Data Science & AI
&
NIC - Param

Python-For Data Science

Language Fundamentals

Lecture No.- 03

By- Pankaj Sharma Sir

Recap of Previous Lecture

Topic Language Fundamentals - 02

Topics to be Covered

Topic Language Fundamentals - 03

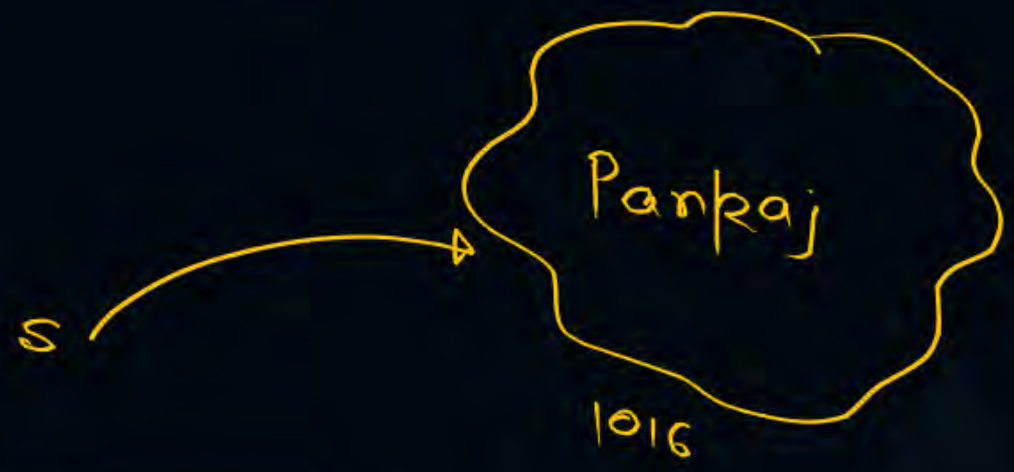
Topic : Language Fundamentals

Phd → 10 year

immutability :

Object

s = "Pankaj"



s[1] = 'u'

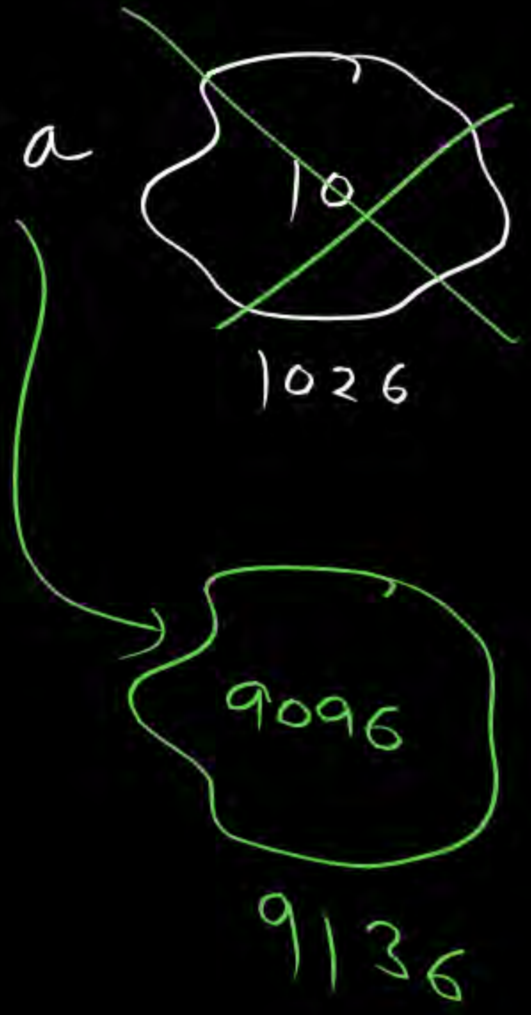
we want to change

p	a	n	k	a	j
0	1	2	3	4	5

$a = 10$

$=$

$a = 9096$



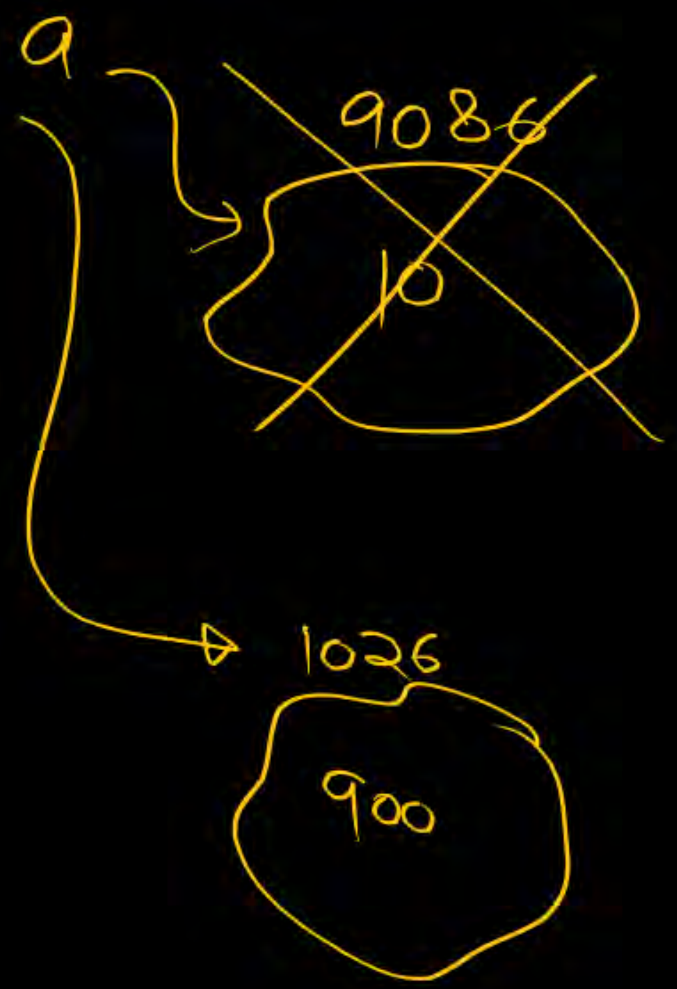
id( ) ⇒

Once an object is created , you can not update it

immutable

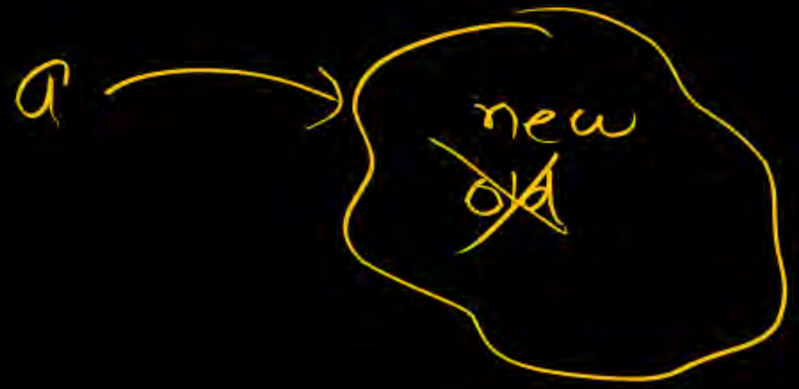
a = 10

a = 900



mutable

a = new



List

Group of elements/values

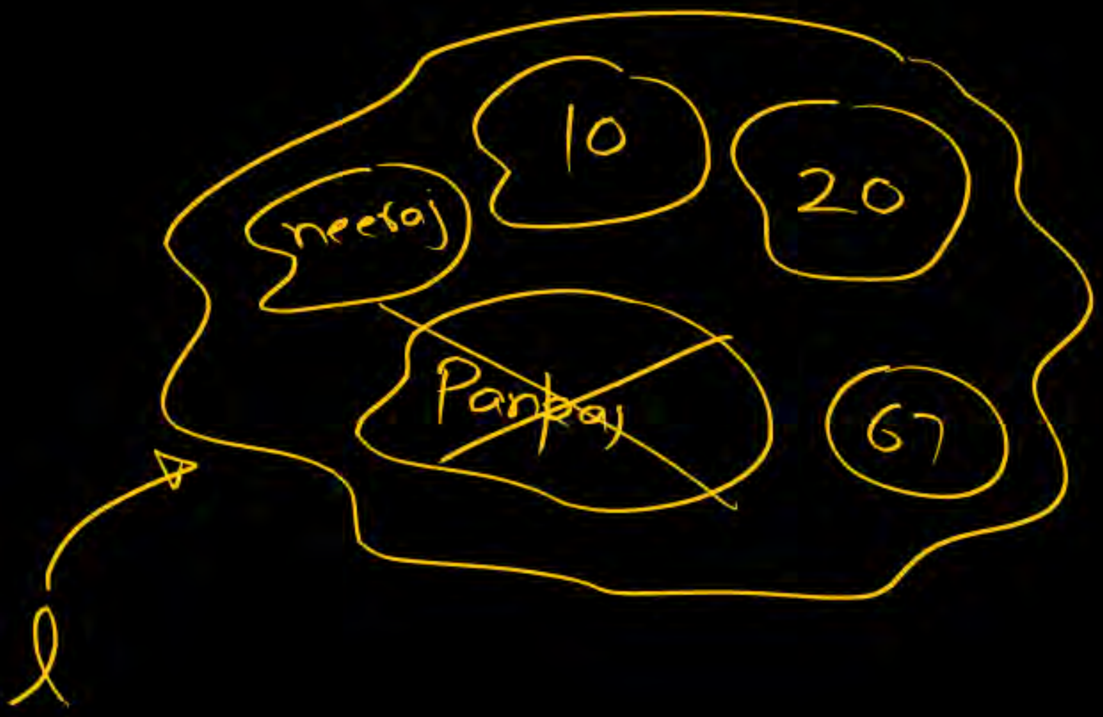
Order preserve (index)

Duplicate elements allowed

Heterogen. elements

Tuple

Same as list but immutable



l = [10, 20, "Pankaj", 67]

l[2] = "Sharma"

Tuple → indexing ✓

→ slicing

t = (10, 20, 30, 10)

range : a sequence of numbers

5 to 9   group of number

r = range(5, 10)

5, 6, 7, 8, 9

1st element in r ?

r[0] ⇒ 5

set

No duplicate

Order not preserve

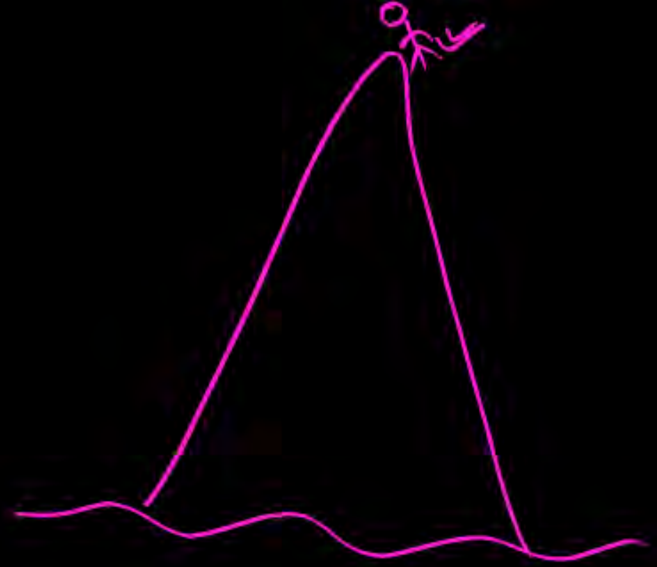
Heterogenous elements.

No concept of index.

No slicing

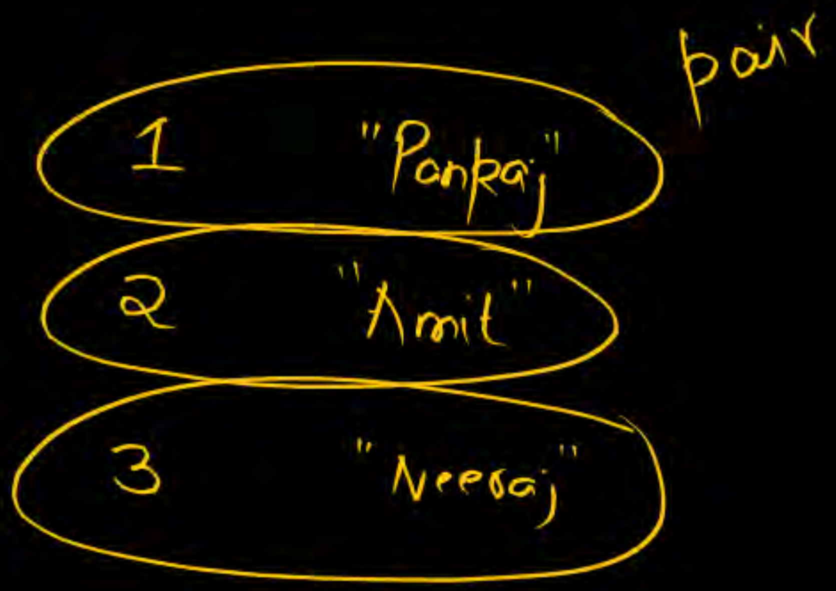
Mutable

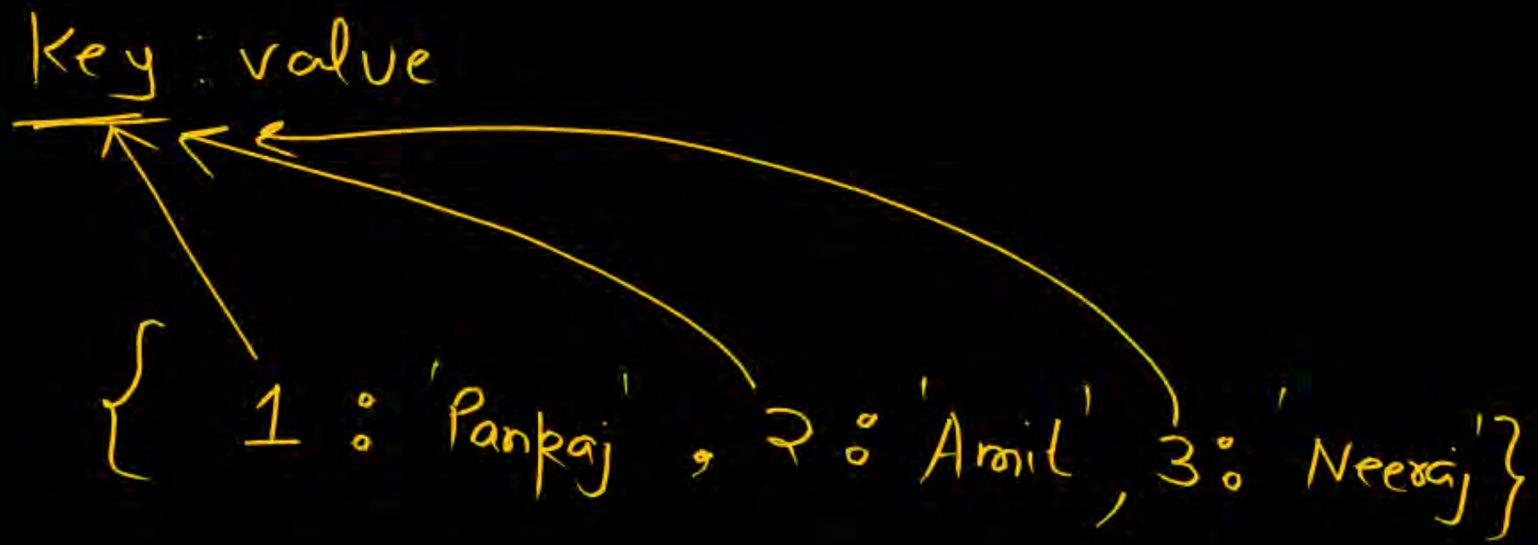
Growable (Add/remove)



{1, 2}, {2, 1} are same set.

dictionary





key : value

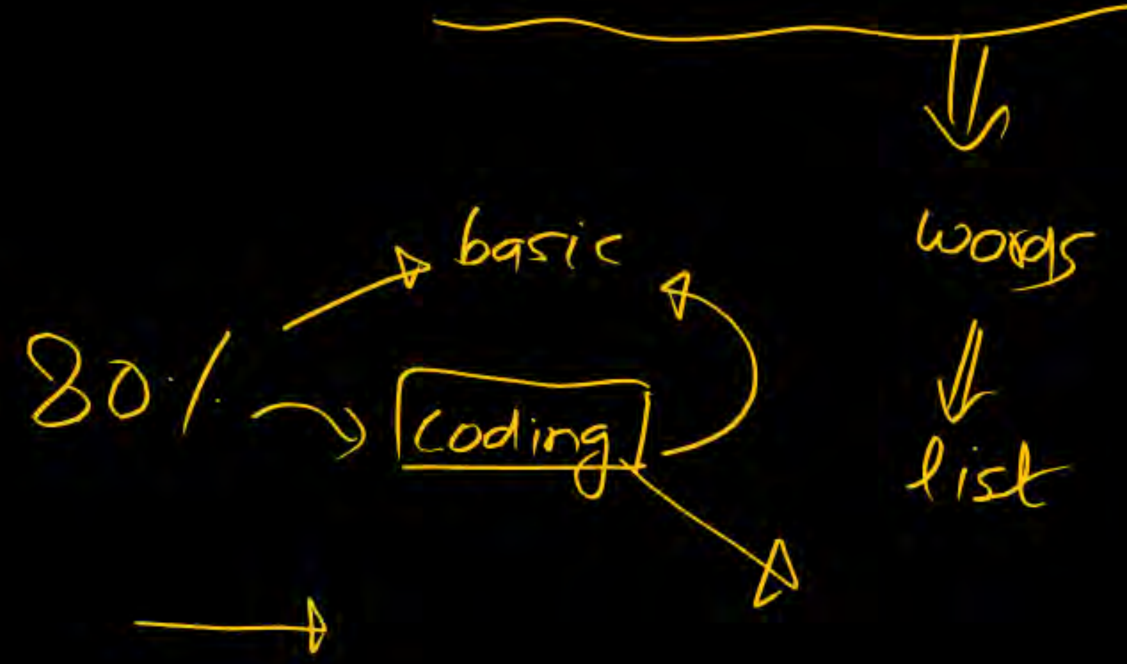
{ 1 : 'Pankaj' , 2 : 'Amit' , 3 : 'Neeraj' }

Duplicate keys not allowed

values can be duplicate

{ 1 : 'Amit' , 2 : 'Amit' } ✓

S = "Pankaj sharma is bad faculty"



Operators in Python

day 3 Python

In [1]:
```
s="pankaj"
```

In [2]:
```
s[1]='u'
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[2], line 1
----> 1 s[1]='u'

TypeError: 'str' object does not support item assignment
```

In [3]:
```
a=10
id(a)
```

Out[3]:
```
140720527873096
```

In [4]:
```
a=9897
id(a)
```

Out[4]:
```
1425201201552
```

In [5]:
```
l=[1,23.4,"pankaj",67]
```

In [6]:
```
l[0]
```

Out[6]:
```
1
```

In [7]:
```
l[1]
```

Out[7]:
```
23.4
```

In [8]:
```
l[2]
```

Out[8]:
```
'pankaj'
```

In [9]:
```
l[-1]
```

Out[9]:
```
67
```

In [10]:
```
l[1:4]  #slicing 1,2,3==>element 2nd,3rd,4th
```

Out[10]:
```
[23.4, 'pankaj', 67]
```

In [11]:
```
l[0]=1001
```

In [12]:
```
l
```

Out[12]:
```
[1001, 23.4, 'pankaj', 67]
```

```
In [13]: #append()===>add an element in the end of list
         l.append("sharma")
```

```
In [14]: l

Out[14]: [1001, 23.4, 'pankaj', 67, 'sharma']
```

```
In [15]: l.append(1001)
```

```
In [16]: l

Out[16]: [1001, 23.4, 'pankaj', 67, 'sharma', 1001]
```

```
In [17]: l.remove(23.4)
```

```
In [18]: l

Out[18]: [1001, 'pankaj', 67, 'sharma', 1001]
```

```
In [21]: l.remove(1001)#only remove first occurence of that element
```

```
In [20]: l

Out[20]: ['pankaj', 67, 'sharma', 1001]
```

```
In [22]: l

Out[22]: ['pankaj', 67, 'sharma']
```

```
In [23]: l.append(67)
```

```
In [24]: l

Out[24]: ['pankaj', 67, 'sharma', 67]
```

```
In [25]: l.remove(67)
```

```
In [26]: l

Out[26]: ['pankaj', 'sharma', 67]
```

```
In [27]: t=(10,12.3,"pankaj",67)
```

```
In [28]: type(t)

Out[28]: tuple
```

```
In [30]: t[0]=1009 #immutable
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[30], line 1
----> 1 t[0]=1009

TypeError: 'tuple' object does not support item assignment
```

```
In [31]: t.append("Sharma")#ud ke laat pdegi
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[31], line 1
----> 1 t.append("Sharma")

AttributeError: 'tuple' object has no attribute 'append'
```

```
In [32]: t.remove(67)#ud ke laat ==>tiger shroff bhaia
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[32], line 1
----> 1 t.remove(67)

AttributeError: 'tuple' object has no attribute 'remove'
```

```
In [33]: #tuple is read only
```

```
In [34]: t=(10,20,10,20,10)#duplicate allowed
```

```
In [35]: t
```

```
Out[35]: (10, 20, 10, 20, 10)
```

```
In [37]: t[0]#index ,order preserve just like list
```

```
Out[37]: 10
```

```
In [38]: t[0]=100
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[38], line 1
----> 1 t[0]=100

TypeError: 'tuple' object does not support item assignment
```

```
In [39]: l=[1,2,"pankaj",34]
```

```
In [40]: l[2]="neeraj"
```

```
In [41]: l
```

```
Out[41]: [1, 2, 'neeraj', 34]
```

```
In [42]: l[2][0]='d'
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[42], line 1
----> 1 l[2][0]='d'

TypeError: 'str' object does not support item assignment
```

In [43]:
```
t=(10,20,30,40)
```

In [44]:
```
t[1:4]#index1,2,3
```

Out[44]:
```
(20, 30, 40)
```

In [45]:
```
#range  :represent a sequence of numbers
```

In [46]:
```
range(5)#it will generate numbers starting from 0 to 5-1
```

Out[46]:
```
range(0, 5)
```

In [47]:
```
for element in range(5):print(element)#loop
```

```
0
1
2
3
4
```

In [48]:
```
#format 2
#range(start,end)
range(5,10)# sequence of numbers from 5 to 9
```

Out[48]:
```
range(5, 10)
```

In [49]:
```
for element in range(5,10):print(element)
```

```
5
6
7
8
9
```

In [50]:
```
#format 3
#range(start,end,step)
# range(2,20,2) ===>2,4,6,8,10,12,14,16,18  2 to 19 with a step size of 2
```

In [51]:
```
for i in range(2,20,2):print(i)
```

```
2
4
6
8
10
12
14
16
18
```

```
Out[80]: {1, 2, 3, 4, 'pankaj', 'sharma'}
```

```
In [81]: a=set("pankaj")#"pankaj" is a group of 6 elements in which 'a' is duplicated
```

```
In [82]: a
```

```
Out[82]: {'a', 'j', 'k', 'n', 'p'}
```

```
In [83]: a={1,2,3,4}
         a.remove(5)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[83], line 2
      1 a={1,2,3,4}
----> 2 a.remove(5)

KeyError: 5
```

```
In [84]: #frozenset ===>same as set but it is immutable
         #no add()
         #no remove
         s={1,2,3,4}
         x=frozenset(s)
```

```
In [85]: x
```

```
Out[85]: frozenset({1, 2, 3, 4})
```

```
In [86]: type(x)
```

```
Out[86]: frozenset
```

```
In [87]: x.add(12)#error
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[87], line 1
----> 1 x.add(12)

AttributeError: 'frozenset' object has no attribute 'add'
```

```
In [88]: x.remove(1)#error
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[88], line 1
----> 1 x.remove(1)

AttributeError: 'frozenset' object has no attribute 'remove'
```

```
In [89]: d={1:'amit',2:'rahul',3:'pankaj'}
```

```
In [90]: d
```

```
Out[90]: {1: 'amit', 2: 'rahul', 3: 'pankaj'}
```

```
In [91]: d[1]='arun' #we are trying ==>{1:'arun'}
```

```
In [92]: d
```

```
Out[92]: {1: 'arun', 2: 'rahul', 3: 'pankaj'}
```

```
In [93]: #if we try to add/insert an entry with existing key then old value will be replaced
         #with new value
```

```
In [94]: d={}
```

```
In [95]: type(d)
```

```
Out[95]: dict
```

```
In [96]: d[5]='pavitra'
```

```
In [97]: d
```

```
Out[97]: {5: 'pavitra'}
```

```
In [98]: d[6]='neeraj'
```

```
In [99]: d
```

```
Out[99]: {5: 'pavitra', 6: 'neeraj'}
```

```
In [100… #mutable but order is not preserved
```

```
In [101… #None type===>functions
```

```
In [103… s=set()
```

```
In [104… s
```

```
Out[104]: set()
```

```
In [105… print(s)
```

```
set()
```

```
In [106… s=(1,)
```

```
In [107… s
```

```
Out[107]: (1,)
```

```
In [108… type(s)
```

```
Out[108]: tuple
```

In [ ]:

THANK - YOU