3. Tuple: Tuple is a collection of homogenous & heterogenous datatype.

- The values of the tuple may or may not be enclosed inside the ().
- The values of the tuple should be seperated by ,
- Syntax: 1. var = (val1, val2, ...valn)
  t = (90.5, 30.6, 40.8)
  2. var = val1, val2.... valn
  d = False, 58+3j, 59.6, 'ab'.
  3. var = (value,) / var = val,
- Example: 1) t = (56) → type(t) → <class 'int'>
  2) t = (56,) → type(t) → <class 'tuple'>
  3) t = 'abc' → type(t) → <class 'str'>
- Tuple is an ordered datatype
- Tuple supports both +ve & -ve indexing.
- Tuple is immutable datatype.

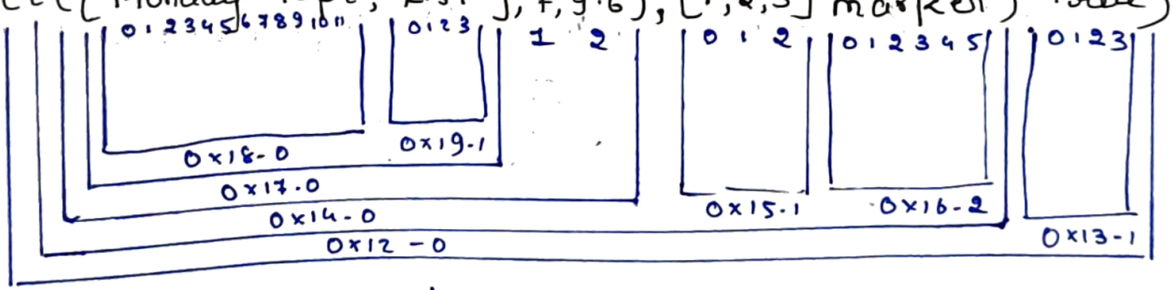
Example: t = (1,2,3)
t[0] = 10

TypeError: 'tuple' object does not support item assignment

* but → t = (1,2,3,[10,4])
t[3][0] = 5
t = (1,2,3,[5,4])

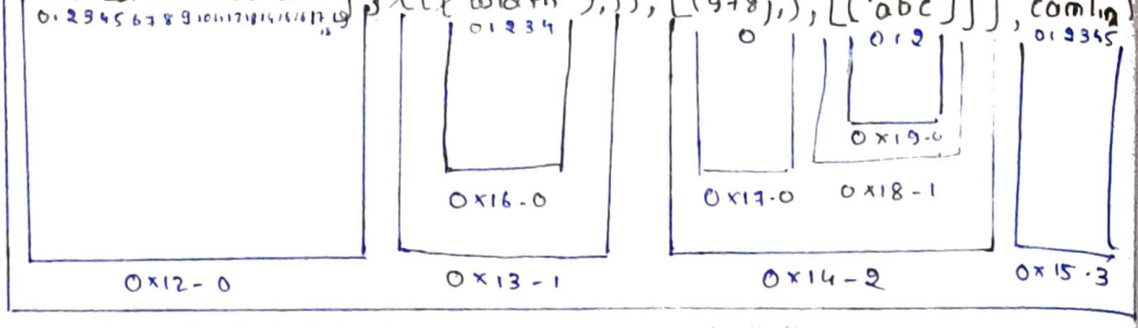
- Tuple is the most secured datatype because it is an immutable datatype & orginal values of the data remains as it is.
- The default value of Tuple datatype is ().

1. t = (((['Monday-tuple', 'List'], 7, 9.6), [1,2,3], 'marker'), True)



1) 'n' = t[0][0][0][0][2]
2) 'p' = t[0][0][0][0][3]
3) 2nd 'd' = t[0][0][0][1][3]
4) 9.6 = t[0][0][2]
5) 2 = t[0][1][1]
6) 'k' = t[0][2][3]
7) T = t[1][0]
8) 2nd 'e' = t[1][3]

2 tu = "tested at laboratory", {{{'width'},)}, [(9+8j,), ['abc']]], 'comline'



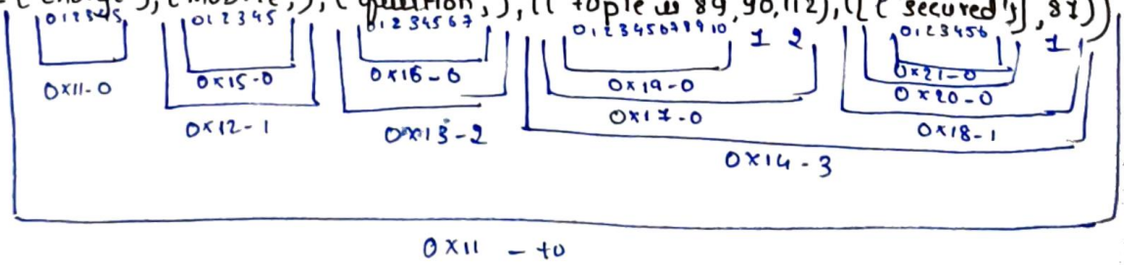
tu - 0x11

1) 2nd 't' = tu[0][3]
2) 1st 'o' = tu[0][4]
3) 1st 'r' = tu[0][14]
4) 'y' = tu[0][19]
5) 2nd 'a' = tu[1][0][2]
6) [8+9j] = tu[2][0][0]
7) 2nd 'a' = tu[2][0][1][0][0][1]
8) 'm' = tu[3][2]

3. t = 1,2,3,[[[[['mnbucxz',),'wonder']),'sticky note',controlling']notuple)))

1) t[3][0][0][0][0][1]
2) 3 = t[2]
3) m = t[3][0][0][0][0][0]
4) z = t[3][0][0][0][0][6]
5) w = t[3][0][0][1][0]
6) r = t[3][0][0][1][5]
7) s = t[3][0][1][0]
8) o -> t[3][0][2][1]
9) d -> t[3][1][6]
10) u - t[3][1][3]

4. tu = ('charge'),('mobile',),('question',),(('tuple is',89,90,112),([['secured']],81))



tu - 0x11

1) 2nd 'a' - tu[0][0]
2) 'b' - tu[2][0][0]
3) 3rd 'n' - tu[3][1][1][3][0]
4) 'w' - tu[1][0][0]
5) 'd' - tu[3][0][0][2]
6) 'r' - tu[3][0][1][0][0][0]
7) 'r' - tu[3][0][0][1][3]
8) 9 - tu[3][0][0][0][10]
9) 90 - tu[3][0][1]
10) 81 - tu[3][1][1]
11) 'b' - tu[0][4]
12) 1st 't' - tu[2][0][4]
13) 1st space - tu[3][0][0][5]
14) 2nd 'r' - tu[3][1][0][0][4]
15) 2nd 'l' - tu[3][0][0][3]