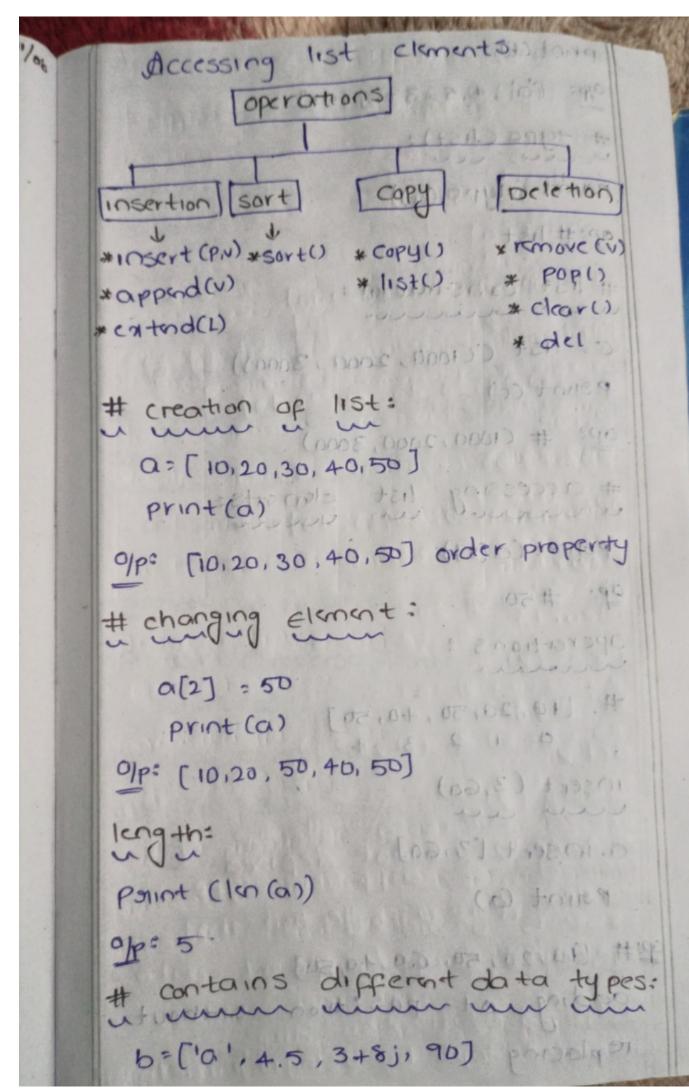
List: A list is a collection that "/" contains elements of different/ same data type > in list multiple values are Assigned to single variable name > we create a list by using? [] square Brackets notation (nothing but symbol) syntan: sldblum konstre fell List - Name > [Vi, V2 ... Vn] que mall. [10,20,30,40] list concepts: -> create 108.08.00 bombio - ordered DE DE 190 - mutable locate of bersking -> Allow dups 01.08 DE 190 Is Tog the oc. 01/ 18 of soll que woll by alist contained different data type (sleepports) storbion -> type() wasto inche Darson - 11st Constructor



```
Print (b) Parada part post sooth
  O/P: [a', 4.5, 3+8) 190]
  # type clist):
  Paint (type (a))
  P:# list Oppor Other Contine
  # list constructor:
  C= fist ((1000, 2000, 3000))
  Panot (c) - For act mothers th
  op: # (1000, 2000, 3000)
      0 - (10,20,30) +0,501 ,08(618
  # accessing list elements:
  point (a[2]) (14.04 08 (000) 1910
  op: #50
operations:
  # [10,20,50,40,50] (0) +0199
                0(21):50
             [10,20,50,40,50]
  Insert (3,60)
  a.10ser+ [3,60]
             13 (0 1 (0 to) (0 to) ) traine?
  Point (a)
              The feel of the state of the
92# [10,20,50, 60,+0,50]
  adding at position without
 replacing TOP WELE PARTO DES
```

```
appoid (70)
           ( DED C. PODE DOOR) H-
a. append (70) (0000 10000 10001) 11
Ponot (a)
Op# [10,20,50,60,40,59,70]
 adding end of the list
Extend > Li+L2 lombing:
 a.ea tend (b)
  paint (a)
# (10,20,50,60,40,50,70), (0',4.5,(3+4))
sorting Ascending order: (E) Holey
  a. sort () (0+ ,00,00,00,00,01) 1990
 point (a)
 OP: # [10,20,40,50,50,60,70]
 a. sort ( reverse = True) ( ) torks
  point (a)
OP: [70, 60,50, 50, 40, 20, 10]
X=[10)3.4,7+8j]
   x. sort () 6 type
   Paint (xi) solles of the soly
copy: x = c.copy ()
      Paint (x)
Existe tople by of stops and on
```

op-# [1000,2000,3000] # [1000, 2000, 3000] deletion: (a) tomes a. remove (50) + 6+ 60 , 50 (01) #4 Parint (a) of all go tops por OP: [70,60,50,40,20,10] (d) brot 45.0 Pop: 7 = [10,20,30,40,50,30,40] Z. POP Pain+ (2) 19619 (2000) Op: [10,20,30,50,30,40] clears (D) (D) foney OP # (10,20,40,00,50,60,40) clear() 0. 301 F (18/050 = TICO) (E) trile9 OP: #[] (a) 4 meg del 2 10/100 10+ 100 100 100 100 190 bailot (5) 6 [19++ + 18 (01) A # name '2' is not depined? ruple: it is a collection that contains some / different data type (x) min we create tuple by using 1) paranthesis

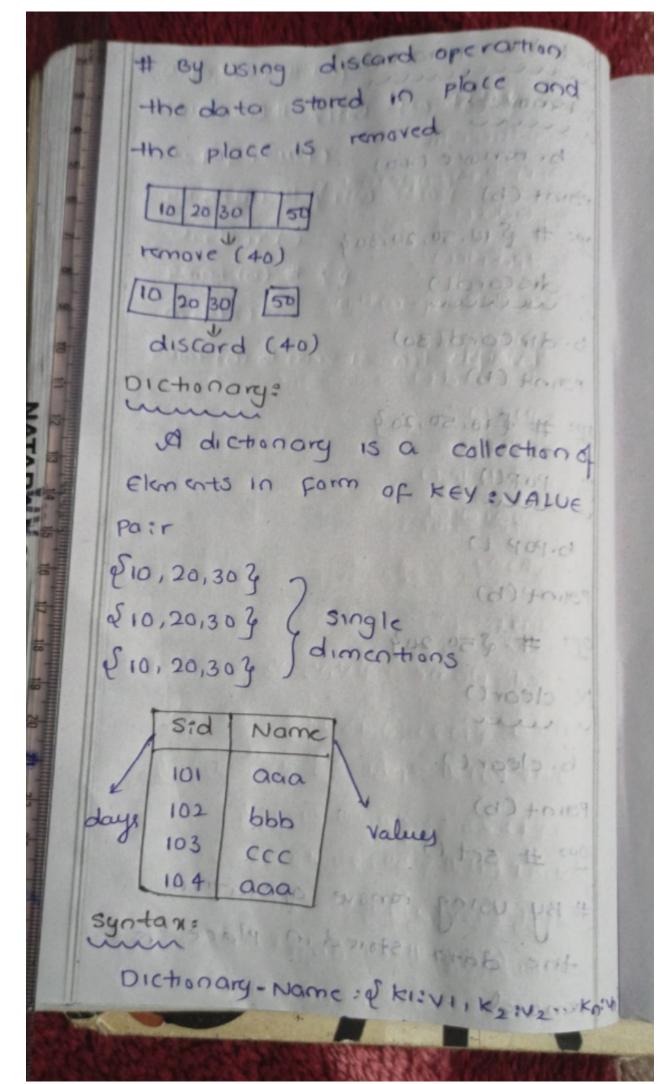
```
Properties: We have the total the
 > ordered
 >1mmutable
-> Allowed duplicantes
 a [10,20,30,40,50]
  Pain+(a)
 OP: # (10,20,30,40,50)
 Point (a[3])
cpitt 50
 0[3] = 60
 Polint(a)
 op: type error: (be as on # 19)
+ tuple object does not support item
 assignment
 b= list[a]
 Parint (6) (05,05,61) to
 OP: [10,20,30,59 40,50]
 b[3] = 60 1 1 1 1 1 1
 perint (b) booking
 op: [10, 20, 30, 60, 40, 507
 a= tuple(6) 16 months (1) the self the
 Polint (a)
```

op: # (10,20,30,60,40,50) Packed tuple : horobio of oldet unimit a=[10,20,30] (102030) unpacked tuple: (05.0+,00,00,00) 0 (x,y, Z) (10) (20) (30) X = (10, 20, 30)(P,9,0)=X=(E,P,9) Ponnt (x) OP: # (10,20,30) ponot (9) # 20,000 b 1000 01910+ 2=80 OSSER BONNESSED point (9) # 80 P9110 + (X) # (10,20,30) set: A set is a collection of elements of different type & d > it is un ordered (el) Housey - It is immutable 15 180 - it doesn't allow dups significa Parish (C) This is a

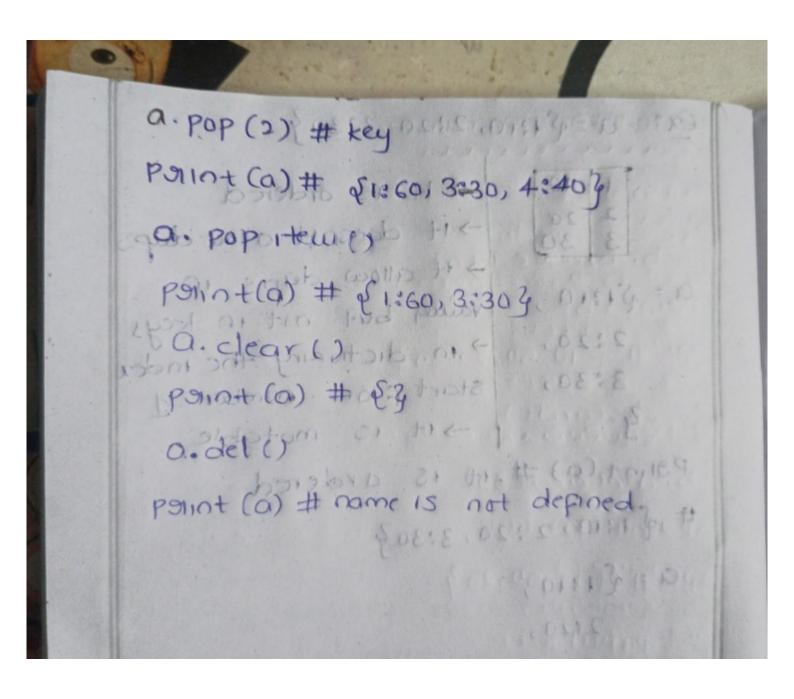
```
operations: Do out the so.
  screation sols tooldo took took
  -> constructor tomogrado was.
 sinsertion operation
   * add()
   * update()
 > beletion operations.
   * remove ()
  * dis Card ()
   * POP ()
   * clear()
   * deletel) (0+) bbo.d
    notonics to the the (a) tores
                 (00) bbs
 creation:
Syntan: 08:05:05:00 $ # (d) to
   Set Name = &V, V2, V3...
Ex: a: $10,20,80,40,503+ (10) +11119
Polint (a) # {50, 20, 40, 10, 303
a = $10,20,30,40,50,30,203
porint (a) # $50, 20, 40, 10,303
Point (In (a)) #5
point (0[2]) # set abject 18 not
su bscriptable
```

```
0[2] :60 # NO 0000033 11 184790
    a(a) # set object does not suppor
                                                                       constructory,
     item assignment
     b = set((10, 20,30))
    Point (b) # (10,20,303)
    insertion operation:
  is add()
b= 5c+ ((10, 20,30))
  Palint (b)
  b. add (40)
  Parat (6) # $40, 10, 20,30}
  b. add (50)
  Paint (b) # &40,10,50,20,303
ii, update ()
a update (b)
point (a) # & 40,50, 10,20,303
                                     $ 30,00 00 18 18 18 10 80,00 6 0
    a. union (b)
                      $ 05 Ch 0+,05,00 9 th (0) 401109
  pain+ (a) (co) 101) third
  op: # $40150,10,20,30 } (10)
       THE PROPERTY OF A PROPERTY OF
```

De letion operation: 10 months po par bereten wetabions 3, temovec) is proposed the sold out b. remove (40) Polint (b) Op: # \$ 10,50,20,303 in, discordi) b. dircard (30) (04) Point (b) OP: # \$10,50,203 to consolid is a collection of POPC) A TO MADE OF ETTO ONE 6. POP () Polint (b) op: # \$50, 203 ?v, clear() STADIN bid b. clear() Pann+(b) ope # set () wer ded (0) 105 | 666 | # By Using remove operation the data stored in place is remared invite by omour-proportion



```
Ex: a= $1:10, 2:20, 3:30 }
      10 1 14 113 ordered
           -> It doesn't allow dups
      30) > it allow dups in
a = $1:10, Values but not in keys
   2:20, > in dictionary the index
    3:30, start from number 1
   3 -> It is mutable
Point(a) # it is ordered
# $ 1:10, 2:20, 3:303
a = 21:10,
     2:20,
     3:30
print(a)# it doesnot allow deps.
# 21:10, 2:20, 3.303
 Parint (a.key (1)
# 21,2,33
Ponnt (a. Values())
# dict-values (10,20,30)
 0[1]:60
Polint (a)
# (1: 60,2:30,3:30)
Ponnt (a [0]) # Key error
```



```
Tuple It is a Collection that Contains Same/Different
      Data Type.
      We Create Tuple By Using ( )
      Properties
      Oredred
      Immutable
      Allow Duplicates
```

```
a=(10,20,30,50,40,50)
print(a) "'TypeError:
#(10, 20, 30, 40, 50) 'tuple' object
#(10, 20, 30, 50, 40, 50) does not support
print(a[3])#50 item assignment'''
#a[3]=60
print(a)
```

```
#Extend -> L1+L2 Combine
                               #Descending Order
|b.extend(a)
                                a.sort(reverse=True)
print(b)
                                print(a)
#['a', 4.5, (3+8j), 90, 10, 20,
                               #[70, 60, 50, 50, 40, 20,
50, 60, 40, 50, 70]
                                10]
|#[10, 20, 50, 60, 40, 50, 70]
                               x=[10,3.4]
#Sorting
                                x.sort()
#Ascending Order
                                print(x) #[3.4, 10]
a.sort()
                                #copy()
```