

15 – Tuples

Le **tuple** sono strutture dati **ordinate e immutabili** in Python, simili alle liste ma con alcune differenze chiave. Essendo immutabili, una volta create non possono essere modificate: non si possono aggiungere, rimuovere o cambiare elementi. Questo le rende **più sicure** per dati costanti e **più efficienti** in termini di memoria.

Creazione di tuple

Si possono creare usando parentesi tonde () o senza parentesi (tuple implicite):

```
t1 = (1, 2, 3)
t2 = 4, 5, 6
t3 = ("a", "b", "c")
```

Tuple vuota e con un solo elemento

```
vuota = ()
un_elem = (5,) # la virgola è obbligatoria
```

Accesso agli elementi

Le tuple sono **indicizzate** e supportano lo slicing come le liste:

```
t = (10, 20, 30, 40)
print(t[0])      # 10
print(t[-1])     # 40
print(t[1:3])    # (20, 30)
```

Operazioni sulle tuple

- **Concatenazione:** `t1 + t2` → crea una nuova tupla
- **Ripetizione:** `t1 * 2` → ripete gli elementi
- **Verifica presenza:** `20 in t`
- **Lunghezza:** `len(t)`
- **Min/Max/Somma** (solo su elementi numerici): `min(t)` , `max(t)` , `sum(t)`

Tuple annidate

Le tuple possono contenere altre tuple:

```
t = (1, 2, (3, 4))  
print(t[2][0]) # 3
```

Vantaggi delle tuple

- **Immutabili**, quindi sicure per dati che non devono cambiare.
- **Più leggere** e veloci delle liste in termini di memoria.
- Possono essere usate come **chiavi di un dizionario** (le liste no).
- Ideali per rappresentare **record o coordinate**.

Vuoi procedere con il **paragrafo 16 – Swapping Variables?**