PDPM IIITDM Jabalpur

Joint Winter FDP on Deep Learning & Applications
09-13 Dec., 2019

Introduction to Python

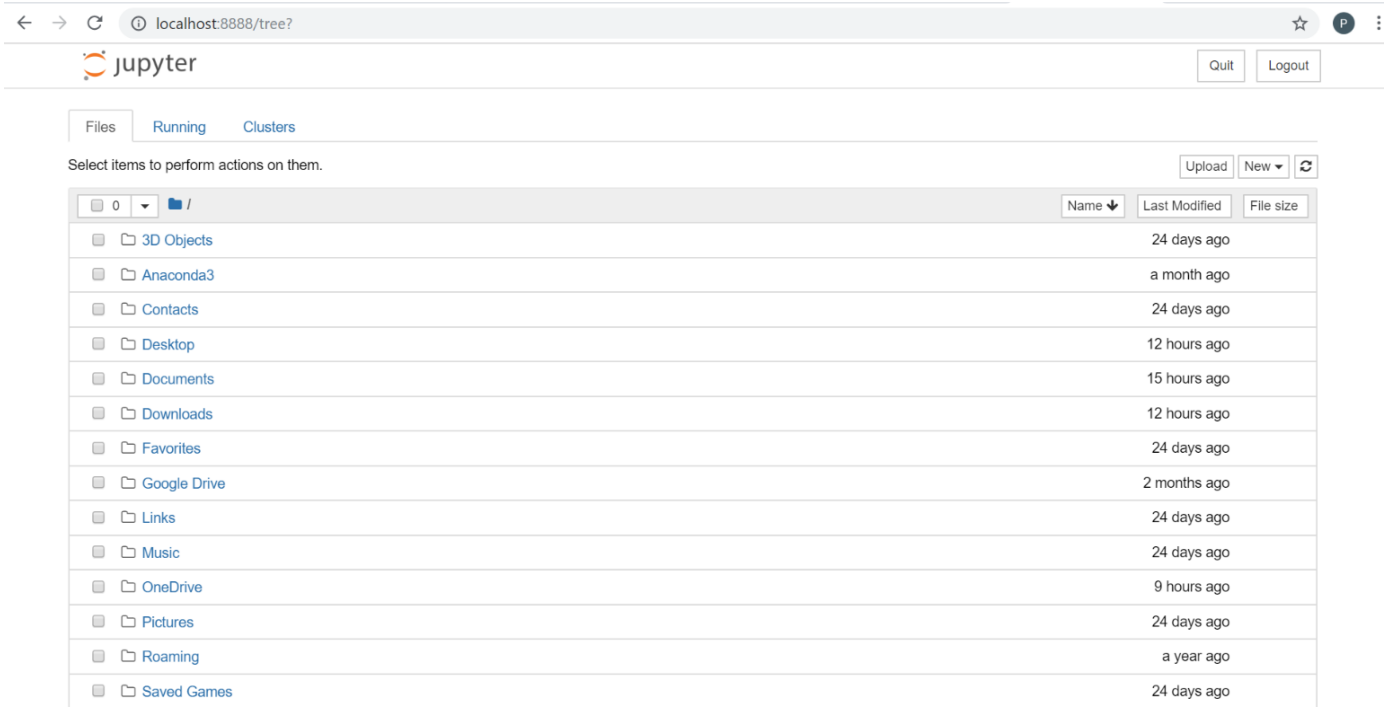
Before You Begin

Instruction to download and use ***ipynb*** file -

- Please download all the ***.ipynb*** and other format files from the google drive
- Put them all in a single folder
- Click on the start menu then type ***jupyter notebook***, its icon will appear
- Click on the icon , it will open **Jupyter Notebook**.

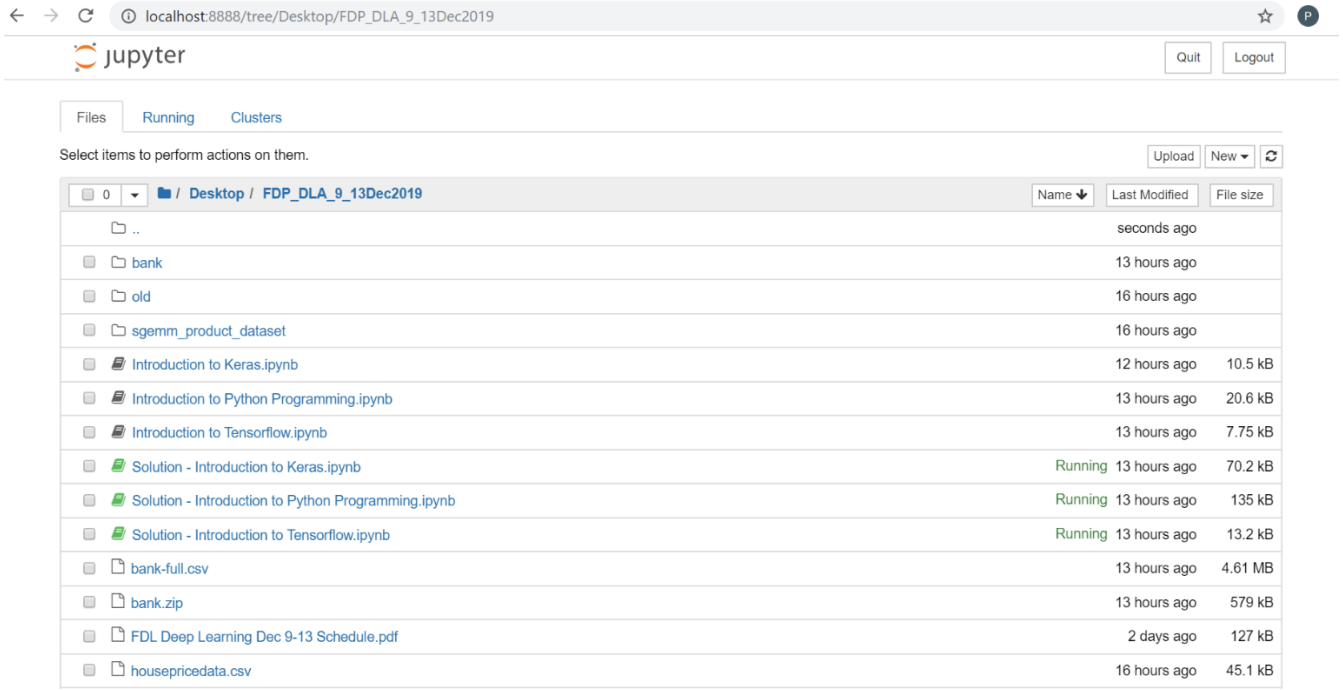
Before You Begin

- This window with file menu will appear.



Before You Begin

- Go to the respective folder and open ***.ipynb*** file



Now you are all set to read the Python Tutorial!

Let us begin!

Introduction to Python Programming

- Python is an high-level programming language under GPL developed by Guido van Rossum in 1985-90.
- Python is Portable and simple to code. It is widely used in data science. Well suited for coding deep learning algorithms using NumPy, Tensorflow and Keras.
- We will be using Jupyter notebook programming environment for writing codes.

Python Basics

- We can use ***print()*** function for displaying any data.

print ('Welcome to the Joint NKN FDP on Deep Learning & Applications!')

- We can take user's input through ***input()*** function

name = input('What is your name?\n')

Python is Indent Sensitive

- Python is case sensitive.
- Python is also indent sensitive.
- If you observe the 'for loop' and the 'if-else' statement, there is indent within the loop.
- For example if you run the following code, it will show "IndentationError".
- Run the cell to see the error and correct the code by putting indent.

```
for number in [1,10,20,30,40,50]:
print(number,"\n")
```

```
  File "<ipython-input-1-c5ba0b101069>", line 2
    print(number,"\n")
        ^
IndentationError: expected an indented block
```

Python Basics cont.....

- For single line comments # is use at the beginning of the line

  # We are creating a single line comment

- For multiple lines comment we can use “”” as open and closing tags.

  ””” Creating multiple line comments.
  to increase the readability of code
  ”””

If else Statement

- If else statement is similar to all other programming languages

```
num = input("number")
if int(num) > 0:
    print("Positive number")
elif int(num) == 0:
    print("Zero")
else:
    print("Negative number")
```

For and While loop

- *For loop:*

```
names = ['IIITDM Jabalpur', 'NIT Warangal', 'MNIT Jaipur', 'NIT Patna']
print('Participating Academies:')
for i, name in enumerate(names):
    print ("{iteration} : {name}".format(iteration=i+1, name=name))
```

- *While loop :*

```
names = ['IIITDM Jabalpur', 'NIT Warangal', 'MNIT Jaipur', 'NIT Patna']
value=0
print('Participating Academies:')
while value<len(names):
    print("{iteration} : {name}".format(iteration=value+1, name=names[value]))
    value+=1
```

Defining Function

- Just write **def function_name(parameters):**

```
def add(x,y):
    print ('Addition is :', x+y)

add(12,-10)
```

Dictionary

- "Dictionary is a useful data type built into Python, similar to “associative memories” or “associative arrays” in some languages. A dictionary is indexed by keys, Unlike a sequence, which is indexed by a range of numbers. A Key can be any immutable type, strings and numbers can always be keys.

```
prices = {'apple': 5, 'banana': 4}
my_purchase = {'apple': input('apple :'),'banana': input('banana :')}
grocery_bill = sum(prices[fruit] * int(my_purchase[fruit])
                   for fruit in my_purchase)
print ('Total Bill : %.2f' % grocery_bill +' rupees')
```

List

- List is a collection which is ordered and changeable.
  dynamic_languages = ['Python', 'Ruby', 'Groovy']
  dynamic_languages.append('Lisp')
  dynamic_languages

Exception Handling

- **Try** and **except** blocks are used for exception handling :

```
while True:
    try:
        x = int(input("Please enter a number: "))
        break
    except ValueError:
        print("Oops!  That was no valid number.  Try again...")
```

Classes in Python

- It is same as other language class

```
class BankAccount(object):
    def __init__(self, initial_balance=0):
        self.balance = initial_balance
 def withdraw(self, amount):
        self.balance -= amount
    def overdrawn(self):
        return self.balance < 0
my_account = BankAccount(100)
withdraw=input('Amount you want to withdraw : ')
my_account.withdraw(int(withdraw))
print (my_account.balance, my_account.overdrawn())
```

Numpy

- NumPy is a general-purpose fundamental package for scientific computing with Python.

```
import numpy as np
b1 = np.array([1,2,3,5,5])
b2 = np.array([4,5,6,7,7])
print("Adding two numpy arrays")
print(b1+b2)
print("multiply 3 in all the elements of numpy array")
print(b2 * 3)
print("No. of dimensions: ", b1.ndim)
```

Matplot Library

- Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers, and four graphical user interface toolkits (source: https://matplotlib.org/ , https://matplotlib.org/citing.html)

```
import numpy as np
import matplotlib.pyplot as plt
x = np.arange(0, 3 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)
plt.plot(x, y_sin)
plt.plot(x, y_cos)
plt.xlabel('x axis label')
plt.ylabel('y axis label')
plt.title('Sine and Cosine')
plt.legend(['Sine', 'Cosine'])
plt.show()
```

File Handling

Writing data into file

- file1 = open("sample1.txt", "w+")
- file1.write("Welcome to the first Python Tutorial Session!\n");

Reading data from a file

- file2 = open("sample3.txt", "r+")
- str = file2.read(80);

Closing the file

- file1.close()

Images using matplotlib

- Matplot library can be useful for reading images.

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
img=mpimg.imread(ict.png')
imgplot = plt.imshow(img)
plt.show()
```

Exercise

Exercise 1. Take radius input from user and then calculate the area of circle.

Exercise 2. Ask to user for amount he want to withdraw and then calculate the number of 2000, 500, 100, 50, 20 and 10 rupees notes required to fulfill the requirement.

Exercise 3. Create following list: data = [2,34,23,16,56,45,34,26,78,56,1,16]

Now delete the duplicate values and print the list in ascending order.

Exercise 4. Combine following dictionaries to create a new one.

```
dic1={'name1':'IIITDM', 'place1':'jabalpur'}
dic2={'name2':'MNIT', 'place2':'jaipur'}
dic3={'name3':'NIT', 'place3':'warangal'}
dic4={'name4':'IIT', 'place4':'patna'}
```

Exercise

Exercise 5. Write a Python program to convert temperatures to and from celsius, fahrenheit

Exercise 6. Write a Python program to print pattern for your name's first alphabet.

Exercise 7. From the following list create two seperate list of even and odd numbers in sorted order

data=[12,13,45,6,7,32,8,9,57,31,92,44]

Exercise 8. Compute the mean, standard deviation, and variance of a random numpy array.

Exercise

Exercise 9. In the following dataframe replace all the NaN values with Zero's and then plot the data

month	sell
jan	65211
feb	29424
march	29877
april	30990
may	61303
june	71781
july	NaN
aug	11072
sep	113702
oct	64731
nov	NaN
dec	NaN

THANK YOU