

Python Tutorials for Absolute Beginners in Hindi

Playlist Link:

#6 Comments, Escape, Sequences & Print ()

<https://youtu.be/uhUgIT68CoU> (<https://youtu.be/uhUgIT68CoU>)

```
In [18]: print ('c:\\ncert_books_download \\t ', end='')
print ('download NCERT books from this link')
print ('''to know more about escape sequences in python, google
"escape sequences in python" and go to the official website for python 3''')
exit
```

```
c:\ncert_books_download          download NCERT books from this link
to know more about escape sequences in python, google
"escape sequences in python" and go to the official website for python 3
```

Out[18]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#7 Variables, Datatype & Typecasting

<https://youtu.be/z1-zfCvxybl> (<https://youtu.be/z1-zfCvxybl>)

```
var1='Kolkata'
var2=8
var3=9.8
print (type (var1), type (var2), type (var3))
var4=' rocks now'
print (var1 + var4)
x= '573'
y= '42'
print (int (x) + int (y))
print (5*str(int (x) + int (y)))
exit
print ('put value of x, the output will be 5x')
x= input ()
print (int(x)*5)
#x is always taken as a str in input function.
# So in case of mathematical operation we need to convert str into int or float
exit
print ('\n')
print ('additive calculator, 2 numbers')
print ('enter 1st number')
x= input ()
print ('enter 2nd number')
y= input ()
print ('the addition is ', float(x)+float(y))
exit
```

#8 Strings & Related Functions

<https://youtu.be/IPZn7zcGXQo> (<https://youtu.be/IPZn7zcGXQo>)

```

In [19]: mystr = 'ABCDEFGHIIJKLMNOPQRSTUVWXYZ'
print ( len(mystr),'\n', mystr[0:6], mystr[0:9], '\n',mystr[0:], mystr[:7])
# Len (mystr) gives the total number of characters present in the string. Lower Limit of string is include
print (mystr[1:7:3])
# start from 1 and end at (7-1). The Last term denotes that (3-1) entries are skipped. The third term is u
print (mystr[2::2], mystr[:6:4], mystr[:,], mystr[:,16])
print (mystr[-5:-1], mystr[-4465:-2])
#
"""
a negative sign means that counting will be occurred from the opposite direction.
The sequence of counting : (for len=7)
'0,1,2,3,4,5,6' or '-7,-6,-5,-4,-3,-2,-1'
"""
print (mystr[134:0:-1], mystr[-1:-5:-1])
# If we use negative sign in the 3rd term, we will get reversed terms in the output. In this case we must
print ('Functions')
print (mystr.isalnum(), mystr.isalpha())
#bif there remains any space in the string, we will get False in the output using functions isalnum and is
print (mystr.endswith("xyz"), mystr.endswith("XYZ"))
print (mystr.count('A'))
# It counts no. of desired characters present in the string.
print (mystr.capitalize())
# it makes the 1st word capital and the others will be in small letters.
print (mystr.find('IJK'))
# it gives the position of the searched entry in the string.
print (mystr.lower(), mystr.upper())
# Lower makes all the characters small, while upper makes all the characters capital.
print (mystr.replace('DEF', 'def'))
# by using replace, the 1st term in the string will be replaced by the second term.
print (mystr)
# After all these operations the real string remains unchanged.
print ("for more string functions, google 'string functions in python'")
exit

```

```

26
ABCDEF ABCDEFGHI
ABCDEFGHIIJKLMNOPQRSTUVWXYZ ABCDEFG
BE
CEGIKMQSUWY AE ABCDEFGHIIJKLMNOPQRSTUVWXYZ AQ
VWXY ABCDEFGHIIJKLMNOPQRSTUVWXYZ
ZYXWVUTSRQPONMLKJIHGFEDCB ZYXW
Functions
True True
False True
1
Abcdefghijklmnopqrstuvwxyz
8
abcdefghijklmnopqrstuvwxyz ABCDEFGHIIJKLMNOPQRSTUVWXYZ
ABCdefGHIIJKLMNOPQRSTUVWXYZ
ABCDEFHIIJKLMNOPQRSTUVWXYZ
for more string functions, google 'string functions in python'

```

Out[19]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#9 Lists in python

<https://youtu.be/neTsPE9XFfsQ> (<https://youtu.be/neTsPE9XFfsQ>)

```

In [20]: list1 = ['a', 'b', 'c', 'd', 'e']
print (len(list1), list1[3])
list1.append('f')
list1.remove('f')
list1.insert(0,0)
print (list1)
nums=[3,9,9,6,18,9,3,3,7,6,81,3,6,8]
print (nums)
nums.sort()
print (nums)
nums.reverse()
print (nums)
print (nums [1:6], nums [:19], '\n', nums [::3], nums [::-3], nums [1:9:2], nums [9:1:-2])
# sort and reverse functions change the original list but SLICING keeps the list unchanged.
nums.append(6) # add element at the end
nums.insert(7,'random numbers') # the 1st term denotes the position of list and the 2nd term contains the element
print (nums)
nums.remove(6)
print (nums) # if the element we want to remove is present at multiple positions of the list, the element is removed from the first position
nums.pop() # remove 1 element from the end
print (nums)
nums[0] = 5.0 # to change the elements
nums[3] = 2.0
print (nums)
# mutable - can change, immutable - cannot change. List is mutable but Tuple is immutable.
tp1 = (61,5,76,8) # tuples. different from lists. parenthesis
print (tp1)
tp2= (6)
print (tp2) # for 1 element in tuple, only the element is printed without brackets. (unlike list)
a = 5
b = 6
t = a
a = b
b = t # to interchange the values of a and b
print (a,b)
c = 7
d = 4
c,d = d,c # to interchange the values of c and d
print (c,d)
print ('google python list functions and explore')
exit

```

```

5 d
[0, 'a', 'b', 'c', 'd', 'e']
[3, 9, 9, 6, 18, 9, 3, 3, 7, 6, 81, 3, 6, 8]
[3, 3, 3, 3, 6, 6, 6, 7, 8, 9, 9, 9, 18, 81]
[81, 18, 9, 9, 9, 8, 7, 6, 6, 6, 3, 3, 3, 3]
[18, 9, 9, 9, 8] [81, 18, 9, 9, 9, 8, 7, 6, 6, 6, 3, 3, 3, 3]
[81, 9, 7, 6, 3] [3, 3, 6, 9, 18] [18, 9, 8, 6] [6, 6, 8, 9]
[81, 18, 9, 9, 9, 8, 7, 'random numbers', 6, 6, 6, 3, 3, 3, 3, 6]
[81, 18, 9, 9, 9, 8, 7, 'random numbers', 6, 6, 3, 3, 3, 3, 6]
[81, 18, 9, 9, 9, 8, 7, 'random numbers', 6, 6, 3, 3, 3, 3]
[5.0, 18, 9, 2.0, 9, 8, 7, 'random numbers', 6, 6, 3, 3, 3, 3]
(61, 5, 76, 8)
6
6 5
4 7
google python list functions and explore

```

Out[20]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#10 Dictionary in python

<https://youtu.be/5y1sh0-oKTE> (<https://youtu.be/5y1sh0-oKTE>)

```
In [21]: # Dictionary is nothing but key value pairs
d1 = {} # type - dictionary. immutable
print (type(d1))
d2 = {"PE effect" : "Einstein", "Compton effect" : "Compton", "Black body radiation" : "Planck", "Quantum mechanics" : "De Broglie"}
print (d2['PE effect'], d2['Quantum mechanics'], d2['Quantum mechanics']['2nd'])
d2['Gravity'] = 'Newton'
d2[12] = 'SKP'
print (d2)
del d2[12]
print (d2)
d3 = d2
del d3 ['Quantum mechanics']
print (d2) # d3=d2 doesn't make a copy. If we delete an element from d3, that element will also be deleted
print (d2.get ('Quantum mechanics')) # to get a deleted element.
d2.update ({'Laws of motion' : 'Newton'}) # to update a new element
print (d2)
print (d2.keys(), '\n', d2.items())
print ('google dictionary functions and explore.')
exit
```

```
<class 'dict'>
Einstein {'1st': 'Planck', '2nd': 'De Broglie', '3rd': 'Schrödinger', '4th': 'Heisenberg'} De Broglie
{'PE effect': 'Einstein', 'Compton effect': 'Compton', 'Black body radiation': 'Planck', 'Quantum mechanics': {'1st': 'Planck', '2nd': 'De Broglie', '3rd': 'Schrödinger', '4th': 'Heisenberg'}, 'Gravity': 'Newton', 12: 'SKP'}
{'PE effect': 'Einstein', 'Compton effect': 'Compton', 'Black body radiation': 'Planck', 'Quantum mechanics': {'1st': 'Planck', '2nd': 'De Broglie', '3rd': 'Schrödinger', '4th': 'Heisenberg'}, 'Gravity': 'Newton'}
{'PE effect': 'Einstein', 'Compton effect': 'Compton', 'Black body radiation': 'Planck', 'Gravity': 'Newton'}
None
{'PE effect': 'Einstein', 'Compton effect': 'Compton', 'Black body radiation': 'Planck', 'Gravity': 'Newton', 'Laws of motion': 'Newton'}
dict_keys(['PE effect', 'Compton effect', 'Black body radiation', 'Gravity', 'Laws of motion'])
dict_items([('PE effect', 'Einstein'), ('Compton effect', 'Compton'), ('Black body radiation', 'Planck'), ('Gravity', 'Newton'), ('Laws of motion', 'Newton')])
google dictionary functions and explore.
```

Out[21]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#11 Exercise 1 - Apni Oxford Dictionary

https://youtu.be/y_aRifWZbnY (https://youtu.be/y_aRifWZbnY)

```
In [22]: print ('Create a dictionary. Include 4-5 words.')
```

Create a dictionary. Include 4-5 words.

#12 Sets in Python

<https://youtu.be/iVJv3zdgkD4> (<https://youtu.be/iVJv3zdgkD4>)

```
In [23]: s0= set ()      # empty set
s1 = set ()
s2= set ([5,8,9,7,5,3])
print (s0, s2, type (s1), len(s1), len(s0), min (s2)) # one element is used one time only
s1.add (1)
s1.add (1)
s1.add (2)
print (s1) # one element is added one time only
s3 = s1.union ({5,9,'abc'})
s4 = s1.union (s2)
print (s3,s4)
s5= s4. intersection ({1,2,3,4,5,6})
print (s5)
print (s4.isdisjoint (s5), s0.isdisjoint (s4)) # disjoint test
s5.remove (2)
print (s5)
exit
```

```
set() {3, 5, 7, 8, 9} <class 'set'> 0 0 3
{1, 2}
{1, 2, 5, 9, 'abc'} {1, 2, 3, 5, 7, 8, 9}
{1, 2, 3, 5}
False True
{1, 3, 5}
```

```
Out[23]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>
```

#13 If Else & Elif Conditionals in python

<https://youtu.be/3VejlihDfwU> (<https://youtu.be/3VejlihDfwU>)

```
var1 = 4
var2 = 12
var3 = int(input('enter your number: ')) # string to integer
if var3 > var2:
    print('greater than 12')
elif var3 == var2: # if used - statement is checked always. elif used - statement is checked only when
previous statement is false.
    print('equal to 12')
else:
    print('lesser than 12')
exit

list1 = [55, 71, 6, 78, 64, 41]
print('\n', 71 not in list1)
if 71 not in list1:
    print('not in list')
else:
    print('in list')

exit
```

#14 Exercise 1 Solutions

https://youtu.be/6_hq_8FsKkE (https://youtu.be/6_hq_8FsKkE)

```
# Create a dictionary and take input from the user and return the meaning of the
# word from the dictionary
Dict = {"ignore":"refuse to take notice of or acknowledge", "abandon":"cease to support or look after",
        "exaggerate":"enlarged or altered beyond normal proportions", "prejudice":"preconceived opinion
that is not based on reason or actual experience", "programming":"the process of writing computer
programs"}
print("Enter the Word")
Data1 = input()
print(Data1, "means", Dict[Data1])
exit
```

#15 Exercise 2 - Faulty Calculator

<https://youtu.be/VP8s9NiFToM> (<https://youtu.be/VP8s9NiFToM>)

```
In [24]: print ('''Design a calculator which will correctly solve all the problems
except the following ones :
        45*3 = 555, 56+9 = 77 and 56/6 = 4
your program should take operator
and the two numbers as input from the user and return the result.''' )
exit
```

Design a calculator which will correctly solve all the problems
except the following ones :
45*3 = 555, 56+9 = 77 and 56/6 = 4
your program should take operator
and the two numbers as input from the user and return the result.

Out[24]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#16 for loops in python

<https://youtu.be/sSyCRQx5WM0> (<https://youtu.be/sSyCRQx5WM0>)

```
In [25]: list1 = [[26,8],9,(6,7),[44,7],54]
for item in list1 :
    print (item) # ordinary print = 1 line. for Loop and print = one element in one line.
list2 = [('Argentina','Messi'), ('Brazil', 'Neymar'), ('Portugal', 'Ronaldo')]
dict1 = dict (list2)
print (dict1)
for country, captain in dict1.items () :
    print ('\t', country, 'is led by', captain) # tab is taken in each line
for country, captain in dict1.items () : # function dict1.items () used here
    print ('\n', country, 'is led by', captain) # extra line is taken in each line
for item in dict1 :
    print ('\t', item)
list3 = ['SKP', 61,81,6,88,5,7,8,65,8,10,6]
for item in list3 :
    if str(item).isnumeric () and item>=10 :
        print ('item greater than 10 in list3 is', item) # function str(item).isnumeric ()
exit
```

```
[26, 8]
9
(6, 7)
[44, 7]
54
{'Argentina': 'Messi', 'Brazil': 'Neymar', 'Portugal': 'Ronaldo'}
    Argentina is led by Messi
    Brazil is led by Neymar
    Portugal is led by Ronaldo

Argentina is led by Messi

Brazil is led by Neymar

Portugal is led by Ronaldo
    Argentina
    Brazil
    Portugal
item greater than 10 in list3 is 61
item greater than 10 in list3 is 81
item greater than 10 in list3 is 88
item greater than 10 in list3 is 65
item greater than 10 in list3 is 10
```

Out[25]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#17 while loops in python

<https://youtu.be/qyWuPZNOeGg> (<https://youtu.be/qyWuPZNOeGg>)

```
In [26]: i=0
while (i<5) :
    print (i)
    i = i + 1.5
exit
```

```
0
1.5
3.0
4.5
```

Out[26]: <IPython.core.autocall.ZMQExitAutocall at 0x2683b828bb0>

#18 break & continue statements

<https://youtu.be/aXGC1fx6QQo> (<https://youtu.be/aXGC1fx6QQo>)

```
i = 0
while (i<30) :
    i = i+2
    print (i+1, end=' ')
i=0
while (True) : # while (True) is a loop that continues to infinity until we break it
    print (i+4)
    i= i+11
    if (i>100) :
        break # stop the loop
i=0
while (True) :
    if i+1<5 :
        i=i+1
        continue
    print (i+1, end = ' ')
    if (i==50) :
        break
    i= i+1
# making a code in which we want to check if the input is greater than 100 or not
while (True) :
    inp = float(input (" \n\nEnter a number \n"))
    if inp>100 :
        print ('Yeah ! the entered number is greater than 100')
        break
    else :
        print ('try again')
        continue
exit
```

#19 Exercise 2 - Faulty Calculator Solution

<https://youtu.be/s0lxxTHWx6w> (<https://youtu.be/s0lxxTHWx6w>)

```
# 45 * 3 = 555, 56+9 = 77, 56/6 = 4
# Design a calculator which will correctly solve all the problems except the following ones:
# 45 * 3 = 555, 56+9 = 77, 56/6 = 4
# Your program should take operator and the two numbers as input from the user
# and then return the result
print("Enter 1st Number")
num1 = float(input())
print('Enter 2nd Number')
num2 = float(input())
print('so What you Want?'+'+,-,/,%,*')
num3 =input()
if num1 ==45 and num2==3 and num3=='*':
    print("555")
elif num1 == 56 and num2 == 9 and num3 == '+':
    print("77")
elif num1 == 56 and num2 == 6 and num3 == '/':
    print("4")
elif num3=='*' :
    num4=num1*num2
    print(num4)
elif num3 == '+':
    plus=num1+num2
    print(plus)
elif num3 == '/':
    Dev=num1/num2
```

```
    print(Dev)
elif num3 == '-':
    Dev=num1-num2
    print(Dev)
elif num3 == '%':
    percent=num2%num1
    print(percent)
else:
    print("Error! Please check your input")
exit
```

#20 Exercise 3 - Guess the Number

<https://youtu.be/2loUi5KxZ3E> (<https://youtu.be/2loUi5KxZ3E>)

```
In [27]: print(
'''
total number of guesses = 9
print number of guesses left
number of guesses took to finish / game over
'''
)
```

```
total number of guesses = 9
print number of guesses left
number of guesses took to finish / game over
```

In []: