Python101

August 1, 2018

```
In [1]: # Simple addition
        5+98
```

```
Out[1]: 103
```

```
In [2]: #-v value addition
        -34+4
```

```
Out[2]: -30
```

```
In [3]: #Simple Substraction
        56-34
```

```
Out[3]: 22
```

```
In [4]: #Simple Division
        21/7
```

```
Out[4]: 3.0
```

```
In [5]: # Division without a floating value as answer
        21//7
```

```
Out[5]: 3
```

```
In [6]: #division with floating values
        52.777777/32.34444
```

```
Out[6]: 1.6317418697000166
```

```
In [7]: #simple multiplication
        56*4
```

```
Out[7]: 224
```

```
In [8]: #exponential function
        2**5
```

```
Out[8]: 32
```

```
In [10]: #order of operation follows PEDMAS="Parenthesis ,exponential,Division,multiplication,
         #addition-substraction

         5**2+(2+3)*4/2

Out[10]: 35.0

In [12]: ##variables--"its a container for some values."
         my_variable=30
         my_variable2=40
         my_variable

Out[12]: 30

In [13]: my_variable2

Out[13]: 40

In [14]: #python is a case sensitive language
         My_variable


         ---------------------------------------------------------------------------

         NameError                                 Traceback (most recent call last)

         <ipython-input-14-cae254a64efc> in <module>()
           1 #python is a case sensitive language
     ----> 2 My_variable


         NameError: name 'My_variable' is not defined


In [ ]: # It will throw in an error as M is in caps

In [15]: #we can perform arithmetic operations on the variables
         my_variable+my_variable2

Out[15]: 70

In [17]: # we can store this opertion in another variable
         my_addition=my_variable+my_variable2
         my_addition

Out[17]: 70

In [18]: #getting user_input
         value=input("Enter a Value:")
```

```
Enter a Value:2
```

```
In [19]: value
```

```
Out[19]: '2'
```

```
In [20]: # any value the user returns is a string so we need to change the data type to perfor
```

```
In [23]: # so what we can do is
         value=int(input("Enter a value :"))
```

```
Enter a value :50
```

```
In [24]: value+60
```

```
Out[24]: 110
```

```
In [25]: #COOL!!
```

```
In [5]: # List Tuples and Sets

        # List

        courses=["DBMS","Stats","Data Science in R","EDA"]
        print(courses)
```

```
['DBMS', 'Stats', 'Data Science in R', 'EDA']
```

```
In [6]: ## Indexing in python starts with 0

        courses[2]
```

```
Out[6]: 'Data Science in R'
```

```
In [7]: # A negetaive sign represents from the end of the list
        courses[-1]
```

```
Out[7]: 'EDA'
```

```
In [8]: courses[0:2]
```

```
Out[8]: ['DBMS', 'Stats']
```

```
In [9]: # here 0:2 means starting from 0th index  print all elements but not second index.
        # the below codes prints all the elements starting from index 1 to the end
        courses[1:]
```

```
Out[9]: ['Stats', 'Data Science in R', 'EDA']
```

```
In [10]: # A list might contain additional list or even numeric data
         my_list=[1,2,5,"foo","bar",["rock","paper","scissors"]]
         my_list

Out[10]: [1, 2, 5, 'foo', 'bar', ['rock', 'paper', 'scissors']]

In [11]: #subetting the last element of my_list
         my_list[-1]

Out[11]: ['rock', 'paper', 'scissors']

In [12]: # it returns the last element which is a list itself
         # subsetting a list within a list of my_list

         my_list[-1][0:2]

Out[12]: ['rock', 'paper']

In [13]: #adding values to list
         courses.append("ML")
         courses

Out[13]: ['DBMS', 'Stats', 'Data Science in R', 'EDA', 'ML']

In [14]: # list.insert takes 2 arguments first is the index where you want to add the data , n
         courses.insert(0,"Neural Networks")
         courses

Out[14]: ['Neural Networks', 'DBMS', 'Stats', 'Data Science in R', 'EDA', 'ML']

In [15]: #lets try adding two lists
         courses_2=["Inferential Statistics","AI"]
         courses.append(courses_2)
         courses

Out[15]: ['Neural Networks',
          'DBMS',
          'Stats',
          'Data Science in R',
          'EDA',
          'ML',
          ['Inferential Statistics', 'AI']]

In [16]: # As we can observe here list.append is just adding the whole list as a list in cours
         # want to add the elements of the second list as elements we use list.extend() method
         #We will use a method called list.pop() to remove the last element of the list to mak

In [17]: courses.pop()
         courses
```

```
Out[17]: ['Neural Networks', 'DBMS', 'Stats', 'Data Science in R', 'EDA', 'ML']

In [18]: courses.extend(courses_2)
         courses

Out[18]: ['Neural Networks',
          'DBMS',
          'Stats',
          'Data Science in R',
          'EDA',
          'ML',
          'Inferential Statistics',
          'AI']

In [19]: #Great
         #Now lets try removing elements from a list
         #we have already seen list.pop() in action which removes the last element of the list

         courses
         courses.remove('AI')
         courses

Out[19]: ['Neural Networks',
          'DBMS',
          'Stats',
          'Data Science in R',
          'EDA',
          'ML',
          'Inferential Statistics']

In [20]: courses.remove('Stats')

In [21]: courses

Out[21]: ['Neural Networks',
          'DBMS',
          'Data Science in R',
          'EDA',
          'ML',
          'Inferential Statistics']

In [22]: # We can see now courses dont have the elements AI and Stats

In [24]: ## Now lets see some examples of sort
         # now suppose we want to reverse the elements of the List courses
         courses.reverse()
         courses
```

```
Out[24]: ['Inferential Statistics',
          'ML',
          'EDA',
          'Data Science in R',
          'DBMS',
          'Neural Networks']

In [26]: # As we can see above the elements of courses has been reversed
         #now lets use list.sort() method
         courses.sort()
         courses

Out[26]: ['DBMS',
          'Data Science in R',
          'EDA',
          'Inferential Statistics',
          'ML',
          'Neural Networks']

In [28]: # We can see the list have been sorted alphabetically.
         # now suppose we want it (courses) to be sorted alphabetically in descending order we
         courses.sort(reverse=True)
         courses

Out[28]: ['Neural Networks',
          'ML',
          'Inferential Statistics',
          'EDA',
          'Data Science in R',
          'DBMS']

In [29]: # Cool

In [30]: # now we will use a function to short the list
         courses_sorted=sorted(courses)
         courses_sorted

Out[30]: ['DBMS',
          'Data Science in R',
          'EDA',
          'Inferential Statistics',
          'ML',
          'Neural Networks']

In [37]: # lets print out the index of elements in courses
         courses.index('EDA')

Out[37]: 3

In [38]: #lets now see if an element exists in a list (courses) or not
         'EDA' in courses
```

```
Out[38]: True

In [39]: "AI" in courses

Out[39]: False

In [31]: ### Lets see some more functions
         #lets print out the minimum value from the list
         my_list=[23,45,78,32,12,67]

         min(my_list)

Out[31]: 12

In [32]: # now  lets print maximum

         max(my_list)

Out[32]: 78

In [33]: # and now for the sum of the elements in the list

         sum(my_list)

Out[33]: 257

In [48]: # We can print the elements of a list using a for loop
         for elements in courses:
            print(elements)

Neural Networks
ML
Inferential Statistics
EDA
Data Science in R
DBMS


In [50]: # we can print the index and the elements of list using enumerate
         for index,items in enumerate(courses):
             print (index,items)

0 Neural Networks
1 ML
2 Inferential Statistics
3 EDA
4 Data Science in R
5 DBMS
```

In [52]:
```
#  print from a particular index

        for index,items in enumerate(courses,start=2):
            print(index,items)
```

```
2 Neural Networks
3 ML
4 Inferential Statistics
5 EDA
6 Data Science in R
7 DBMS
```

In [ ]: