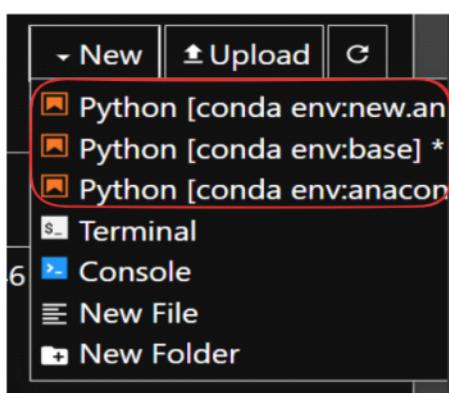
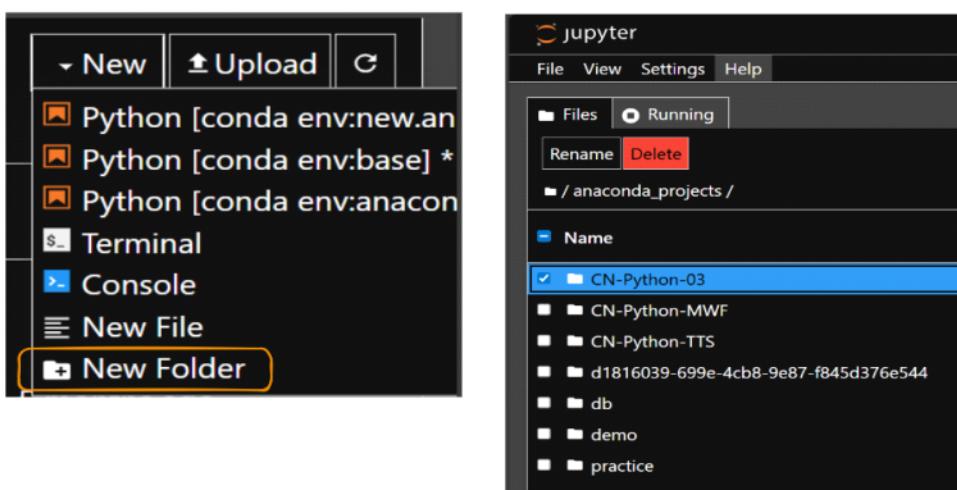


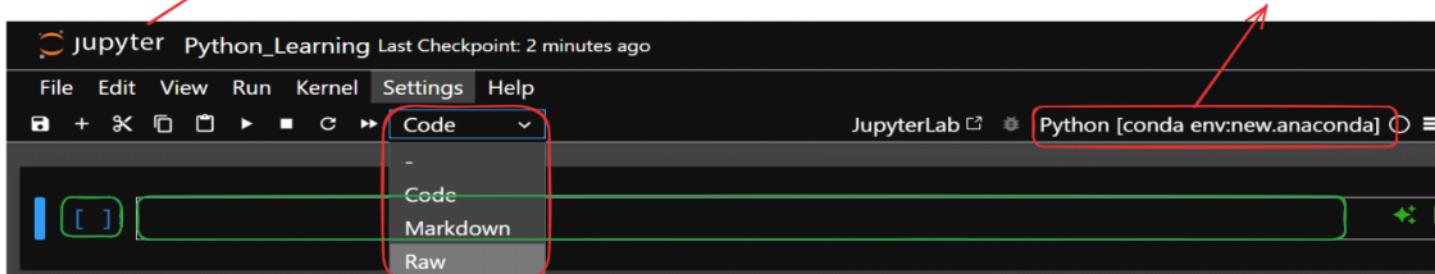
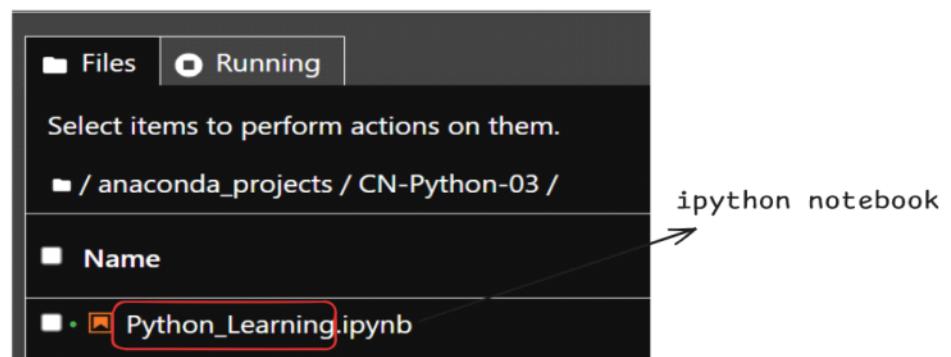
## Introduction to Python-II

### Session Objectives:

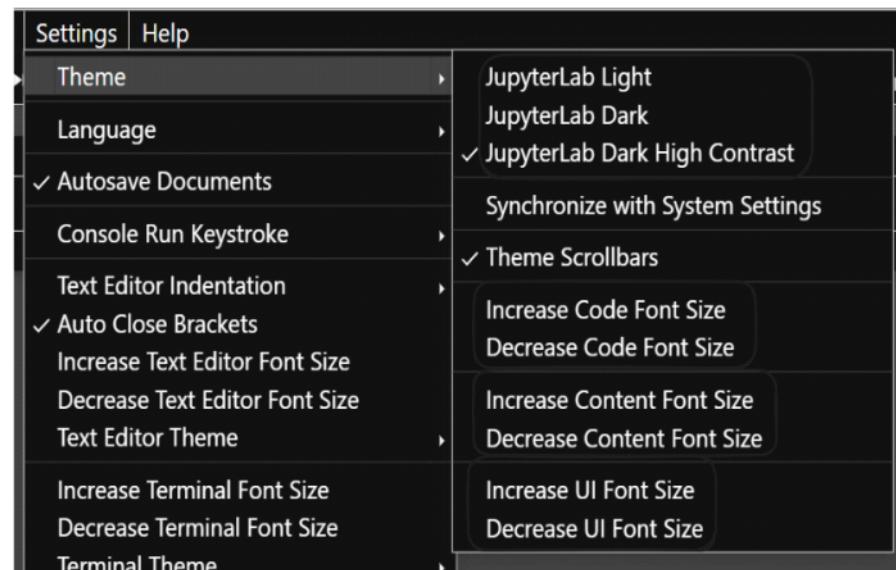
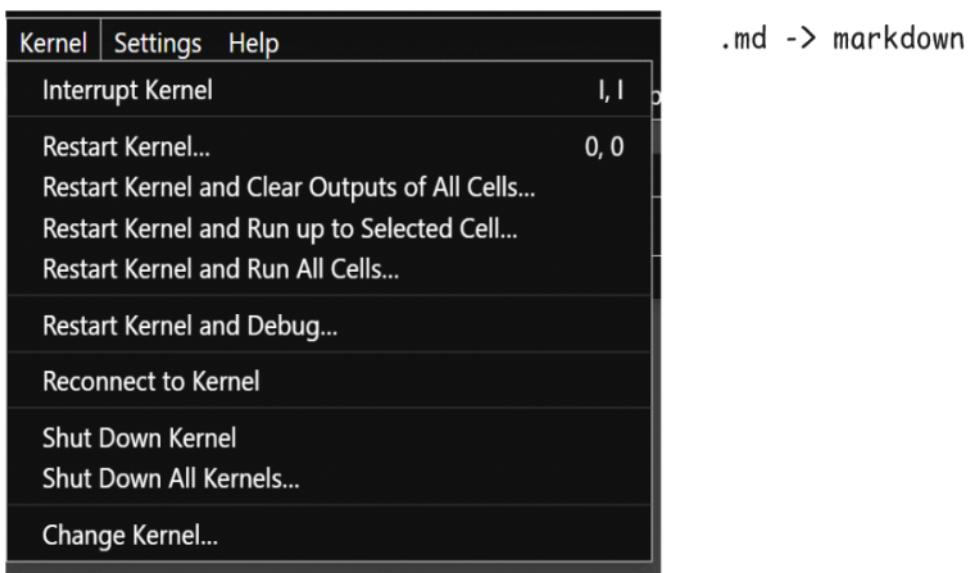
- Understand the basic syntax of Python.
- Learn about variables and their usage.
- Declare and assign values to variables.
- Differentiate between variables, identifiers, and keywords.
- Explore data types, check them, and perform type conversion.



Kernal [.ipynb]



localhost:8888/notebooks/anaconda\_projects%2FCN-Python-03%2FPython\_Learning.ipynb



A screenshot of a Jupyter Notebook cell. The cell number is [1]. The code entered is:

```
[1]: print("Hello World!")
```

The output of the cell is:

```
Hello World! console
```

Annotations with arrows point to different parts of the cell:

- An arrow points from the text "function <Keyword> [Predefined]" to the word "print".
- An arrow points from the word "context" to the word "console".

Block of code run successfully.

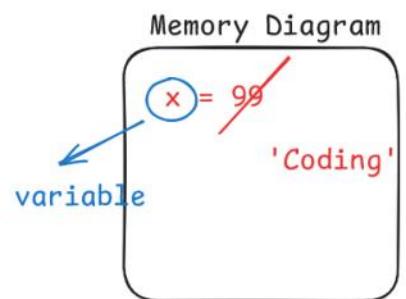
Comment [Single Line Comment]

```
# Variable -> is a container which stores data in it.
# [Python] -> variable auto detects the data type
x = 99
print(x)      Top to Bottom
print(type(x)) # What data type 'x' stores ['int']

99
<class 'int'>

x = 'Coding'
print(x) # 99 it is updated to 'Coding'
print(type(x)) # 'string' type , '' or ""

Coding
<class 'str'>
```



Console [Output]

```
99      'Coding'
'int'   'str'
```

Raw File

```
# Java
int x = 10
System.out.println(x)
String x = 'Coding'
System.out.println(x)
```

```
x = 'X' # char
print(x)
print(type(x)) # 'str'

X
<class 'str'>

x = int('21') + 1
print(x) # 22
print(type(x)) # 'int'

22
<class 'int'>
```

```
x = 99.99
print(x)
print(type(x))

99.99
<class 'float'>
```

```
x = 11 + 9j
print(x)
print(type(x))

(11+9j)
<class 'complex'>
```

```
x = True
print(x)
print(type(x)) # 'bool'

True
<class 'bool'>
```

```
# INSERT INTO Customers(customer_id, customer_name , customer_phone) -> parameters
# VALUES ('K0191' , 'Akancha Ranjan', 04727678627) -> arguments
# print [Function -> argument [end,sep]]
print('Python') \n
print('Programming')

Python
Programming

# By Default end = '\n' [which moves the content to the next Line]
print('Python', end = ' ')
print('Programming')
print("Coding is Awesome 🔥")
```

Python Programming  
Coding is Awesome 🔥

Win + ':' [To open a emoji keyboard]

```
print('Python', end = '\t') # tab like space
print('Programming')

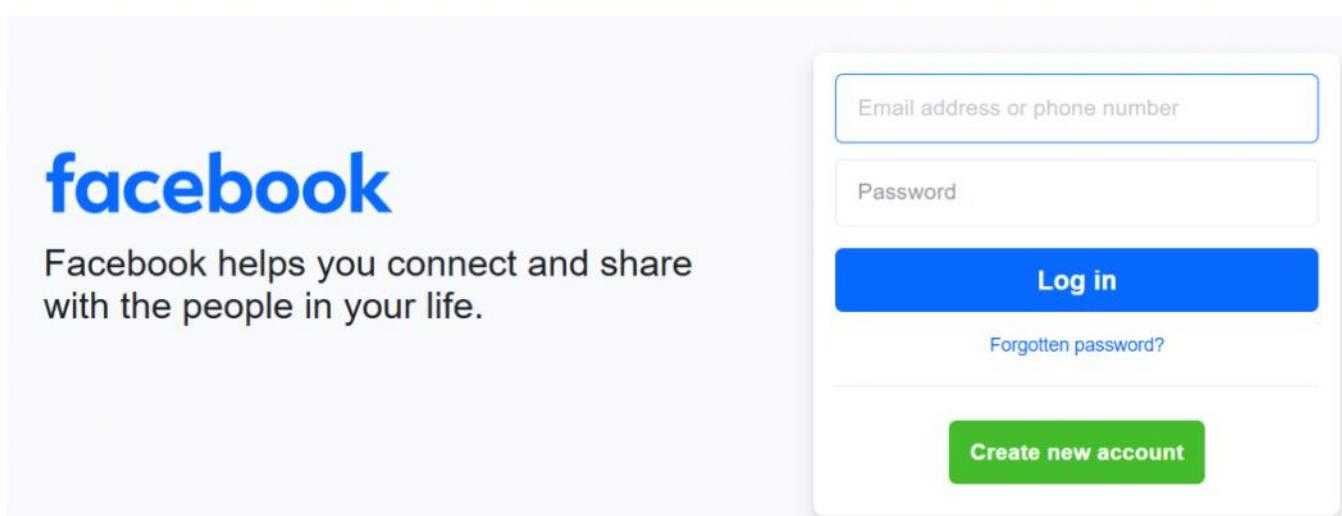
Python  Programming

# car_list = ['Creta', 'Safari', 'Sierra' , 'Harrier' , 'Duster'] ',' [Sept]
print('Mon','Tues','Wed', 'Thurs', 'Fri')

Mon Tues Wed Thurs Fri

print('Mon','Tues','Wed', 'Thurs', 'Fri' , sep = ' - ')
Mon - Tues - Wed - Thurs - Fri
```

Input



**Input() :** ¶

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

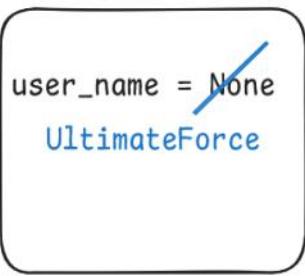
```
user_name = input("Enter your UserName") # String
print(user_name)
```

Enter your UserName **UltimareForce** User Defined  
UltimareForce

```
user_name = input("Enter your UserName") # String
print(user_name)
```

Enter your UserName **PardhanJi**  
PardhanJi

### Memory Diagram



```
# TypeCasting
val = input("Enter the Value : ") # String
print(val)
print(type(val)) # 'str'
```

```
Enter the Value : 21
21
<class 'str'>
```

```
val = int(input("Enter the Value : ")) # Int int('21') 21
print(val)
print(type(val))
```

```
Enter the Value : 21
21
<class 'int'>
```

```
# Operations [Arithmetic Operators]
val = int(input("Enter the Value : "))
print(val * 5)
print(type(val))
```

```
Enter the Value : 11
55
<class 'int'>
```

```
val = input("Enter the Value : ")
print(val * 5)
print(type(val))
```

```
Enter the Value : 11
111111111
<class 'str'>
```

```
val = input("Enter the Value : ")
print(val * 5)
print(type(val))
```

```
Enter the Value : a
aaaaa
<class 'str'>
```

```
val = input("Enter the Value : ")
print(val + '11')
print(type(val))
```

```
Enter the Value : 11
1111
<class 'str'>
```

```
x = int(input("Enter a number between 1 to 100:"))
x = x + 11
print(x)
```

```
Enter a number between 1 to 100: 66
77
```

Check\_Constraints -> Error Handling ['Raise'] [Try/Catch]

```
# String Pattern
# F-String
user_name = input("Enter the UserName : ")
designation = input('Enter Your Designation : ')
print(f"Hey , {user_name}! Welcome to the Team. You are hired as a {designation} role.")
```

```
Enter the UserName : Abhishek
Enter Your Designation : Senior Analyst
Hey , Abhishek! Welcome to the Team. You are hired as a Senior Analyst role.
```

```
# String Pattern
# F-String
user_name = input("Enter the UserName : ")
designation = input('Enter Your Designation : ')
print(f"Hey , {user_name}! Welcome to the Team. You are hired as a {designation} role.")
```

```
Enter the UserName : Annu Mishra
Enter Your Designation : Data Architect
Hey , Annu Mishra! Welcome to the Team. You are hired as a Data Architect role.
```

```
# MultiLine Comments
...
    This is a multiline Comments ....
    Put the important Stuff here relevant
    to your python Programming
...
print('Hi, EveryOne!')
```

```
Hi, EveryOne!
```

```
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: 22
Enter the Second_Value: 15
The final Result is : 330
The final Result is : 330
```

Memory Diagram

num1 = None 22  
num2 = None 15  
result = None 330

Console:

The final Result is : 330  
The final Result is : 330

```
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: 5
Enter the Second_Value: 16
The final Result is : 80
The final Result is : 90
```

```
print(result)
```

```
80
```

```
# Invoice Billing
item1 = float(input("Enter the price of Item1: "))
item2 = float(input("Enter the price of Item2: "))
item3 = float(input("Enter the price of Item3: "))
item4 = float(input("Enter the price of Item4: "))
item5 = float(input("Enter the price of Item5: "))
billing_amount = item1 + item2 + item3 + item4 + item5
gst_tax = billing_amount * 0.05
total_payment = billing_amount + gst_tax
print("Welcome Customer!, You final billing amount is" , total_payment)
print(f"Welcome Customer!, You final billing amount is {total_payment} ")
```

```
Enter the price of Item1: 99.99
Enter the price of Item2: 129.99
Enter the price of Item3: 499.99
Enter the price of Item4: 59.87
Enter the price of Item5: 83.55
Welcome Customer!, You final billing amount is 917.0595
Welcome Customer!, You final billing amount is 917.0595
```

```
print(billing_amount)
873.39
print(gst_tax)
43.6695
print(873.39 + 43.6695)
917.0595
```