

@pythonessdatadiaries

Python Tuple: Immutability + Concept-based Questions & Answers (for Beginners)

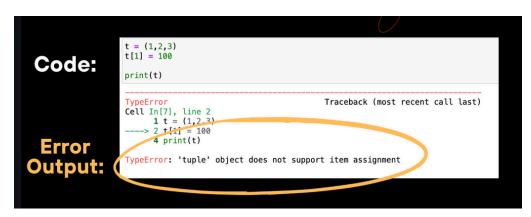
1. What is Tuple Immutability?

- In Python, Tuples are immutable, which means:
- Once a tuple is created, you cannot modify it.
- You cannot add, remove, or change its elements.
- But you can read/access its elements.

2. Why Tuple Immutability is Important?

- Safer: Data cannot accidentally change.
- Faster: Tuples are quicker to process than lists.
- Useful: Perfect for storing fixed data (like dates, coordinates, RGB values).

3. Example: Tuple Immutability



✓ You can read values, but cannot change them.

Concept-based Tuple Questions with Answers:

- What is a Tuple?
- **☑** Tuple is an ordered, immutable collection of items, written with round brackets ().
- 2 How do you create a tuple with one element?
- ✓ Add a comma after the single item:

```
t = (10,)
print(type(t))
<class 'tuple'>
```

- Without comma, it will not be a tuple.
- 3 Can we change values inside a tuple?
- ▼ No. Tuples are immutable.
- 4 Are tuples faster than lists?
- ✓ Yes. Tuples are faster because they are immutable and smaller in size.
- 5 Can you delete a tuple?
- Yes. You can delete the entire tuple using del:

But you cannot delete specific elements inside a tuple.

- 6 What happens if you forget the comma in a single-element tuple?
- ✓ It becomes not a tuple, just that data type:

```
code:
    t = (10)
    print(type(t))

<class 'int'>
```

- Always add a comma: (10,)
- What is the difference between list and tuple?
- **/**

Feature	List	Tuple
Brackets	[]	()
Mutable	Yes	No
Faster	No	Yes

- 8 Can you use a tuple as a dictionary key?
- ✓ Yes! Because tuples are immutable and hashable.

- When should you use tuple over list?
- ☑ When your data must not change use Tuple (like dates, days, configurations).