



## 1. Objectifs

- Comprendre et savoir utiliser les widgets tkinter de Python.
- Découvrir la programmation en Python utilisant une interface graphique.

## 2. Exercice 01

- Ecrire un programme **bonjour.py** Python utilisant la bibliothèque tkinter qui permet :
- De tracer formulaire de taille 400 x 80, de fond ivory, au centre de ce formulaire il y'a:
  - Un label avec le texte 'Bonjour tout le monde !'
  - Un bouton avec le texte 'Quitter'
- d'apparaître la fenêtre représenté par l'image suivante



- Déterminer les autres propriétés pour avoir le même rendu ci-dessus

```
# -*- coding: utf-8 -*-
# script bonjour.py
from tkinter import *

# Création de la fenêtre principale (main window)
Mafenetre = Tk()
Mafenetre.geometry('400x80')
Mafenetre.title("Application01")
Mafenetre.configure(background='ivory')
# Création d'un widget Label (texte 'Bonjour tout le monde !')
Labell = Label(Mafenetre, text = 'Bonjour tout le monde !',
               fg = 'red',bg="ivory")
Labell.config(font=("Courier", 20))
# Positionnement du widget avec la méthode pack()
Labell.pack()

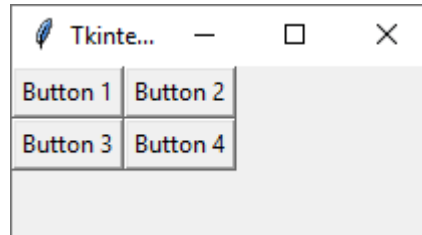
# Création d'un widget Button (bouton Quitter)
Bouton1 = Button(Mafenetre, text = 'Quitter',bg="sky blue",
                 width=15, command = Mafenetre.destroy)
Bouton1.pack()

# Lancement du gestionnaire d'événements
Mafenetre.mainloop()
```



## Exercice 02

- En utilisant le système de gestion des widgets Tkinter Grid Layout, écrire un programme en Python Tkinter qui affiche la vue suivante:



# coding: utf-8

```
from tkinter import *
root = Tk()
root.geometry("300x100")
root.title("Tkinter Grid Layout")
```

# création des boutons

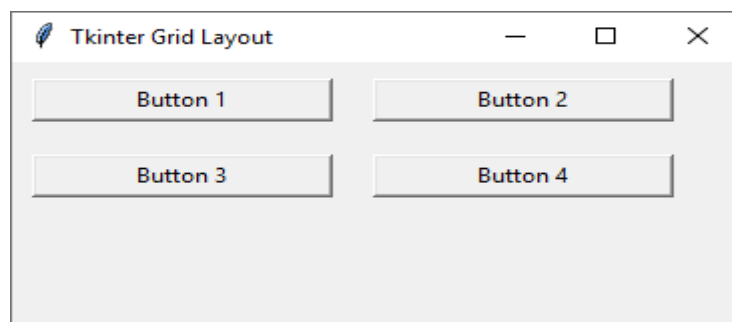
```
btn1 = Button(root , text = "Button 1" )
btn2 = Button(root , text = "Button 2" )
btn3 = Button(root , text = "Button 3" )
btn4 = Button(root , text = "Button 4" )
```

# emplacement des boutons avec la method grid()

```
btn1.grid(row = 0 , column = 0 )
btn2.grid(row = 0 , column = 1 )
btn3.grid(row = 1 , column = 0 )
btn4.grid(row = 1 , column = 1 )
```

```
root.mainloop()
```

- Améliorer l'affichage de la boîte de dialogue de l'exercice précédent (Exercice2) en ajoutant les paramètres width, padx et pady aux boutons:



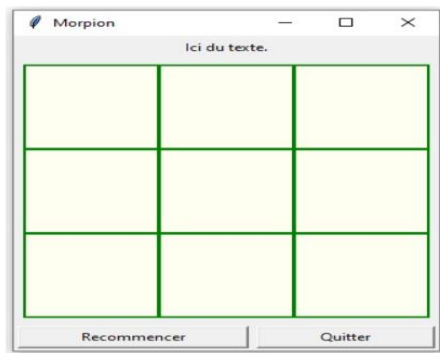


- Reprendre l'exercice précédent (Exercice2) en utilisant la méthode place():

### Exercice 03

Ecrire un programme en **python Tkinter** qui affiche à l'utilisateur une fenêtre affichant un **bouton de commande** qui affiche au click un message "**Hello World !**" sur un **label**.

### Exercice 04



1. écrire une application tkinter python qui fait apparaître la fenêtre d'interface du jeu de **morpion**.
2. *Pour jouer une partie de morpion, il suffit de tracer sur une feuille blanche une grille de 3 cases sur 3 (selon les variantes, il est possible d'augmenter le nombre de cases). Le but du jeu est d'aligner avant son adversaire 3 symboles identiques horizontalement, verticalement ou en diagonale.*
3. La fenêtre d'interface du morpion doit contenir :
  1. Un **Label** (zone d'affichage de texte) où s'afficheront les messages nécessaires au jeu ;
  2. Un **Canvas** (zone d'affichage graphique) carré de 301 pixels de côté ;
  3. Deux **Buttons**, un bouton "Quitter" et un Bouton "Recommencer".
4. Pour cela :
  1. Créez un nouveau fichier JeuMorpion.py.
  2. Dans ce fichier (vide) définissez chaque widget et disposez-le dans la fenêtre.
  3. La grille contient 9 carrés blancs de 97 pixels de côté.

#### 4. Solution

```
##-----Importation des Modules-----##
from tkinter import *
##----- Définition des Variables globales -----##
```

```
##----- Définition des Fonctions -----##
##-----Création de la fenêtre-----##
fen = Tk()
fen.title('Morpion')
##-----Création des zones de texte-----##
message=Label(fen, text='Ici du texte.')
message.grid(row = 0, column = 0, columnspan=2, padx=3,
pady=3, sticky = W+E)
##-----Création des boutons-----##
bouton_quitter = Button(fen, text='Quitter', command=fe
n.destroy)
bouton_quitter.grid(row = 2, column = 1, padx=3, pady=3
, sticky = S+W+E)

bouton_reload = Button(fen, text='Recommencer')
bouton_reload.grid(row = 2, column = 0, padx=3, pady=3,
sticky = S+W+E)
##-----Création du canevas-----##
dessin=Canvas(fen, bg="ivory", width=301, height=301)
dessin.grid(row = 1, column = 0, columnspan = 2, padx=5
, pady=5)
##-----La grille-----##
lignes = []
for i in range(4):
    lignes.append(dessin.create_line(0, 100*i+2, 303, 1
00*i+2, width=3, fill='green'))
    lignes.append(dessin.create_line(100*i+2, 0, 100*i+
2, 303, width=3, fill='green'))
##-----Programme principal-----##
fen.mainloop()# Boucle d'attente des événements
```