

```
In [1]: #Gives you the ascii value
```

```
print(ord('A'))
```

```
65
```

```
In [2]: #Gives you the character at the ascii value  
print(chr(65))
```

```
A
```

```
In [3]: #Get octal value  
print("octal value if 12 :",oct(12))  
#Get Hex.Value  
print("hex value if 106 :",hex(106))
```

```
octal value if 12 : 0o14  
hex value if 106 : 0x6a
```

```
In [4]: #Get the binary value  
print("Binary of 10",bin(10))
```

```
Binary of 10 0b1010
```

```
In [8]: #Tuples  
'''=> This is also a datatype in python  
This is represented by small bracket / ( ) pair of parenthesis  
{ } is curly braces  
< > is angular bracket  
[ ] is the square bracket
```

```
we use it to make the immutable homogeneous and heterogeneous lists
```

```
'''
```

```
t=(1,2,3)  
print("Type of t is :",type(t))
```

```
print("What if we print the t tuple ",t)
```

```
print("at the 1st index : ",t[0])  
# t[0]=4 : will give error : TypeError: 'tuple' object does not support item assignment
```

```
Type of t is : <class 'tuple'>  
What if we print the t tuple (1, 2, 3)  
at the 1st index : 1
```

```
In [9]: #min , max , len , sorted , sum , all will work in it  
# also the concatenation on the Tuples will lead to creation of a new Tuple
```

```
a=(1,2)  
b=(3,4)  
c=a+b  
print("Resulting tuple c is :",c)
```

```
Resulting tuple c is : (1, 2, 3, 4)
```

```
In [11]: #repeat the elements  
print("repeat the a tuple 2 times :",a*2)
```

```
repeat the a tuple 2 times : (1, 2, 1, 2)
```

```
In [ ]: #slicing is same as that of lists (task to test)  
  
t=(1,2,3,4,5)  
print("print from index 0 to 2 ",t[0:3])
```

```
print from index 0 to 2 (1, 2, 3)
```

```
In [14]: # now consider  
  
t=(1,2,3)  
print("currently tuple is : ",t)  
# to change 1 with 21  
#step 1  
l = list(t) # tuple became mutable  
l[0] = 21  
#step 2  
t = tuple(l)  
print("New changes : ",t)
```

```
currently tuple is : (1, 2, 3)  
New changes : (21, 2, 3)
```

```
In [15]: #tuple has only count and index  
  
t=() # is an empty tuple  
# we want to store it into the tuple  
#1st we have to convert the t into list  
#list has the append function to add the elements  
#Then convert it back into tuple  
print("current t : ",t)  
l=list(t)  
l.append(10)  
l.append(20)  
t=tuple(l)  
print("new changes is : ",t)
```

```
current t : ()  
new changes is : (10, 20)
```

```
In [17]: # count()  
print("Count of element 1 in t is : ",t.count(1))  
  
#count() gives you total number of occurrences / repetitions  
t=(1,2,1,3,4,1)  
print("1 has occurred : ",t.count(1)," times")
```

```
Count of element 1 in t is : 0  
1 has occurred : 3 times
```

```
In [18]: #index() in the python  
# it will return index of the given item  
  
print("Index of 3 is : ",t.index(3))
```

```
Index of 3 is : 3
```

```
In [24]: #Dictionary in Python
#Dictionary is composed with words and its meaning.
#This is the sense of the key(word) and value(meaning)
#By the composition of above two a dictionary is made

#one dictionary has 1 element that has one key and value.
#This is represented by a curly bracket

#This is a changable / mutable.

#Syntax : keep the key in the single quotes

# Obj = {key1:keyValue,
#         key2:keyValue,
#         }

d={'Roll':52,'Name':'Ravi'}
print("The dictionary d is :",d)
#There is no concept of index in the dictionary
#No dictionary represents its elements from and index but from its keys

#Since, no index therefore no slicing

print("To access Roll from the dictionary : d['Roll'] =",d['Roll'])
print("To access Name from the dictionary : d['Name'] =",d['Name'])
#key must match , key name is case sensitive , keys must be taken care of.

# to change the name then
d['Name']='Soni'
print("New Name after d['Name']= 'Soni' : d['Name'] =",d['Name'])
```

```
The dictionary d is : {'Roll': 52, 'Name': 'Ravi'}
To access Roll from the dictionary : d['Roll'] = 52
To access Name from the dictionary : d['Name'] = Ravi
New Name after d['Name']= 'Soni' : d['Name'] = Soni
```

```
In [29]: #item and items are related to the dictionary
#This is help us to convert the whole dictionary into the list
#This will convert each item as a tuple and entire dictionary as a list

z=d.items()
print("Converted Dictionary d into list z as :\n",z)
```

```
Converted Dictionary d into list z as :
dict_items([('Roll', 52), ('Name', 'Soni'))]
```

```
In [32]: #print all the keys of the dictionary
print(d.keys()) #it will return the keys in the form of a list.

print("The list form of the values in dictionary d is :\n",d.values())
```

```
dict_keys(['Roll', 'Name'])
The list form of the values in dictionary d is :
dict_values([52, 'Soni'])
```