Introduction to Python-II

🎯 Session Objectives

- ✨ Discover Python's features
- 📈 Understand how Python differs from Excel and SQL
- 🖌️ Learn Python's role in Data Analytics
- 📘 Understand the basic syntax of Python
- 🧠 Learn about variables and their usage
- ✍️ Declare and assign values to variables
- 🪄 Explore data types, check them, and perform type conversion

Input()

- Used to accept user input from the keyboard as a string, which can be further type cast as per the requirement.

```
user_input = input("Enter the user_name: ")
print(user_input)
```

```
Enter the user_name: UltimateForce
UltimateForce
```

Memory

```
user_input =
UltimateForce
```

Input()

- Used to accept user input from the keyboard as a string, which can be further type cast as per the requirement.

```
user_input = input("Enter the user_name: ")
print(user_input)
```

```
Enter the user_name:  Venom
Venom
```

```
user_input = input("Enter the gaming_name: ")
print(user_input)
```

```
Enter the gaming_name:  DaddyDamage
DaddyDamage
```

```
val = input("Enter the value: ")
print(val)
print(type(val))
```

```
Enter the value:  10
10
<class 'str'>
```

```
val = int(input("Enter the value: "))
print(val)
print(type(val))
```

```
Enter the value:  11
11
<class 'int'>
```

```
print(val * 5)
```

```
55
```

```
x = int(input("Enter the value between 1-100 : "))
x = x + 10
print(x)
```

```
Enter the value between 1-100 :  75
85
```

```
x = int(input("Enter the value between 1-100 : "))
x = x + 10 # '75' + 10 - TypeError: can only concatenate str (not "int") to str
print(x)
```

```
Enter the value between 1-100 :  66
76
```

```
x = str(x)
print(x)
print(type(x))
```

```
76
<class 'str'>
```

```
# String Pattern
user_name = input("Enter the username: ")
email = input("Enter your personal mail: ")
print("Hey " + user_name + " Welcome to the Python Course, You being registered with " + email + " :)")
```

```
Enter the username:  utkarshdev
Enter your personal mail:  utkarssh@gmail.com
Hey utkarshdev Welcome to the Python Course, You being registered with utkarssh@gmail.com :)
```

```
# String Pattern
user_name = input("Enter the username: ")
email = input("Enter your personal mail: ")
print(f"Hey {user_name} Welcome to the Python Course, You being registered with {email} :)")
```

```
Enter the username:  iaarifpathan
Enter your personal mail:  iaarifpathan@gmail.com
Hey iaarifpathan Welcome to the Python Course, You being registered with iaarifpathan@gmail.com :)
```

```python
# This is a single line comment
print("Python Programming")
```

```
Python Programming
```

```python
'''
    This is a multiline comments....
    Put some important stuff here relevant to
    your python program
'''
num1 = int(input("Enter the first value: "))
num2 = int(input("Enter the second value: "))
result = num1 * num2
print("The final result is " , result)
print(f"The final result is {result}")
```

```
Enter the first value:  10
Enter the second value:  5
The final result is  50
The final result is 50
```

```python
'''
    This is a multiline comments....
    Put some important stuff here relevant to
    your python program
'''
num1 = int(input("Enter the first value: "))
num2 = int(input("Enter the second value: "))
result = num1 * num2
print("The final result is " , result)
print(f"The final result is {result + 10}")
```

```
Enter the first value:  10
Enter the second value:  5
The final result is  50
The final result is 60
```

```python
fav_car = input("Enter your Dream Car: ")
print("My Dream Car is : " + fav_car)
```

```
Enter your Dream Car:  Defender
My Dream Car is : Defender
```

```python
# Indentation:
z = 15
if z > 10: # Boolean Value[True/False]
    print("z is greater than 10")
print("Outside the if statement")
```

```
z is greater than 10
Outside the if statement
```

```python
# Indentation:
z = 15
if z > 20: # Boolean Value[True/False]
    print("z is greater than 10")
print("Outside the if statement")
```

```
Outside the if statement
```

Variables : What & Why?

Variables acts as an container for storing data, They help with:

- Storing Data
- Manipulating Values
- Reusability
- Improving the Readibility
  - total_sum >> ts
  - compound_interest >> c_i
  - age
  - finding_prod_sum >> finding_the_product_and_sum >> p_s

```python
# Declaring and assigning variables:
val = 99
user_name = "UltimateForce"
print(val)
print(user_name)
```

```
99
UltimateForce
```

```python
a,b,c = 10,20,30
print(a)
print(b)
print(c)
```

```
10
20
30
```

```python
a,b,*c = 10,20,30,40,50
print(a)
print(type(a))
print(b)
print(type(b))
print(c)
print(type(c))
```

```
10
<class 'int'>
20
<class 'int'>
[30, 40, 50]
<class 'list'>
```

```python
a,*b,c = 10,20,30,40,50
print(a)
print(type(a))
print(b)
print(type(b))
print(c)
print(type(c))
```

```
10
<class 'int'>
[20, 30, 40]
<class 'list'>
50
<class 'int'>
```

```python
a,*b,c = 10,20,30,'a','b',40,50
print(a)
print(type(a))
print(b)
print(type(b))
print(c)
print(type(c))
```

```
10
<class 'int'>
[20, 30, 'a', 'b', 40]
<class 'list'>
50
<class 'int'>
```

```python
p=q=r=s= "Ninjas"
print(p)
print(q)
print(r)
print(s)
```

```
Ninjas
Ninjas
Ninjas
Ninjas
```

Rules for naming a "Variables" ¶

- Can include letters, digits and underscore
- Must start with a letter or underscore
- case-sensitive
  - val != Val
  - num != nuM
  - digit != Digit
- Can't use Python Keywords
  - for, input, while, print, if, elif, tuple, list, set, dict, true, false
- Can't have spaces or special characters (except _)
- Variable name can't start with numbers.

```python
my_first_variable = "Python"
print(my_first_variable)
```

```
Python
```

```python
_var = "Coding"
print(_var)
```

```
Coding
```

```python
# String -> Escape Character
print("Hi, I'm good, What about you?")
print('Hi, I\'m good, What about you?')
```

```
Hi, I'm good, What about you?
Hi, I'm good, What about you?
```

```python
print("an \"apple\" a day keeps a doctor away")
```

```
an "apple" a day keeps a doctor away
```

```python
print("Hi Everyone! Welcome to the course")
```

```
Hi Everyone! Welcome to the course
```

```python
# Multiline String -> <pre> tag in HTML [Markup Language]
print("""
    Tu Paas Hai Mere Paas Hai Aise
    Mera Koi Ehsaas Hai Jaise
    Tu Paas Hai Mere Paas Hai Aise
    Mera Koi Ehsaas Hai Jaise

    Haye Main Mar Hi Jaaun
    Jo Tujhko Na Paaun
    Baaton Mein Teri Main
    Raatein Bitaun

""")
```

```
    Tu Paas Hai Mere Paas Hai Aise
    Mera Koi Ehsaas Hai Jaise
    Tu Paas Hai Mere Paas Hai Aise
    Mera Koi Ehsaas Hai Jaise

    Haye Main Mar Hi Jaaun
    Jo Tujhko Na Paaun
    Baaton Mein Teri Main
    Raatein Bitaun
```

```python
# Escape characters "\n"[Next Line] & "\t"[tab space]
print("Hey: What's your name: \t 'Aditya Verma'")
print("Hello\nWorld")
print("x", end=" ")
print("y", end=" ")
print("z", end="\n")
print("Welcome Back!")
```

```
Hey: What's your name:   'Aditya Verma'
Hello
World
x y z
Welcome Back!
```

```python
# type -> conversion()
x = 10.0
print(type(x))
```

```
<class 'float'>
```

```python
num1 = 99.99
print(int(num1))
```

```
99
```

```
x = 55
y = str(x)
print(y)
print(type(y))
```

```
55
<class 'str'>
```

```
_bool = True
int_bool = int(_bool)
print(int_bool)
print(type(_bool))
print(type(int_bool))
```

```
1
<class 'bool'>
<class 'int'>
```

```
_bool = False
int_bool = int(_bool)
print(int_bool)
print(type(_bool))
print(type(int_bool))
```

```
0
<class 'bool'>
<class 'int'>
```