## FILL IN THE BLANKS

1. Elements in a tuple can be of _____ type.
2. Tuples are indexed by an _____ .
3. Tuples are _____, you cannot update or edit the tuples.
4. _____ function is used to converts a sequence data type into tuple.
5. Like lists, Dictionaries are _____ which means they can be changed.
6. A Python dictionary is a _____ of unique keys to values. It is a collection of key-value pairs.
7. The association of a key and a value is called a _____ .
8. To create a dictionary, key-value pairs are separated by _____ .
9. In key-value pair, each key is separated from its value by a _____ .
10. _____ function returns the keys of the dictionary.

## ANSWERS TO FILL IN THE BLANKS

1. dissimilar
2. integer
3. immutable
4. tuple()
5. mutable
6. mapping
7. key-value pair
8. comma
9. colon (:)
10. key()

## SOLVED QUESTIONS

**1.** What is a tuple?

**Ans.** A tuple is an immutable sequence of values which can be of any type and they are indexed by an integer.

**2.** What are the differences between lists and tuples?

**Ans.** The differences between tuples and lists are that tuples cannot be changed unlike lists and tuples use parentheses, whereas lists use square brackets.

**3.** What is the similarity between strings and tuples?

**Ans.** Tuples are immutable like strings, you cannot update or edit a tuple. To change it, you have to create a new one just like strings.

**4.** Tuples can be used as keys in Python dictionaries. True or False?

**Ans.** True

**5.** Tuple is an ordered immutable sequence of objects. Justify your answer.

**Ans.** Tuple is an ordered sequence of objects as each element is accessed by its index and it is immutable as you cannot change the values in place. You cannot update or edit the tuple.

**6.** What is a dictionary?

**Ans.** A Python dictionary is a mapping of unique keys to values. It is a collection of key-value pairs. Dictionaries are mutable which means they can be changed.

**7.** What are the differences between lists and dictionary?

**Ans.** The differences between dictionary and lists are that in lists, elements are enclosed within square brackets, whereas in dictionary, key-value pairs are enclosed within curly braces. In dictionary, index value can be of any data type and is called key, but in list index value is an integer.

**8.** Can list be used as a key to dictionary? State yes or no. Also justify.

**Ans.** Lists cannot be used as key to dictionary as they are mutable.

**9.** Write different ways of creating tuples with one item and print the same.

**Ans.** (a)
```
tup1=5,
print(tup1)
```
(b)
```
tup1=(5,)
print(tup1)
```
(c)
```
tup1=tuple((10,))
print(tup1)
```

**10.** Write a statement to add a single item in a tuple.

**Ans.**
```
>>> t=(10,20,30,40)
t=t+(60,)
```

**11.** Write different ways of creating tuples with the following numbers.
5,10,15,20

**Ans.** (a) `t1=(5,10,15,20)`
(b) `t1=5,10,15,20`
(c) `t1=5,10,15,20,`
(d) `t1=tuple((5,10,15,20))`
(e) `t1=tuple([5,10,15,20])`

**12.** Following are the statements for creating tuples. Find the errors in the statements and rewrite the same after correcting them.

(a) `tup1= 1,6,a,8,`
(b) `tup2=(10)`
(c) `tup1=((0,1,2,3)['my','book'])`
(d) `tup1=((0,1,2,3)(4,5,6))`
(e) `tup1=('a','b','c'[1,2,3])`

**Ans.** (a) `tup1= 1,6,'a',8,`
(b) `tup2=(10,)`
(c) `tup1=((0,1,2,3),['my','book'])`
(d) `tup1=((0,1,2,3),(4,5,6))`
(e) `tup1=('a','b','c',[1,2,3])`

**13.** Consider the following code and find out the error:

(a)
```
tup1=(1,'school',6,7)
print(max(tup1))
```
(b)
```
t=(1,'s','h',5,'p')
t[1]='o'
```
(c)
```
T1=(7)
T2=(1,2,3)
T3=T1+T2
```
(d)
```
T1=(1,2,3)
T2=(4,5,6)
T3=(8,9)
T1,T2,T3=T1,T2
```

(e)
```
T1=(1,2,3)
T4= T1*(2,)
```

(f)
```
tup1=(1,2,3,4)
tup1=tup1+10
print(tup1)
```

(g)
```
tup1=("5",)*'3'
print(tup1)
```

(h)
```
t1=(1,2,3,4)
a,b,c=t1
```

**Ans.** (a) max() will work only if elements in a tuple are of same data type.

(b) Tuples are immutable, we cannot change the value of any element.

(c) '+' operator joins two tuples, here (7) is 'int', so will give error.

(d) For tuple assignment, left hand side and right hand side elements should be the same. T3 is not there on right hand side.

(e) '*' operator requires a tuple and an integer as operands.

(f) The '+' operator simply performs a concatenation with tuples. Here 10 is an integer.

(g) '*' operator requires a tuple and an integer as operands. Here '3' is a string.

(h) For unpacking a tuple, number of variables on the left side of assignment and the number of elements in the tuple must match.

**14.** Find errors, underline them and rewrite the same after correcting the following code:

```
d1=dict[]
i= 1
n=input("Enter number of entries:")
while i<=n:
  a=input("Enter name:")
  b=input("Enter age:")
  d1(a)=b
  i = i+1
l = d1.key[]
for i in l:
  print (i,'\t','d1[i]')
```

**Ans.**
```
d1=dict()
i= 1
n=int(input("Enter number of entries:"))
while i<=n:
  a=input("Enter name:")
  b=input("Enter age:")
  d1[a]=b
  i = i+1
l = d1.keys()
for i in l:
  print (i,'\t',d1[i])
```

**15.** Write the output of the following codes:

(a)
```
>>>d1={1:10,2:20,3:30,4:40,5:50}
>>>d1.items()
>>>d1.keys()
>>>d1.values()
```

(b)
```
d1={1:10,2:20,3:30,4:40}
d2={5:50,6:60,7:70}
d1.update(d2)
print(d1)
```

(c)
```
d1={1:10,2:20,3:30,4:40,5:50,6:60,7:70}
del d1[3]
print(d1)
```

(d)
```
>>>d1={1:10,2:20,3:30,4:40,5:50,6:60,7:70}
>>>d1.pop(5)
>>>print(d1)
```

(e)
```
>>>d1={1:10,2:20,3:30,4:40,5:50,6:60,7:70}
>>>len(d1)
>>>d1.get(6,60)
>>>d1[6]=65
>>>d1[8]=35
>>>print(d1)
>>>d1.clear()
>>>print(d1)
```

**Ans.** (a) dict_items [(1, 10), (2, 20), (3, 30), (4, 40), (5, 50)]
dict_keys [1, 2, 3, 4, 5]
dict_values [10, 20, 30, 40, 50]

(b) {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60, 7: 70}

(c) {1: 10, 2: 20, 4: 40, 5: 50, 6: 60, 7: 70}

(d) 50
{1: 10, 2: 20, 3: 30, 4: 40, 6: 60, 7: 70}

(e) 7
60
{1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 65, 7: 70, 8: 35}
{}

**16.** Write a Python program to display all the elements of the following tuple except 'd'.
t1=('w','o','r','d','s')

**Ans.**
```
t1=('w','o','r','d','s')
t1=t1[0:3]+t1[-1:]
print(t1)
```
**Output:**
```
('w', 'o', 'r', 's')
```

**17.** Write a Python program to remove an element '3' from the following tuple:
t1=(1,2,3,4,5)

**Ans.**
```
t1=(1,2,3,4,5)
L=list(t1)
L.remove(3)
t1=tuple(L)
print (t1)
```
**Output:**
(1, 2, 4, 5)

Alternatively, using pop() method:
with pop() we have to give index of element 3 as shown:
```
t1 = (1,2,3,4,5)
L=List(t1)
L.pop(2) #2 is the indexvalue for element 3
t1=tuple(L)
print (t1)
```
**Output:**
(1, 2, 4, 5)

**18.** t1=(10,20,"book",30,9.5,"item",[12,13],(3,4),30,5,30))

Consider the above tuple 't1' and answer the following questions :

(a) `len(t1)`
(b) `t1[:6]`
(c) `t1[-8:-4]`
(d) `t1[5:]`
(e) `t1.index(20)`
(f) `t1.index(30)`
(g) `t1.index(30,7,10)`
(h) `t1.count(30)`
(i) `t1[:1]*5`
(j) `any(t1)`

**Ans.** (a) `11`
(b) `(10, 20, 'book', 30, 9.5, 'item')`
(c) `(30, 9.5, 'item', [12, 13])`
(d) `('item', [12, 13], (3, 4), 30, 5, 30)`
(e) `1`
(f) `3`
(g) `8`
(h) `3`
(i) `(10, 10, 10, 10, 10)`
(j) `True`

**19.** T1= ("TEA POT",[1,2,'3'], 'S',(3,4,6),"book", 10)

Based on tuple T1 above, write the output for the following expressions:

(a) `T1[1][1]`
(b) `10 in T1`
(c) `50 not in T1`
(d) `T1[4][3]`
(e) `T1[1][2]*3`
(f) `T1[-5:-3]+T1[3]`
(g) `T1[0]+T1[2]`
(h) `T1[4:4]`

**Ans.** (a) `2`
(b) `True`
(c) `True`
(d) `'k'`
(e) `'333'`
(f) `([1, 2, '3'], 'S', 3, 4, 6)`
(g) `'TEA POTS'`
(h) `()`

**20.** Write the output of the following statements:

(a) `>>> tuple("book")`
(b) `>>> tuple([1,2,4])`
(c) `>>>("Riya",)*3`
(d) `>>>max(10,12,73,34)`
(e) `>>>min(40,20,100,60)`

**Ans.** (a) `('b', 'o', 'o', 'k')`
(b) `(1, 2, 4)`
(c) `('Riya', 'Riya', 'Riya')`
(d) `73`
(e) `20`

**21.** Compare the tuples and write the output:

(a)
```
>>>tup1=(1,2,3)
>>>tup2=(1.0,2.0,3.0)
>>>tup1==tup2
```

(b)
```
>>> tup1=('FARIDA','VAIBHAV','MINI','RAGHAV')
>>> tup2=('Ravi','Mala','AMAN','SAKSHI')
>>> tup1<tup2
```

(c)
```
>>>tup1=(10,20,30,40,50)
>>>tup2=(4,5,10,20,30,40,50)
>>>tup2>tup1
```

(d)
```
>>>tup1!=tup2
```

**Ans.** (a) True
(b) True
(c) False
(d) True

**22.**
```
>>>t1=(1,2,3,4)
>>>t2=(10,20,30,40)
>>>t3=(100,200,300,400)
```
**Write the output of the following:**

(a)
```
>>>t1,t2=t2,t1
>>>print(t1)
>>>print(t2)
```

(b)
```
>>>t1,t3=t3,t1
>>>print(t1)
>>>print(t3)
```

(c)
```
>>>t3,t2,t1=t1,t2,t3
>>>print(t1)
>>>print(t2)
>>>print(t3)
```

**Ans.** (a)
```
(10, 20, 30, 40)
(1, 2, 3, 4)
```

(b)
```
(100, 200, 300, 400)
(10, 20, 30, 40)
```

(c)
```
(10, 20, 30, 40)
(1, 2, 3, 4)
(100, 200, 300, 400)
```

**23. Write a Python program to input 'n' names and phone numbers to store it in a dictionary and print the phone number of a particular name.**

```
phone=dict()
i= 1
n=int(input("Enter number of entries:"))
while i<=n:
  a=input("Enter Name:")
  b=input("Enter Phone no:")
  phone[a]=b
  i = i+1
l = phone.keys()
x=input("Enter name to be searched:")
for i in l:
  if i==x:
    print (x,": phone no is:",phone[i])
    break
else:
  print(x, " does not exist")
```

**Ans. Output:**

```
Enter number of entries:4
Enter Name:Riya
Enter Phone no:9999
Enter Name:Aman
Enter Phone no:1111
Enter Name:Pulkit
Enter Phone no:5555
Enter Name: Ziya
Enter Phone no:2222
Enter name to be searched Aman
Aman : phone no is: 1111
```