print ("Hello" + " " + "Mohit")
```

Why the Space?

Strings don't add spaces automatically,  
so " " is used

Important Rules:-

- Only strings can be concatenated
- If you try "Hello" + 5 it gives an error  
because 5 is an integer

### (3) Debugging Basics

Debugging means finding and fixing errors in code.

Common Beginner Errors:

(1). Missing quotes

print (Hello)

Error Hello is not in quotes

(2). Missing parentheses

print "Hello"

python 3 requires parentheses

(3) Spelling mistake

(4) Indentation mistakes

python is indentation sensitive

wrong indentation causes IndentationError

{ Debugging Rule:-

Always read the last line of the error message - it tells the real reason for the error.

(4) Taking User Input - The input () function

The input () function pauses the program and waits for the user to type something

Syntax

input("Question or prompt")

Example

```
name = input("What is your name? ")  
print("Hello" + " " + name)
```

## Important Notes :-

- `input()` always returns a string
- If you want numbers later, you must convert them

## (5) Variables in python

variable are containers used to store data  
you can imagine them like labelled boxes  
where you keep information

### Syntax

Variable name = value

### Example

age = 20

city = "Delhi"

### Rules for naming Variables :-

(1) Cannot have Spaces

user name (✓)

user name (✗)

(2) Cannot start with a number

name1 (✓)

name (✗)

(3) Case-Sensitive

Name and name are different

(4) Use meaningful names

Student age (✓)

SA (✗)

## Reassing / Reassigning variable

python variable can change type anytime (dynamic typing)

`x = 10`

`x = "ten"`

this is allowed

{ (\*) (\*) Mini project - Band Name Generator (\*) (\*) }

(\*) This small project uses everything learned  
so far:-

- `print()`
- `input()`

- variables
- string joining

Concepts Used

- Asking questions using `input()`
- Storing answers in variables
- printing the final combined string

Key Concepts you must Remember (Day-1)

- Programming = giving instructions to the computer in a language it understands.
- `print` → Helps to check your work.
- `input` → always returns a string.
- Variables make your code useful and reusable.
- Debugging is not failure if it is a part of learning process.