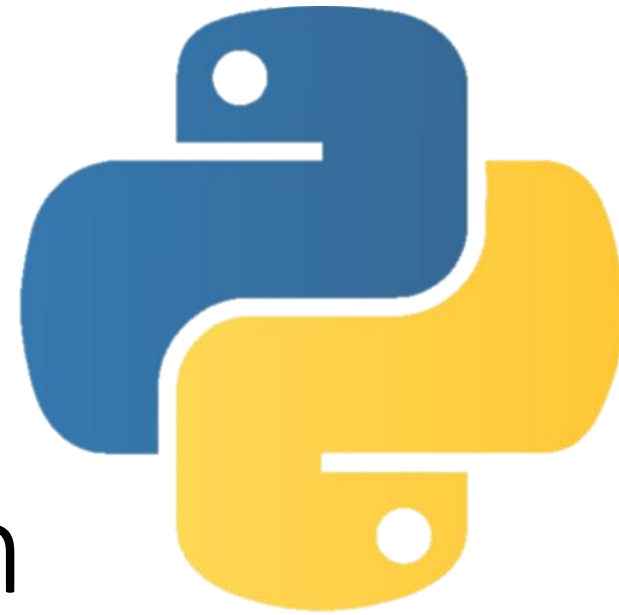


Curso Python
2025

Día 5:
Excepciones e iteración



Hoy:

- ✓ Repaso: Slicing y mutabilidad
- ✓ Excepciones
- ✓ Iteradores
- ✓ Loops



Repaso: Slicing

Extraer un “rango” de elementos de una secuencia.
Indexar pero en vez de **int** utilizamos:

- Notación → **start:stop:step**

`my_seq[start:stop:step]`

- Objeto → **slice(start, stop, step)**

`my_seq[slice(start, stop, step)]`

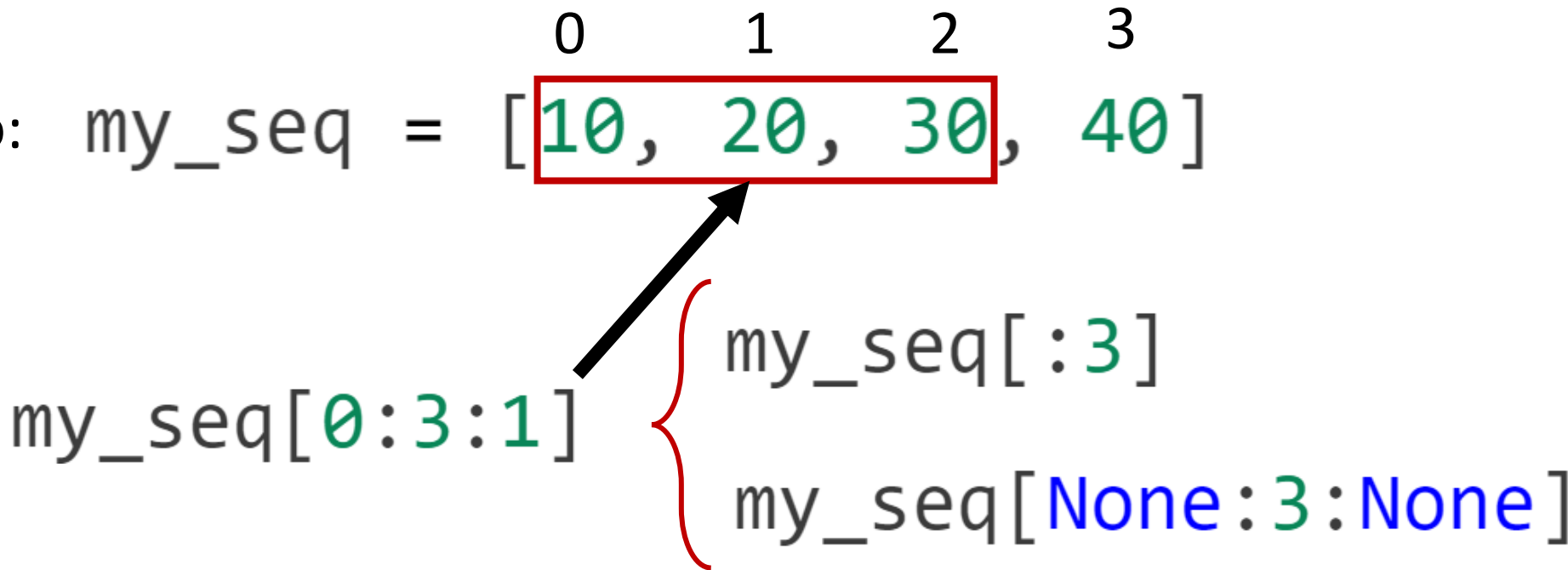
stop no incluido



Recordatorio: `my_sequence[x] == my_sequence.__getitem__(x)`

Repaso: Slicing

Ejemplo: `my_seq = [10, 20, 30, 40]`

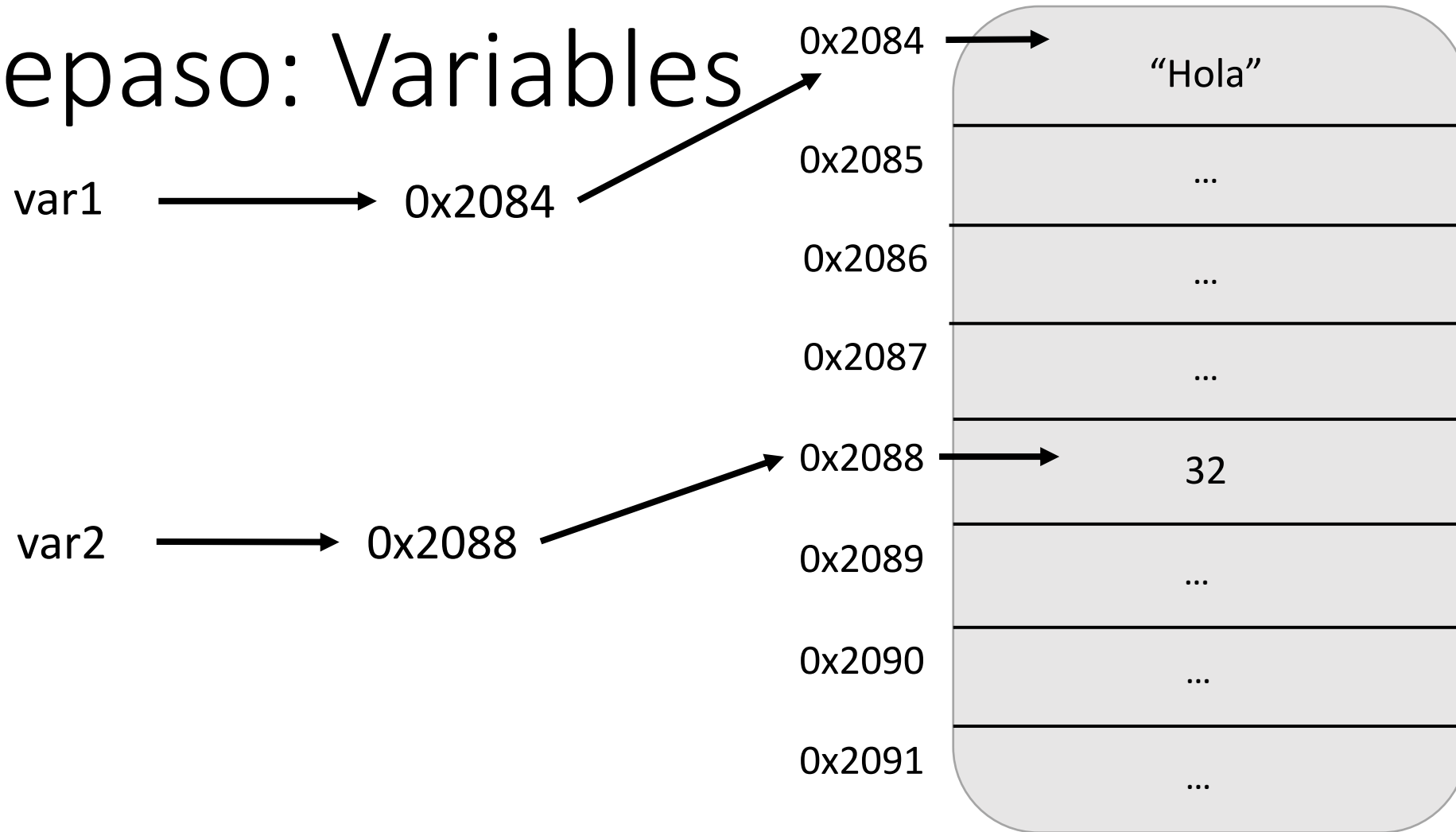


0	1	2	3
10	20	30	40

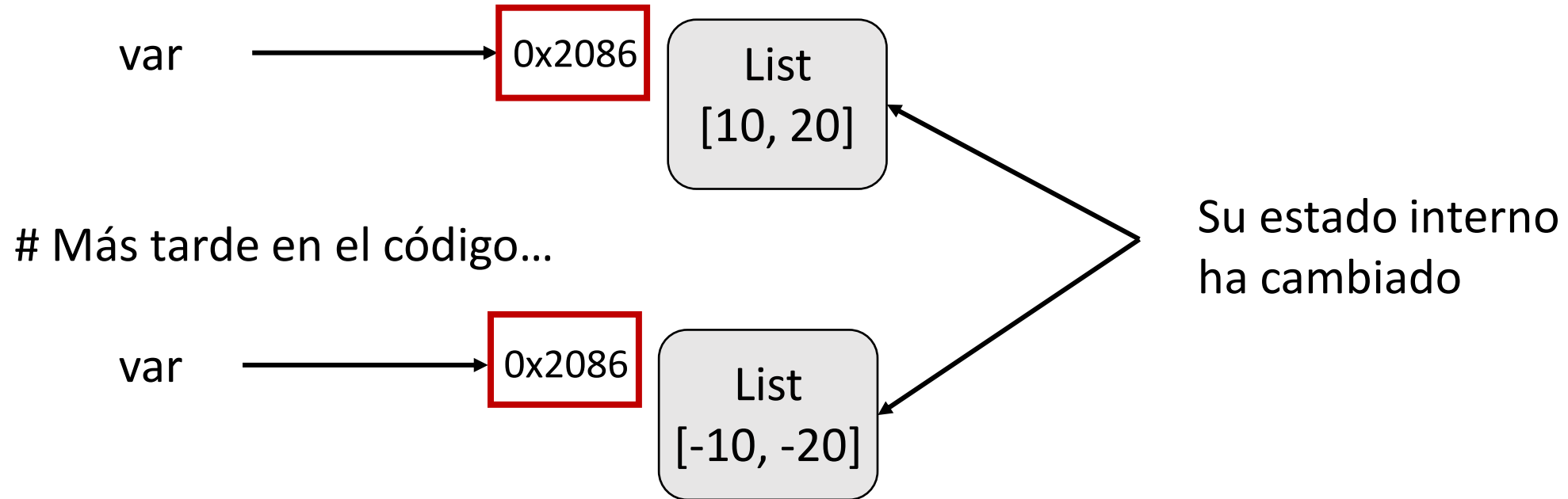
`my_seq[0:3:1]` $\left\{ \begin{array}{l} \text{my_seq}[:3] \\ \text{my_seq}[\text{None}:3:\text{None}] \end{array} \right.$

Nota: Si no especificamos o damos valor **None** a start, stop o step por defecto toman valores 0, `len(my_seq)`, 1

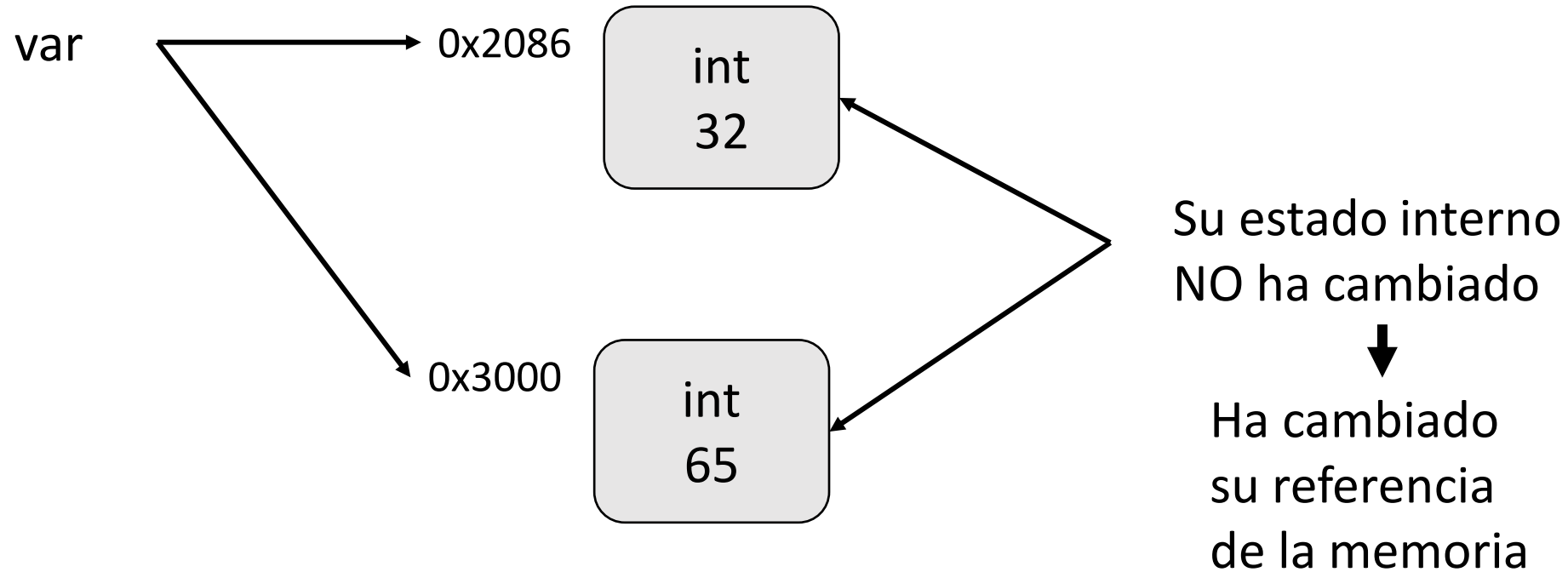
Repaso: Variables



Repaso: Objetos mutables



Repaso: Objetos inmutables



Cuando realizamos una operación entre int obtenemos un nuevo objeto **int** !!!

Repaso: Shared references

Cuidado:

- Es posible que un objeto inmutable contenga objetos mutables
- Cuando asignamos a una variable otra se comparte la referencia !!!

