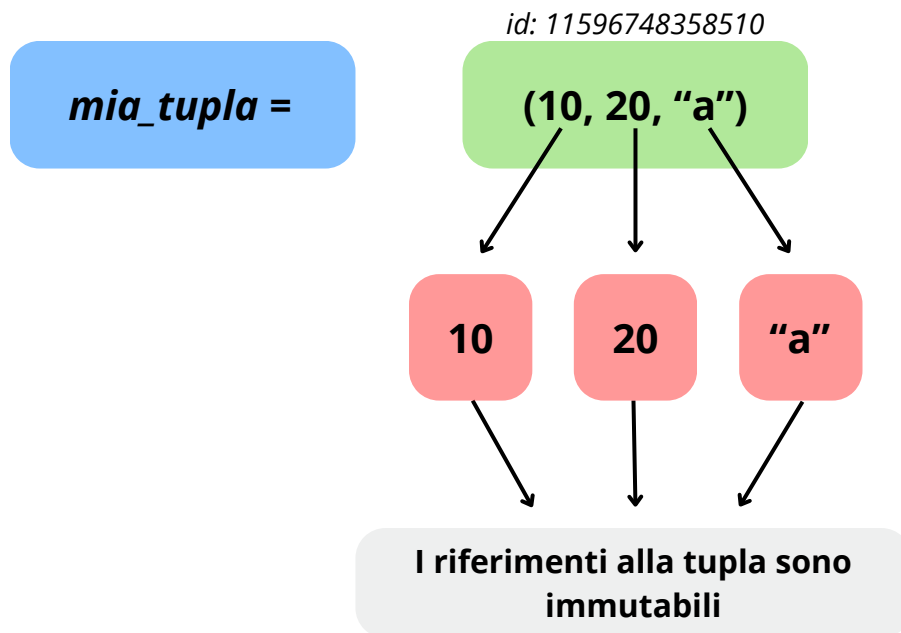


# Gestione della memoria per le Tuple

## Inizializzazione della Tupla

```
mia_tupla = (10,20,"a")
```

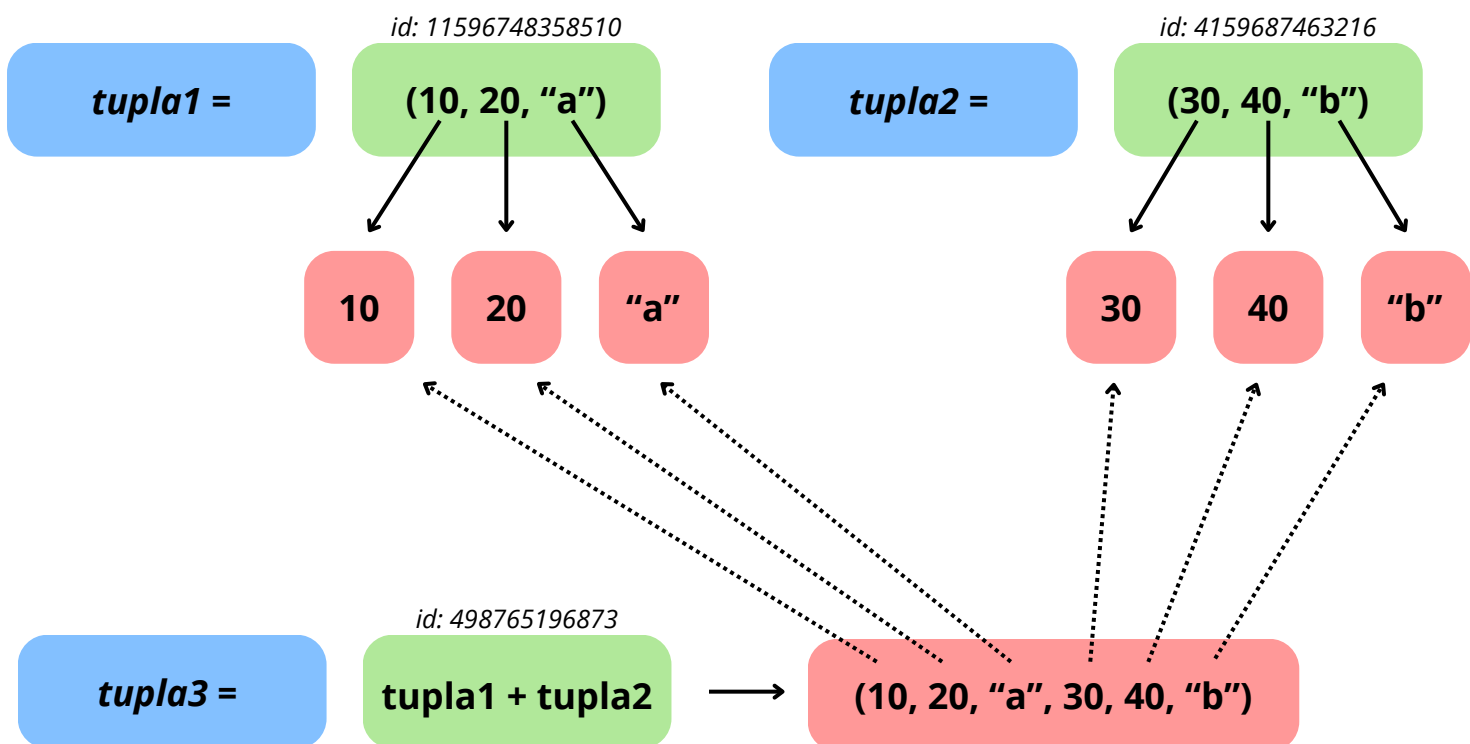


Quando creiamo una tupla con elementi di tipo diverso:

- Python alloca memoria per l'oggetto tupla stesso
- Ogni elemento viene conservato separatamente in memoria
- La tupla contiene riferimenti agli oggetti, non gli oggetti stessi
- I tipi diversi (interi, stringhe) sono gestiti in modo uniforme

## Concatenazione di Tuple

```
tupla1 = (10,20,"a")  
tupla2 = (30,40,"b")  
tupla_concatenata = tupla1 + tupla2
```



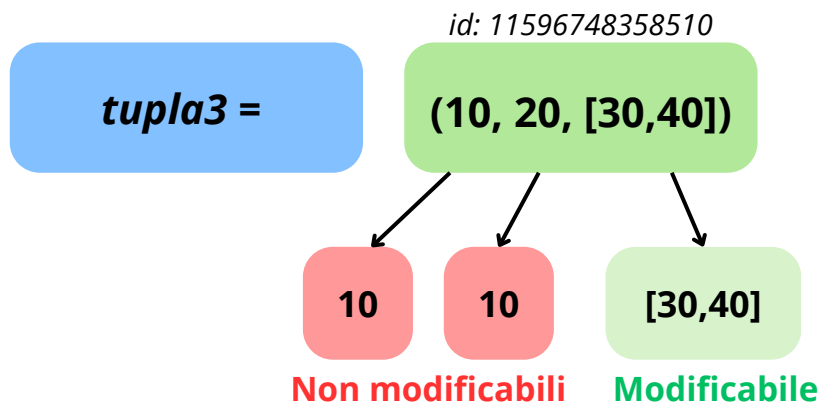
Nella concatenazione di Tuple:

- Viene creata una nuova tupla in memoria
- Le tuple originali non vengono modificate
- I riferimenti agli oggetti vengono copiati nella nuova tupla
- Gli oggetti stessi non vengono duplicati in memoria

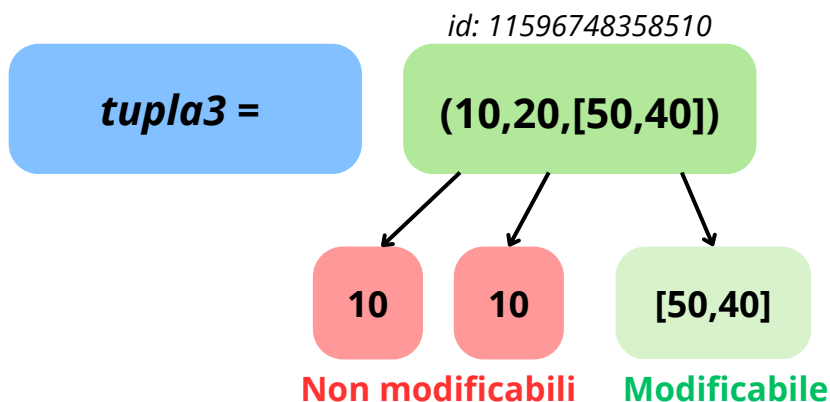
## Concatenazione di Tuple

```
tupla_mista = (10,20,[30,40])
```

*#Possiamo modificare la lista, all'interno della tupla*  
`tupla_mista[2][0] = 50`



### Dopo la modifica



Nelle Tuple annidate con liste

- La tupla contiene un riferimento all'oggetto
- La tupla stessa rimane immutabile (non può cambiare riferimenti)
- Il contenuto della lista può essere modificato, poiché le liste sono mutabili
- In memoria, viene modificato l'oggetto lista, non la tupla