

# Python for Deep Learning ICP\_1

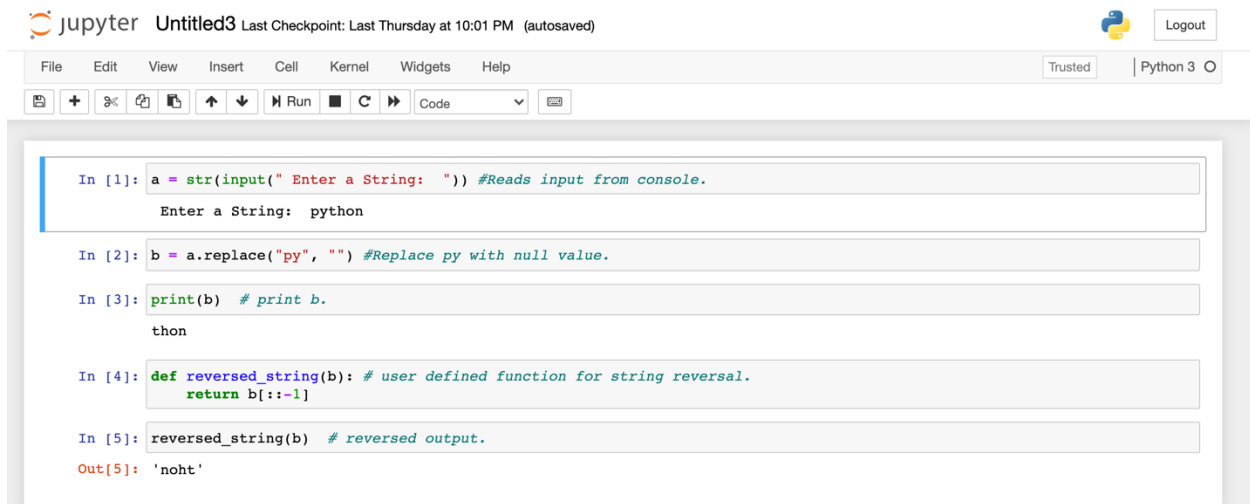
## Lesson 1: Basics of python

Vinay Kumar Reddy Meka  
16290945  
Vmqn4@mail.umkc.edu

## 1. Differences between Python 2.x and 3.x

Python 2.x	Python 3.x
1. It doesn't have improved integer division Eg : $5/2 = 2$	It has improved integer division Eg : $5/2 = 2.5$
2. It has print statement. Eg: Print "hello"	It has print function Eg : Print("hello")
3. Strings are stored as ASCII by default	Text Strings are Stored as Unicode by default
4. Xrange is used in 2.x	Xrange is not defined in 3.x instead we use Range
5. Error handling is not done	Error handling is doneaa

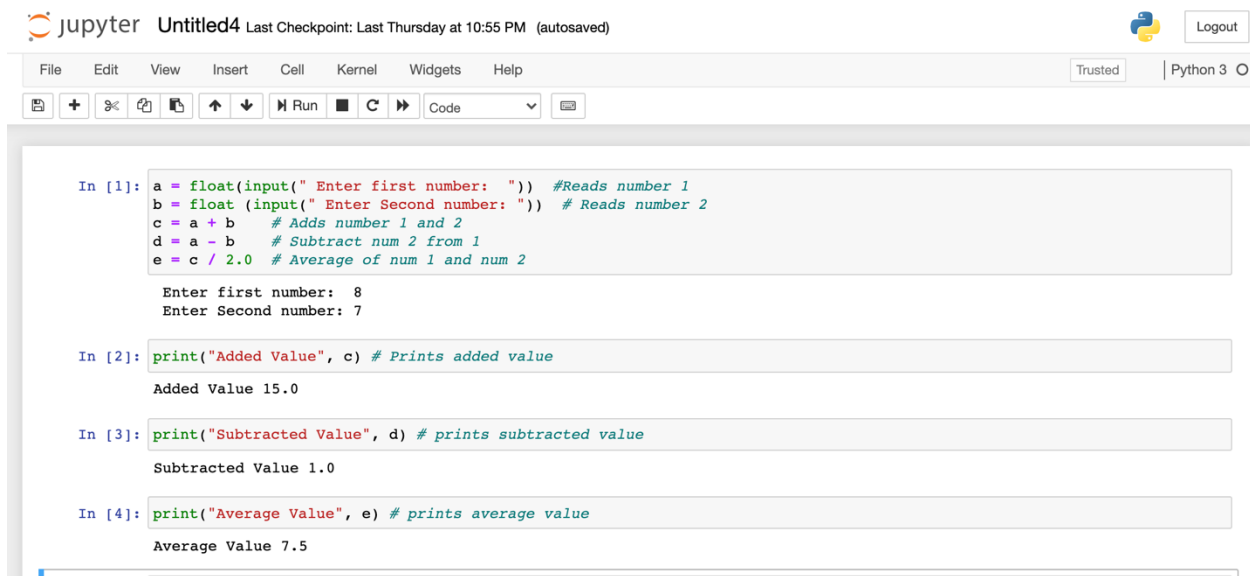
## Q 2.1



The image shows a Jupyter Notebook interface with the title 'Untitled3'. The last checkpoint was on Thursday at 10:01 PM (autosaved). The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a status bar showing 'Trusted' and 'Python 3'. The notebook contains five input cells and one output cell. The first cell takes user input 'python' and stores it in variable 'a'. The second cell replaces 'py' with an empty string in 'a', resulting in 'thon'. The third cell prints 'thon'. The fourth cell defines a function 'reversed\_string' that returns a string reversed. The fifth cell calls this function on 'thon', and the output is 'noht'.

```
In [1]: a = str(input(" Enter a String: ")) #Reads input from console.  
Enter a String: python  
  
In [2]: b = a.replace("py", "") #Replace py with null value.  
  
In [3]: print(b) # print b.  
thon  
  
In [4]: def reversed_string(b): # user defined function for string reversal.  
        return b[::-1]  
  
In [5]: reversed_string(b) # reversed output.  
Out[5]: 'noht'
```

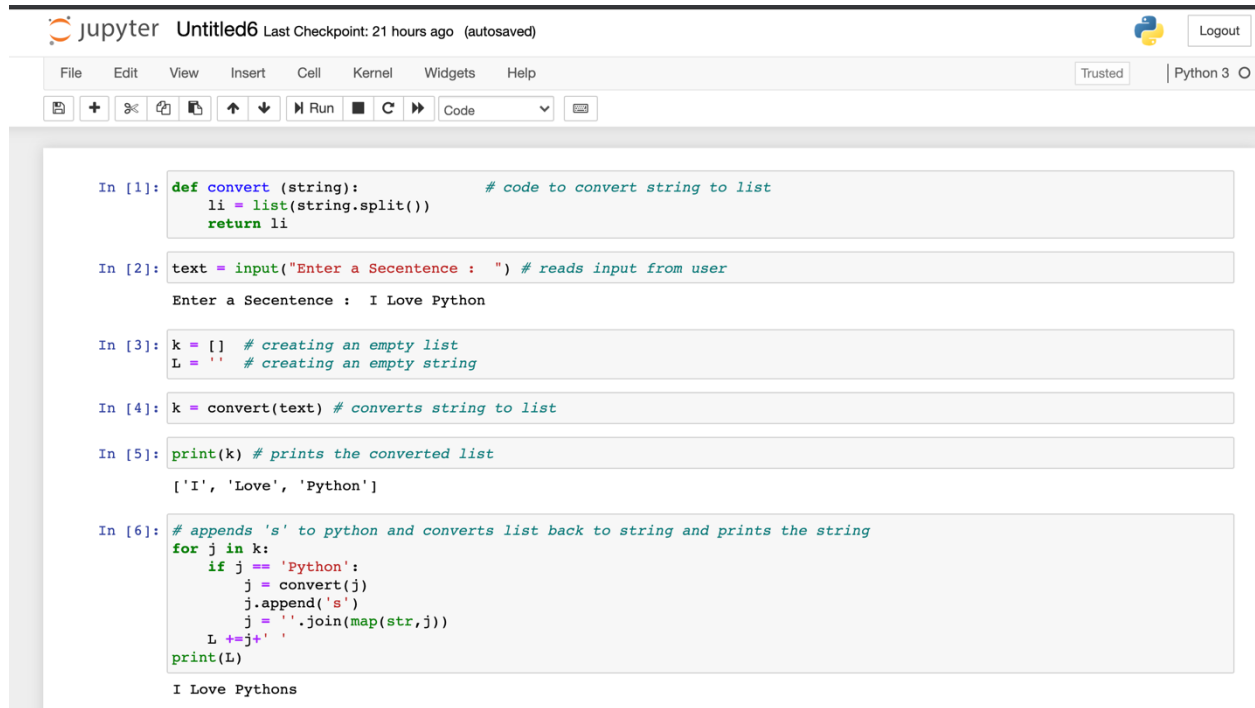
## Q. 2.2



The image shows a Jupyter Notebook interface with the title 'Untitled4'. The last checkpoint was on Thursday at 10:55 PM (autosaved). The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a status bar showing 'Trusted' and 'Python 3'. The notebook contains four input cells. The first cell takes two user inputs, '8' and '7', and stores them in variables 'a' and 'b'. It then calculates 'c' (a + b), 'd' (a - b), and 'e' (c / 2.0). The second cell prints 'Added Value' and '15.0'. The third cell prints 'Subtracted Value' and '1.0'. The fourth cell prints 'Average Value' and '7.5'.

```
In [1]: a = float(input(" Enter first number: ")) #Reads number 1  
b = float(input(" Enter Second number: ")) # Reads number 2  
c = a + b # Adds number 1 and 2  
d = a - b # Subtract num 2 from 1  
e = c / 2.0 # Average of num 1 and num 2  
Enter first number: 8  
Enter Second number: 7  
  
In [2]: print("Added Value", c) # Prints added value  
Added Value 15.0  
  
In [3]: print("Subtracted Value", d) # prints subtracted value  
Subtracted Value 1.0  
  
In [4]: print("Average Value", e) # prints average value  
Average Value 7.5
```

Q. 3



The image shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a code editor. The notebook is titled 'Untitled6' and shows a series of six input cells. The first cell defines a 'convert' function. The second cell takes user input. The third cell initializes variables. The fourth cell calls the 'convert' function. The fifth cell prints the result. The sixth cell appends 's' to the word 'python' and prints the final string.

```
In [1]: def convert (string):           # code to convert string to list
        li = list(string.split())
        return li

In [2]: text = input("Enter a Secentence : ") # reads input from user
Enter a Secentence :  I Love Python

In [3]: k = [] # creating an empty list
        L = '' # creating an empty string

In [4]: k = convert(text) # converts string to list

In [5]: print(k) # prints the converted list
['I', 'Love', 'Python']

In [6]: # appends 's' to python and converts list back to string and prints the string
for j in k:
    if j == 'Python':
        j = convert(j)
        j.append('s')
        j = ''.join(map(str,j))
    L +=j+' '
print(L)
I Love Pythons
```