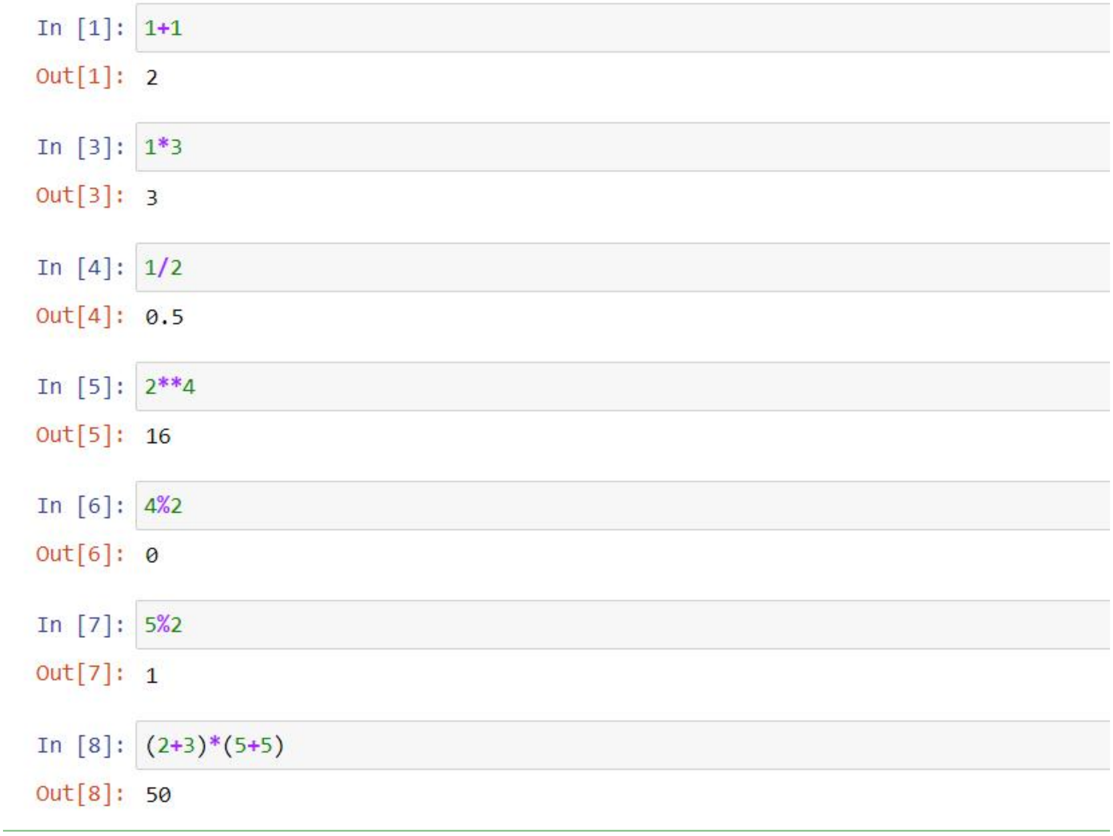
Link Github:

LAB 01

III. Python basics:

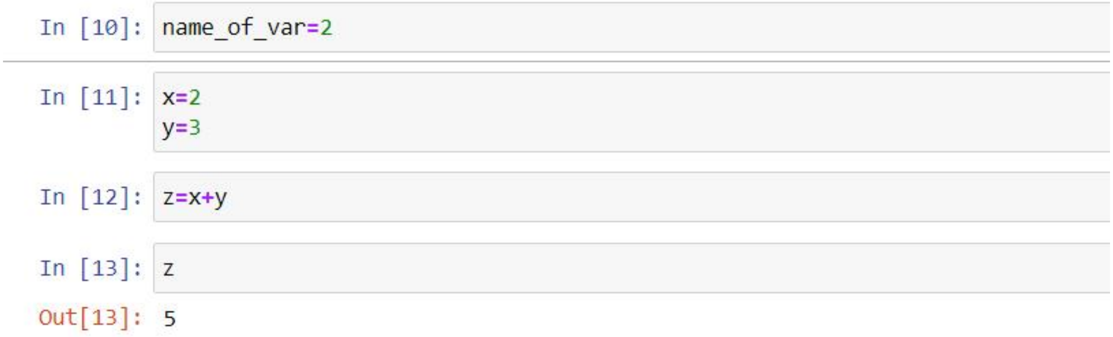
1. Data types:

a. Number:



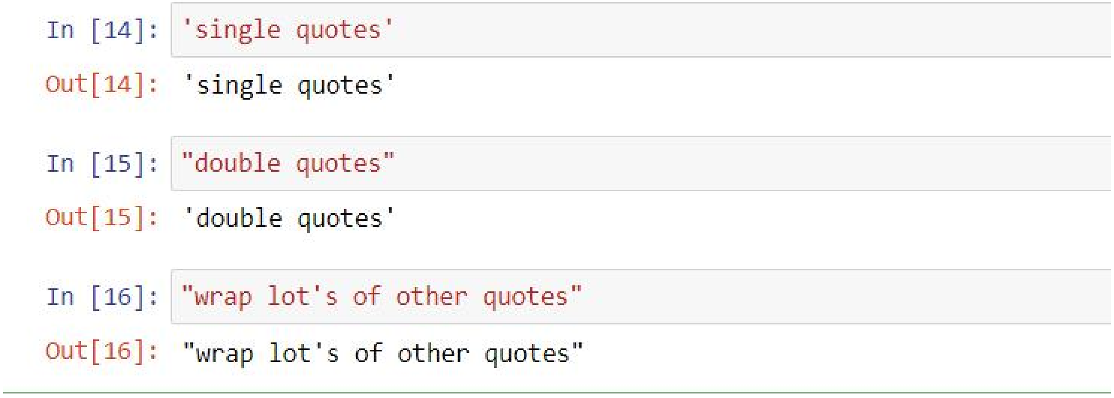
```
In [1]: 1+1
Out[1]: 2

In [3]: 1*3
Out[3]: 3

In [4]: 1/2
Out[4]: 0.5

In [5]: 2**4
Out[5]: 16

In [6]: 4%2
Out[6]: 0

In [7]: 5%2
Out[7]: 1

In [8]: (2+3)*(5+5)
Out[8]: 50
```

b. Variable assignment:



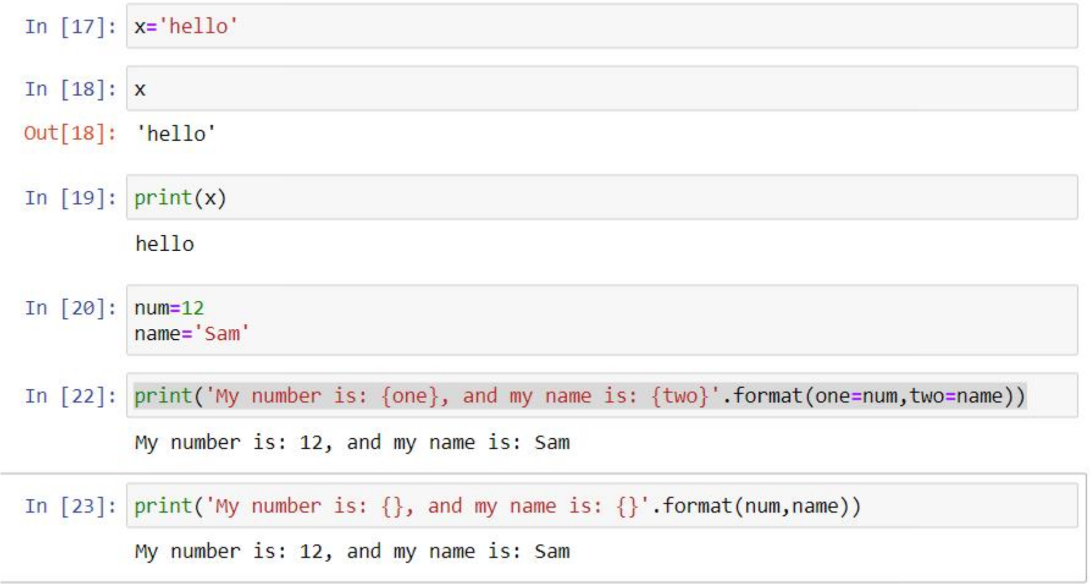
```
In [10]: name_of_var=2

In [11]: x=2
         y=3

In [12]: z=x+y

In [13]: z
Out[13]: 5
```

c. Strings:



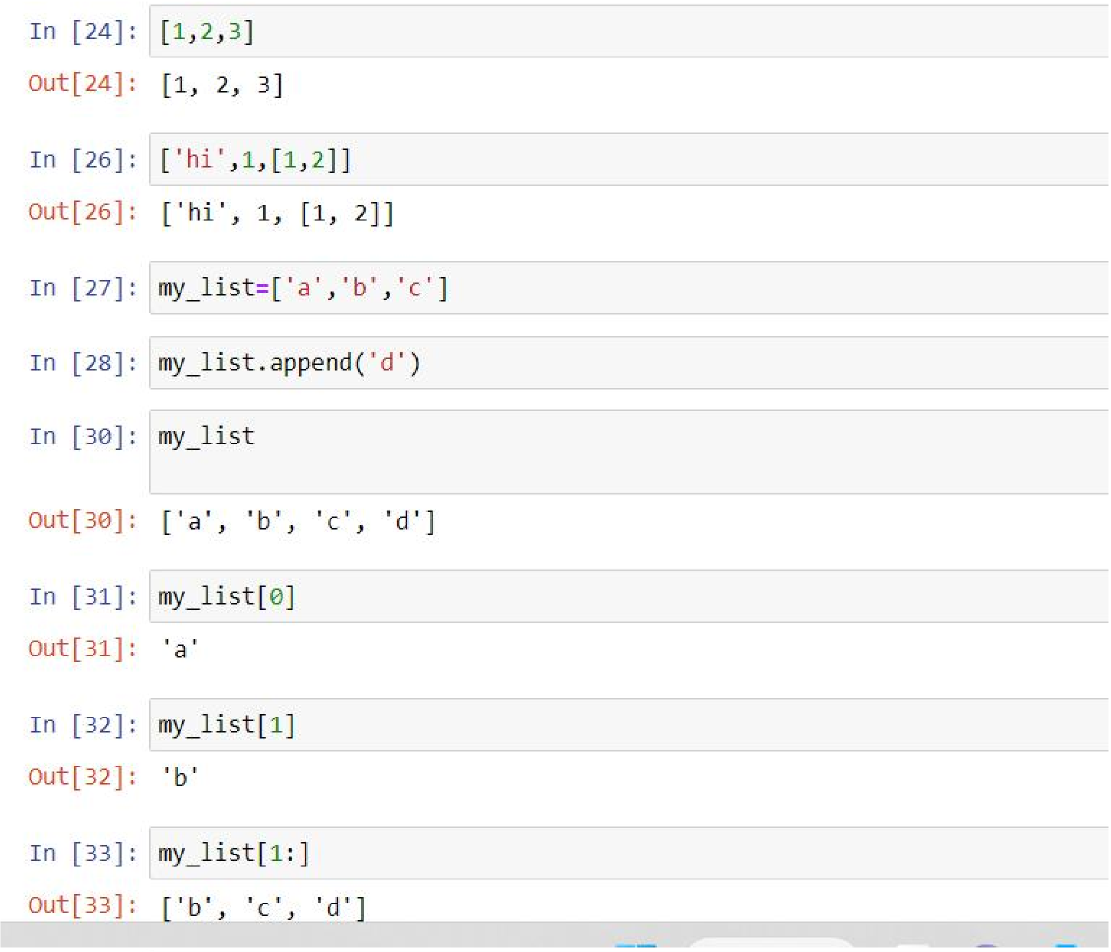
```
In [14]: 'single quotes'

Out[14]: 'single quotes'

In [15]: "double quotes"

Out[15]: 'double quotes'

In [16]: "wrap lot's of other quotes"

Out[16]: "wrap lot's of other quotes"
```

d. Printing:

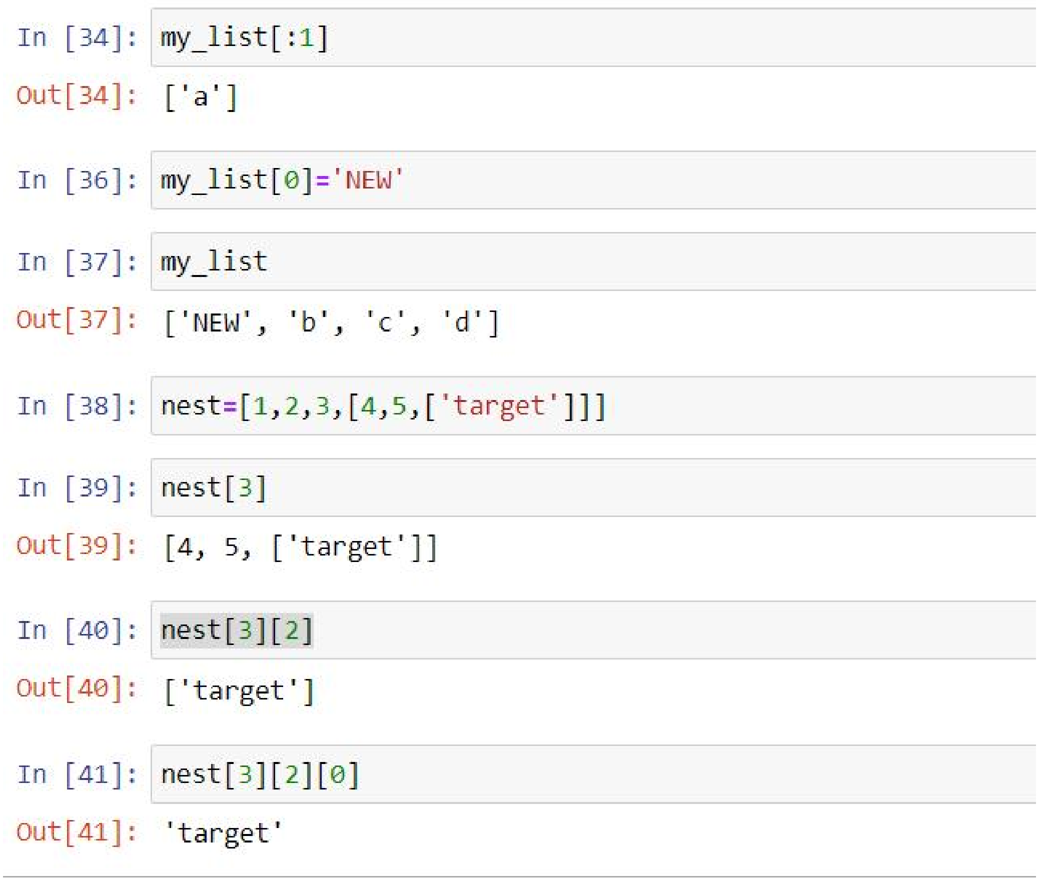


```
In [17]: x='hello'

In [18]: x

Out[18]: 'hello'

In [19]: print(x)

         hello

In [20]: num=12
         name='Sam'

In [22]: print('My number is: {one}, and my name is: {two}'.format(one=num,two=name))

         My number is: 12, and my name is: Sam

In [23]: print('My number is: {}, and my name is: {}'.format(num,name))

         My number is: 12, and my name is: Sam
```

e. Lists:

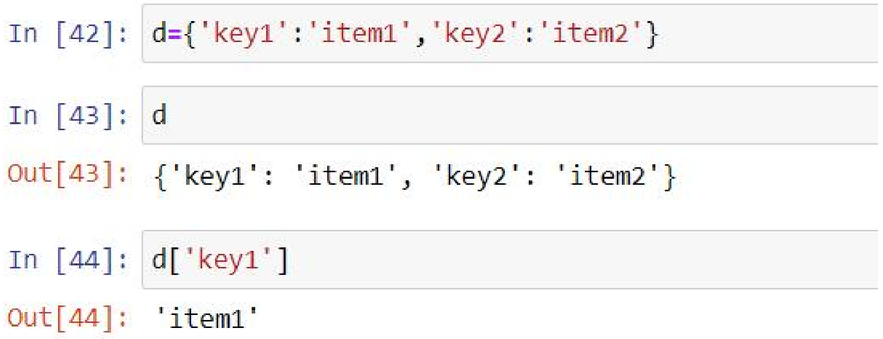


```
In [24]: [1,2,3]

Out[24]: [1, 2, 3]

In [26]: ['hi',1,[1,2]]

Out[26]: ['hi', 1, [1, 2]]

In [27]: my_list=['a','b','c']

In [28]: my_list.append('d')

In [30]: my_list

Out[30]: ['a', 'b', 'c', 'd']

In [31]: my_list[0]

Out[31]: 'a'

In [32]: my_list[1]

Out[32]: 'b'

In [33]: my_list[1:]

Out[33]: ['b', 'c', 'd']
```



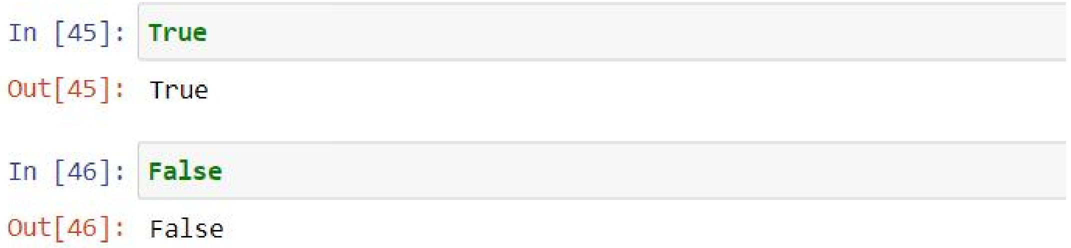
```
In [34]: my_list[:1]

Out[34]: ['a']

In [36]: my_list[0]='NEW'

In [37]: my_list

Out[37]: ['NEW', 'b', 'c', 'd']

In [38]: nest=[1,2,3,[4,5,['target']]]

In [39]: nest[3]

Out[39]: [4, 5, ['target']]

In [40]: nest[3][2]

Out[40]: ['target']

In [41]: nest[3][2][0]

Out[41]: 'target'
```

f. Dictionaries:



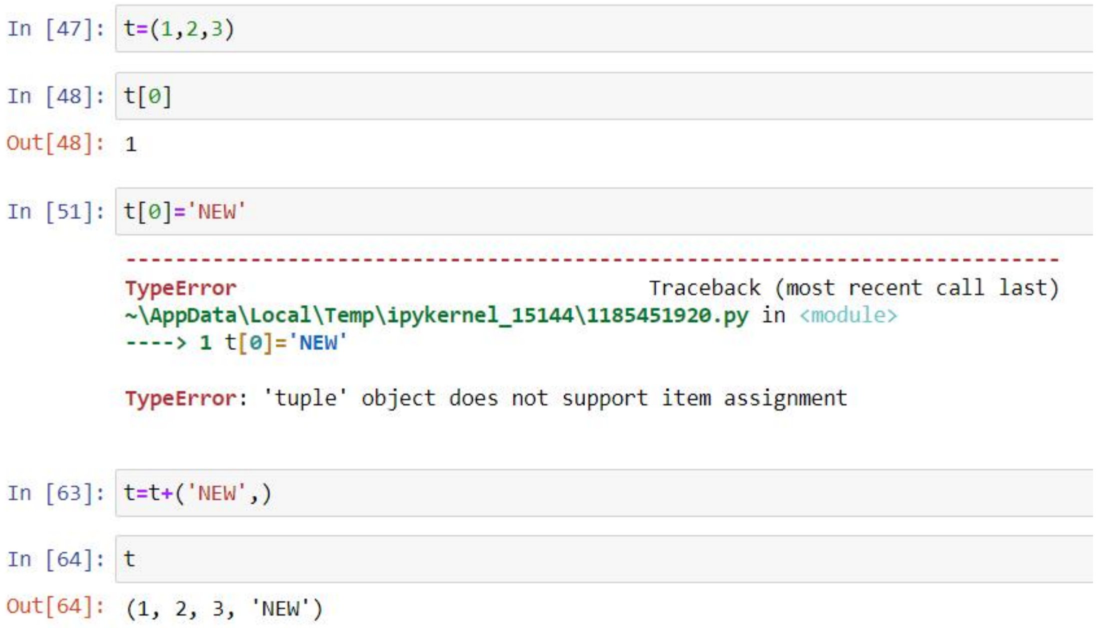
```
In [42]: d={'key1':'item1','key2':'item2'}

In [43]: d
Out[43]: {'key1': 'item1', 'key2': 'item2'}

In [44]: d['key1']
Out[44]: 'item1'
```

g. Booleans:



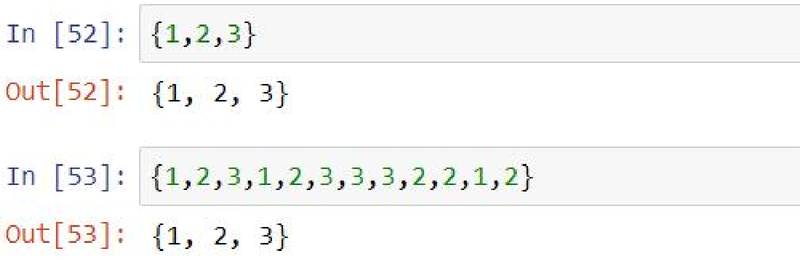
```
In [45]: True
Out[45]: True

In [46]: False
Out[46]: False
```

h. Tuples:



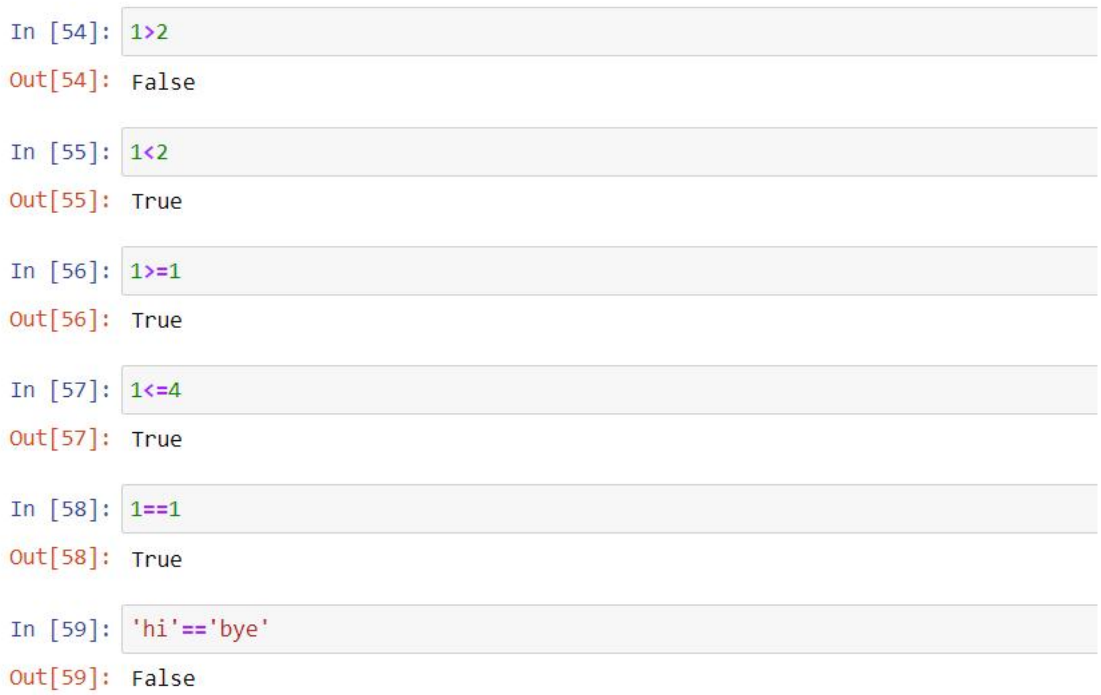
```
In [47]: t=(1,2,3)

In [48]: t[0]
Out[48]: 1

In [51]: t[0]='NEW'

         ---------------------------------------------------------------------------
         TypeError                                 Traceback (most recent call last)
         ~\AppData\Local\Temp\ipykernel_15144\1185451920.py in <module>
         ----> 1 t[0]='NEW'

         TypeError: 'tuple' object does not support item assignment

In [63]: t=t+('NEW',)

In [64]: t
Out[64]: (1, 2, 3, 'NEW')
```

i. Sets:



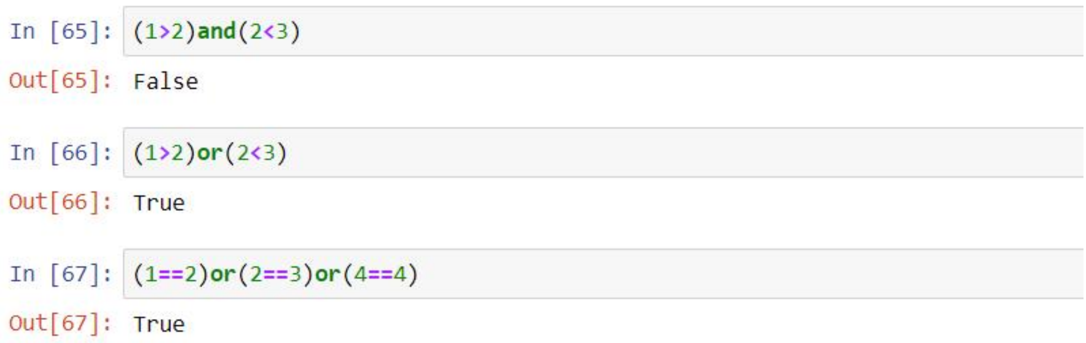
```
In [52]: {1,2,3}
Out[52]: {1, 2, 3}

In [53]: {1,2,3,1,2,3,3,3,2,2,1,2}
Out[53]: {1, 2, 3}
```

j. Comparison operator:



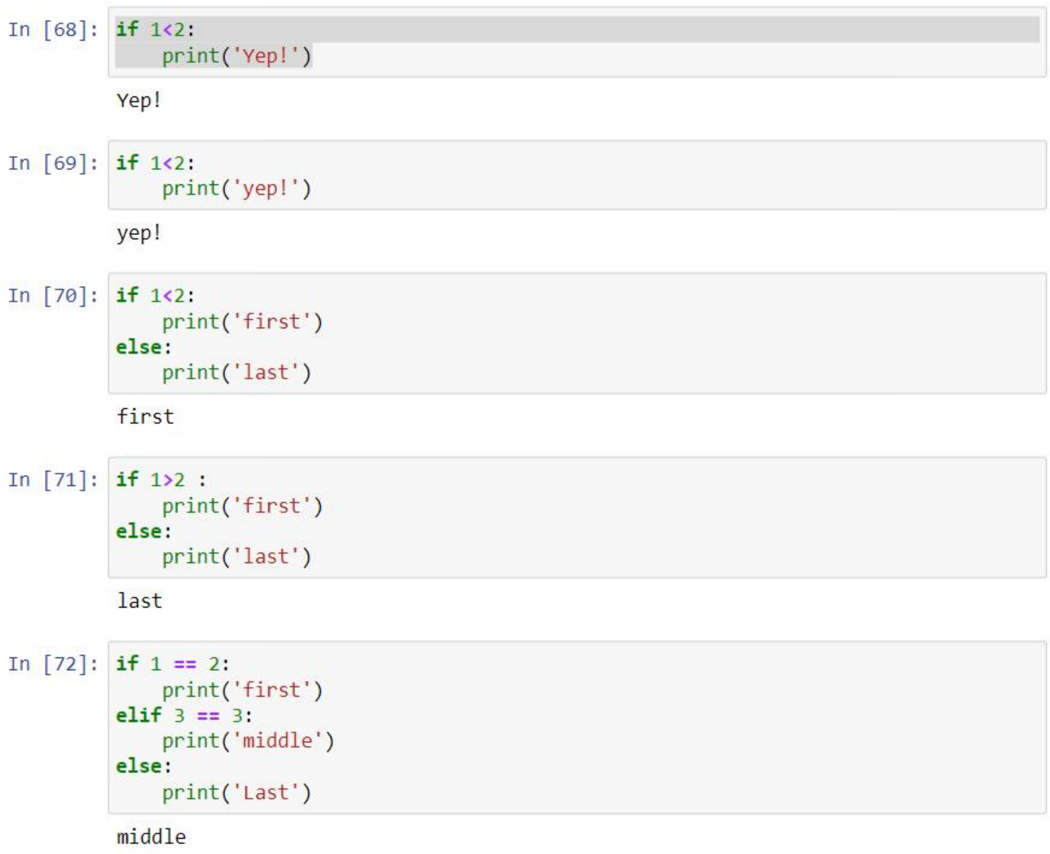
```
In [54]: 1>2
Out[54]: False

In [55]: 1<2
Out[55]: True

In [56]: 1>=1
Out[56]: True

In [57]: 1<=4
Out[57]: True

In [58]: 1==1
Out[58]: True

In [59]: 'hi'=='bye'
Out[59]: False
```

k. Logic Operator:



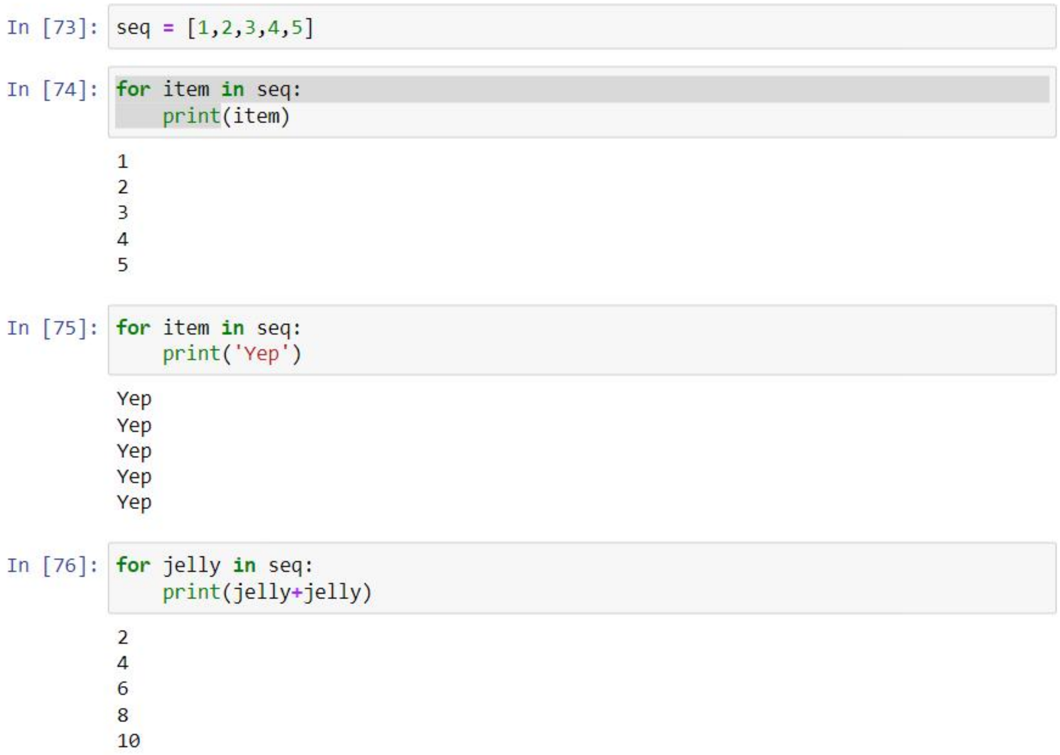
```
In [65]: (1>2)and(2<3)
Out[65]: False

In [66]: (1>2)or(2<3)
Out[66]: True

In [67]: (1==2)or(2==3)or(4==4)
Out[67]: True
```

l. If, elseif, else Statement:



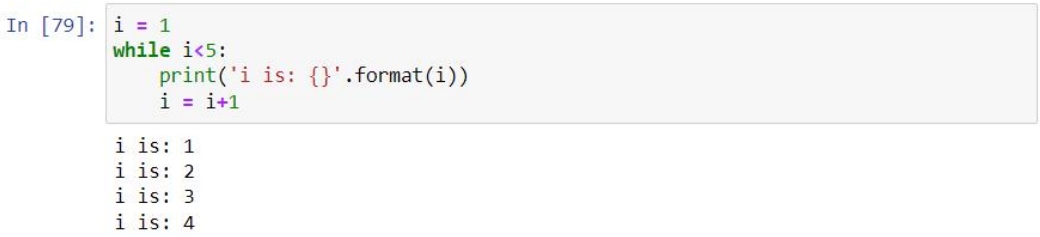
```
In [68]: if 1<2:
             print('Yep!')

         Yep!

In [69]: if 1<2:
             print('yep!')

         yep!

In [70]: if 1<2:
             print('first')
         else:
             print('last')

         first

In [71]: if 1>2 :
             print('first')
         else:
             print('last')

         last

In [72]: if 1 == 2:
             print('first')
         elif 3 == 3:
             print('middle')
         else:
             print('Last')

         middle
```

m. For loops:



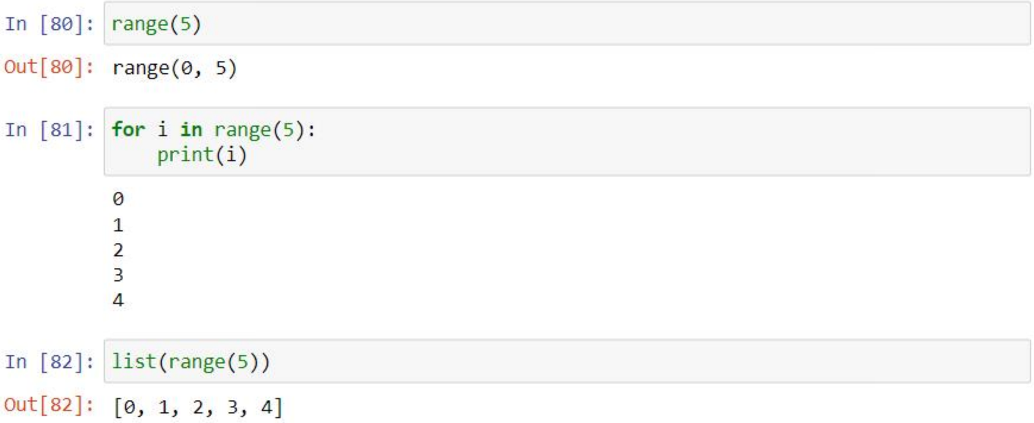
```
In [73]: seq = [1,2,3,4,5]

In [74]: for item in seq:
             print(item)

         1
         2
         3
         4
         5

In [75]: for item in seq:
             print('Yep')

         Yep
         Yep
         Yep
         Yep
         Yep

In [76]: for jelly in seq:
             print(jelly+jelly)

         2
         4
         6
         8
         10
```

n. While loops:



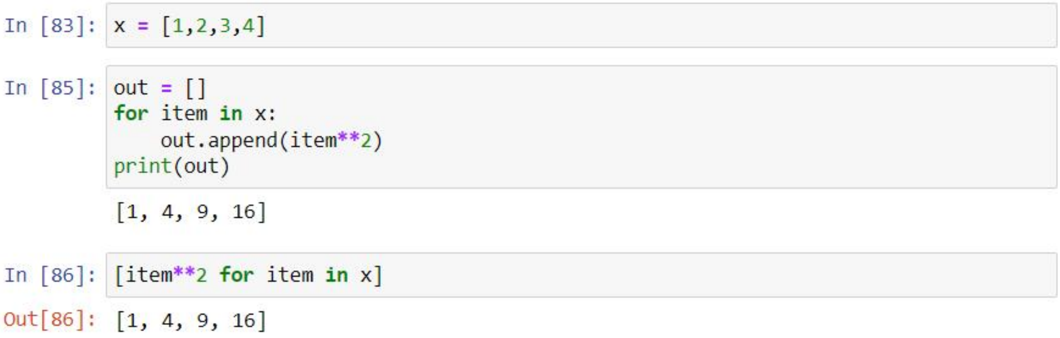
```
In [79]: i = 1
         while i<5:
             print('i is: {}'.format(i))
             i = i+1

         i is: 1
         i is: 2
         i is: 3
         i is: 4
```

o. Ranges:



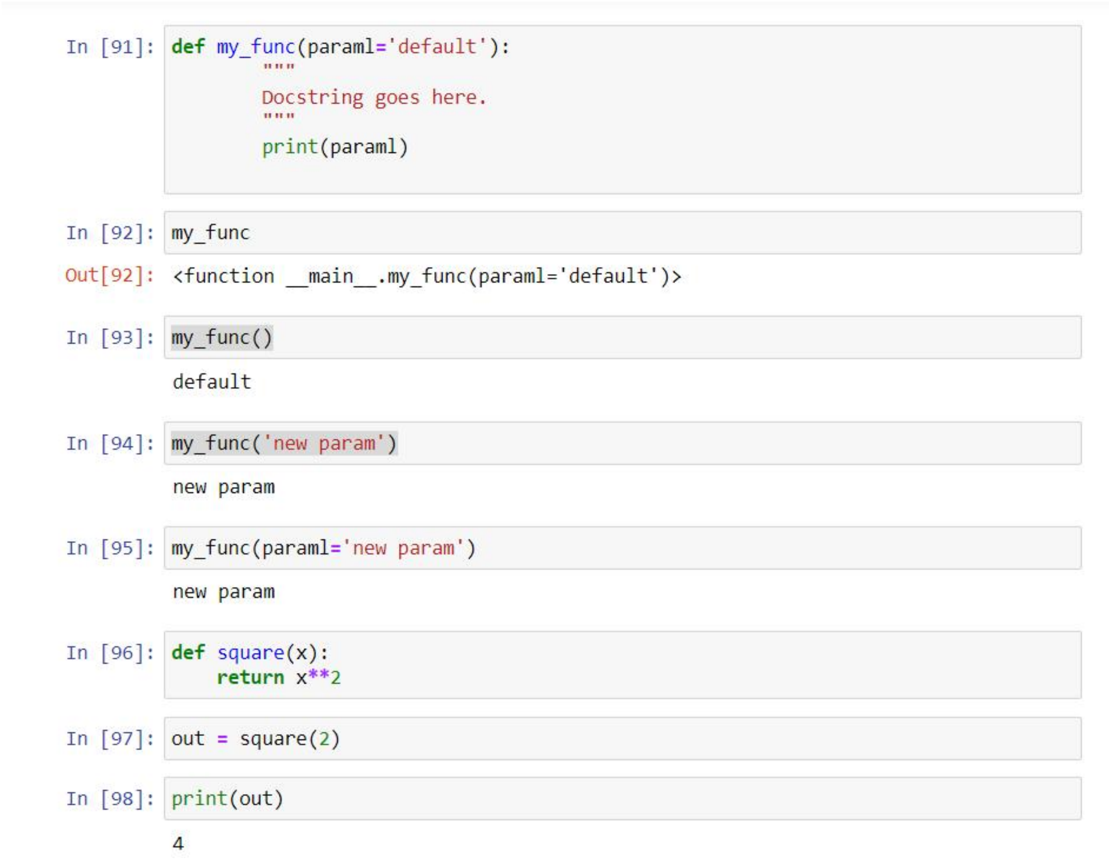
```
In [80]: range(5)

Out[80]: range(0, 5)

In [81]: for i in range(5):
             print(i)

         0
         1
         2
         3
         4

In [82]: list(range(5))

Out[82]: [0, 1, 2, 3, 4]
```

p. List comprehensions:



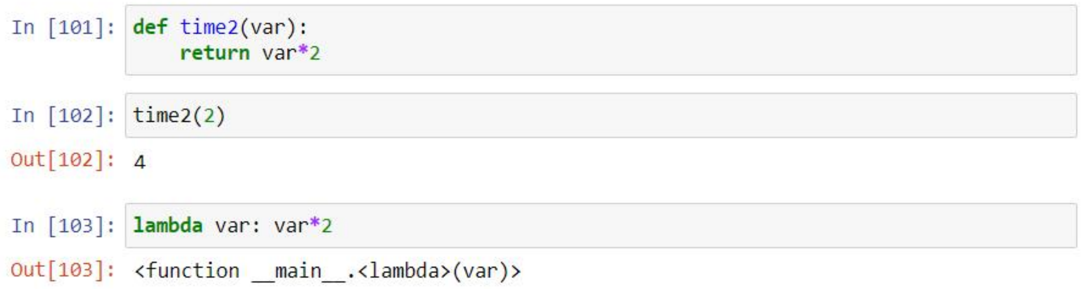
```
In [83]: x = [1,2,3,4]

In [85]: out = []
         for item in x:
             out.append(item**2)
         print(out)

         [1, 4, 9, 16]

In [86]: [item**2 for item in x]

Out[86]: [1, 4, 9, 16]
```

q. Functions:



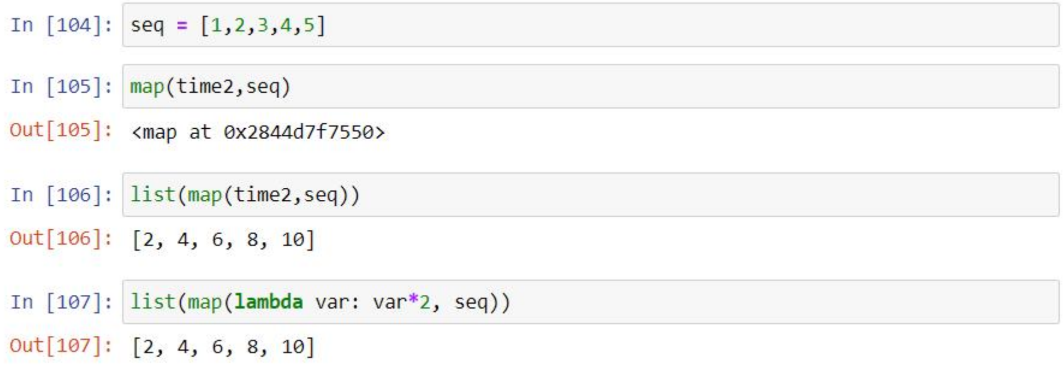
```
In [91]: def my_func(paraml='default'):
                 """
                 Docstring goes here.
                 """
                 print(paraml)

In [92]: my_func

Out[92]: <function __main__.my_func(paraml='default')>

In [93]: my_func()

         default

In [94]: my_func('new param')

         new param

In [95]: my_func(paraml='new param')

         new param

In [96]: def square(x):
             return x**2

In [97]: out = square(2)

In [98]: print(out)

         4
```

r. Lambda expression:



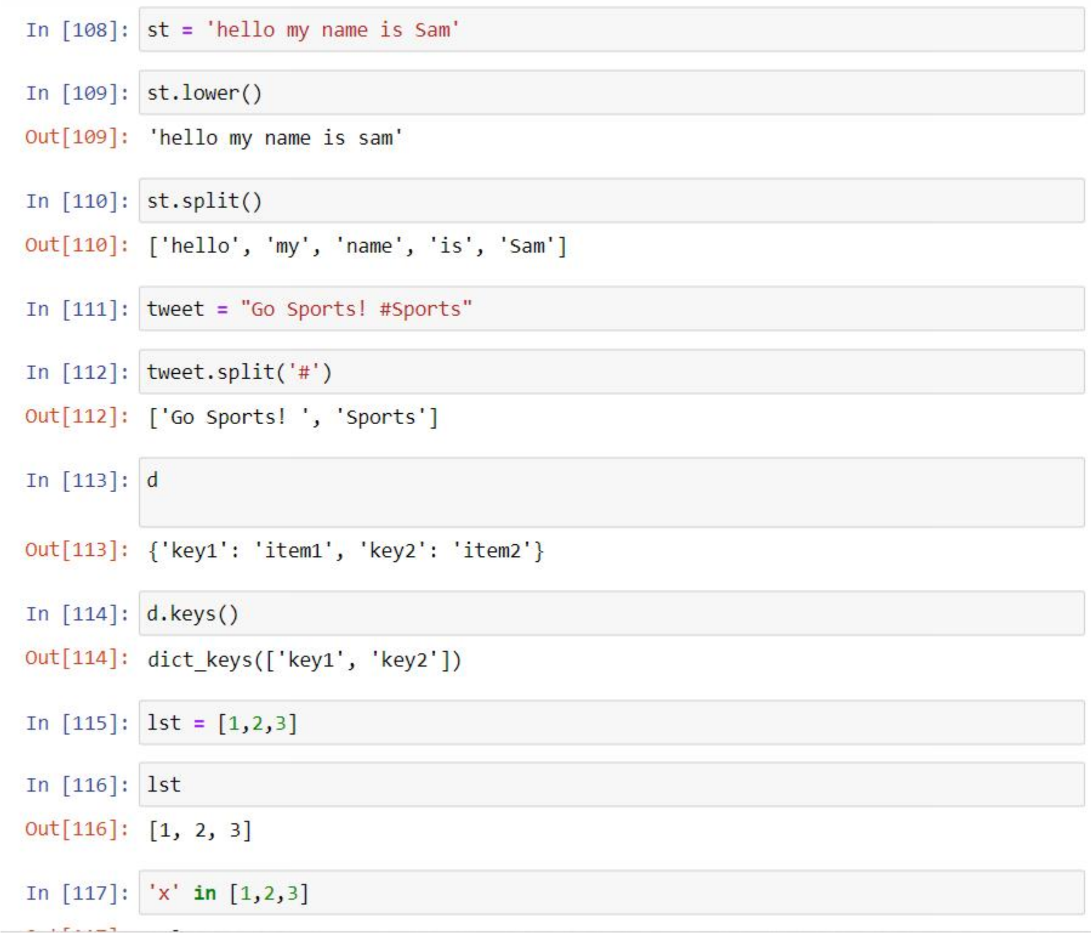
```
In [101]: def time2(var):
              return var*2

In [102]: time2(2)

Out[102]: 4

In [103]: lambda var: var*2

Out[103]: <function __main__.<lambda>(var)>
```

s. Map and filter:

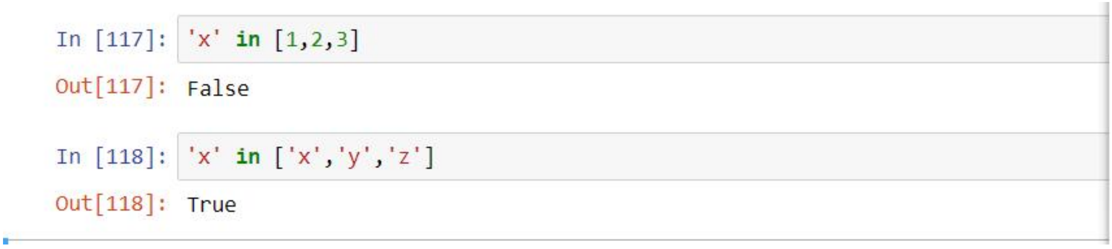


```
In [104]: seq = [1,2,3,4,5]

In [105]: map(time2,seq)

Out[105]: <map at 0x2844d7f7550>

In [106]: list(map(time2,seq))

Out[106]: [2, 4, 6, 8, 10]

In [107]: list(map(lambda var: var*2, seq))

Out[107]: [2, 4, 6, 8, 10]
```

t. Methods:



```
In [108]: st = 'hello my name is Sam'

In [109]: st.lower()
Out[109]: 'hello my name is sam'

In [110]: st.split()
Out[110]: ['hello', 'my', 'name', 'is', 'Sam']

In [111]: tweet = "Go Sports! #Sports"

In [112]: tweet.split('#')
Out[112]: ['Go Sports! ', 'Sports']

In [113]: d

Out[113]: {'key1': 'item1', 'key2': 'item2'}

In [114]: d.keys()
Out[114]: dict_keys(['key1', 'key2'])

In [115]: lst = [1,2,3]

In [116]: lst
Out[116]: [1, 2, 3]

In [117]: 'x' in [1,2,3]
```



```
In [117]: 'x' in [1,2,3]
```

```
Out[117]: False

In [118]: 'x' in ['x','y','z']
Out[118]: True
```

IV. Python basics - exercises:

What is 7 to the power of 4?