

Mémo – Plusieurs actions avec un seul bouton Tkinter

Contexte

Dans Tkinter, un bouton (`tk.Button`) ne peut avoir **qu'un seul paramètre `command`**. Si l'on écrit deux fois `command=...`, le dernier écrase le premier.

Exemple **incorrect** :

```
bouton = tk.Button(root, text="Test",
                    command=lambda: fonction1(),
                    command=lambda: fonction2()) # ✗ seule cette ligne sera
prise en compte
```

Solution simple avec une seule fonction `lambda`

On peut exécuter plusieurs instructions **dans un seul `lambda`** :

```
bouton = tk.Button(root, text="Test",
                    command=lambda: (fonction1(), fonction2()))
```

Les deux fonctions seront exécutées **l'une après l'autre** au clic du bouton.

Exemple concret (sélection de langue)

```
def bouton_action(langue):
    changer_langue(langue)
    charger_aide(langue)

bouton_francais = tk.Button(zone_drapeaux, image=img_fr,
                            command=lambda: bouton_action("fr"))
bouton_anglais = tk.Button(zone_drapeaux, image=img_en,
                           command=lambda: bouton_action("en"))
```

Une seule fonction (`bouton_action`) gère toutes les langues, et chaque bouton transmet simplement le code de langue à la fonction.

Astuce bonus

On peut écrire une seule ligne de `lambda` sans fonction intermédiaire :

```
command=lambda: (changer_langue("fr"), charger_aide("fr"))
```

mais utiliser une fonction séparée garde le code plus lisible.

À retenir

- Une seule `command=` par bouton.
 - Pour plusieurs actions → les regrouper dans un `lambda` ou une fonction.
 - Transmettre des paramètres → `lambda: ma_fonction(param)`.
-

Mémo réalisé pour le projet *Convertisseur de bases numériques*

Jean-Marc Alcaïno – 2025