

Sequence Data Type : TUPLE

Introduction

टपल immutable sequence data type है। ~~निर्देश~~ ^{निर्देश} को parentheses () के अंदर लिखा जाता है तथा इसके element Comma के द्वारा अलग किये जाते हैं। Immutable होने के कारण इसके elements को change या Update नहीं किया जा सकता।

2.

()	Empty Tuple
(1, 2, 3, 4)	Tuple of integer
(1.2, 2.3, 3.4)	" " floating point
('a', 'b', 'c')	" " character
("Delhi", "Baraut")	" " String
(1, "Ram", 14, 2, "Ravi", 15)	" " Mixed value

Output

3.

Creating a Tuple :- ~~निर्देश~~ ^{टपल} बनाने के तीन तरीके हैं -

1. By assign Value
2. By tuple () / By other Sequence
3. Dynamically via keyboard

① Creating tuple by assigning Value :-

Assignment Operator (=) का प्रयोग करके Tuple बनाया जाता है।

Ex.

Ex. num = (1, 2, 3, 4, 5, 6)
 ↓
 assignment Operator

1.
2.

2. By tuple() या By other Sequence —

Tuple function के द्वारा किसी भी Sequence को tuple में बदला जा सकता है।

Ex. a = [1, 2, 3, 4, 5]
b = tuple(a)
print b

Output → (1, 2, 3, 4, 5)

3. Dynamically via keyboard -

input तथा eval function का प्रयोग करके dynamic tuple बनायी जाती है।

eval function के द्वारा elements को टपल के रूप में adjust करता है।

input function element को input करता है।

इसके बाद Tuple function का प्रयोग करके किये गये element tuple में बदल दिये जाते हैं।

Ex. a = eval(input("Enter five element"))
b = tuple(a)
print(b)

Accessing the Tuple : — Tuple को दो प्रकार से किया जाता है -
access

1. Accessing whole Tuple
2. Accessing individual element

1. Accessing whole Tuple :-

Tuple बनाने के समय टपल के elements को index no दिया जाता है।
 टपल के नाम का प्रयोग करके सम्पूर्ण टपल access की जा सकती है।

Ex.

Ex. `t = (3, 2, 1, 4)`
 `print(t)`

Output

2.

2. Accessing individual element :-

Index no. का प्रयोग करके टपल के प्रत्येक element को access किया जा सकता है।

Ex.

Ex. `t = (3, 2, 1, 4)`
 `print(t[2])`

Output

Output 1

3.

Traversing the Tuple :-

टपल के नाम का प्रयोग करके इसके element को individual access करना traversing है।

Ex.

Ex. `t = (1, 2, 3, 4)`
 `for a in t:`
 `print(a, end=" ")`

Tuple Operations -

4.

1. Combining the Tuple →

Concatenation Operator '+' का प्रयोग करके दो या अधिक टपल को जोड़ा जा सकता है।

Ex. $t_1 = (2, 3, 4)$
 $t_2 = (5, 6, 7)$
 $t = t_1 + t_2$
 $\text{print}(t)$
 Output $\rightarrow (2, 3, 4, 5, 6, 7)$

2. Replicating the Tuple -

* ऑपरेटर का प्रयोग करके टपल को दिये गये नम्बर तक repeat किया जा सकता है।

Ex. $t_1 = (2, 3, 4)$
 $t = t_1 * 2$
 $\text{print}(t)$
 Output $\rightarrow (2, 3, 4, 2, 3, 4)$

3. Slicing the Tuple -

एक टपल में से इसके छोटे भाग को निकालने की प्रक्रिया टपल Slicing कहलाती है।

Ex. $t = (11, 22, 33, 44, 55, 66, 77, 88, 99)$
 $\text{print}(t[1, 7, 3]) \rightarrow 22, 55$
 $\text{print}(t[2:6]) \rightarrow 33, 44, 55, 66$
 $\text{print}(t[4:]) \rightarrow 55, 66, 77, 88, 99$
 $\text{print}(t[:6]) \rightarrow 11, 22, 33, 44, 55, 66$
 $\text{print}(t[::1]) \rightarrow 11, 22, 33, 44, 55, 66, 77, 88, 99$

4. Unpacking the Tuple -

टपल के प्रत्येक element को अलग-अलग Variable में स्टोर करना Unpacking कहलाता है।

के हो ग
जोड़ा जा

```
Ex. t = (01, "Rohan", 86)
    jno, name, marks = t
    print(jno, name, marks)
```

NOTE →

Use किये गये Variable की तथा tuple की संख्या equal होनी चाहिए।

Functions of tuple -

1. len() :- टपल में element की संख्या गिनने के लिए इसका प्रयोग होता है।
Element की संख्या = Tuple की लम्बाई

```
Ex. t = (2, 3, 5)
    print(len(t))
```

2. max() मैक्स :- यह फंक्शन टपल में से maximum element को return करता है।

```
Ex. t = (2, 3, 5)
    print(max(t))
```

Output 5

3. min() :- यह फंक्शन टपल में से सबसे छोटी Value return करता है।

```
Ex. t1 = ("Anil", "Kiran", "Kailash", "Hemant")
    print(min(t1))
```

Output Anil

4.

Ex.

Output

5.

Ex.

- Out

No

4. काउंट (count()) :- इस फंक्शन का प्रयोग किसी की उपस्थिति को गिनने के लिए किया जाता है।

Ex. $t = (2, 3, 5, 2, 5, 2)$
`print(t.count(2))`

Output → 3

5. इंडेक्स (index()) :- इस फंक्शन का प्रयोग स्ट्रिंग में दिये गये element की प्रथम उपस्थिति को जाँच करने के लिए किया जाता है।

Ex. $t = (2, 3, 5, 2, 5, 2)$
`print(t.index(2))` — 0
`print(t.index(2, 2))` — 3
`print(t.index(5, 3, 5))` — 4

Output → 0
→ 3
→ 4

NOTE: यह फंक्शन पहली उपस्थिति को check करता है।