

TP1 Informatique 3

Filière : MIP Niveau : 3 Année académique : 2021-2022

Dr. Ing. NOUBISSI Justin-Herve

L'objectif de ce TP est d'apprendre à vous familiariser avec la création des interfaces graphiques sous python. Dans ce TP, nous travaillerons principalement sur l'environnement de développement **pycharm**, la librairie **tkinter**, qui est une librairie permettant la construction des interfaces graphiques.

- 1- La classe **Tk** de tkinter permet la création d'une fenêtre et son initialisation. Créez une Classe **Root** qui hérite de la classe **Tk**, redéfinit la méthode `__init__()` de la classe Tk de la manière suivante :
 - titre de la fenêtre (*title*) : **TP1 info 3**,
 - taille minimale de la fenêtre (*minsize*) : 740x600
 - taille maximale de la fenêtre (*maxsize*) : 1500x900
- 2- Ajoutez à la méthode `__init__()` de la classe Root, une **Frame** nommée *cframe* avec la caractéristique **pack(expand= 1, fill=BOTH, padx=5, pady=5)**
- 3- Créez un **Canvas** nommé *canvas* sur la Frame cframe précédente avec pour caractéristiques :
 - Bg= "white"
 - Définir la grille (*grid*) du canvas avec les valeurs **row=0, column=0, sticky=NW+SE**
- 4- Créez un scrollbar horizontal (**hbar**) et un scrollbar vertical (**vbar**) sur cframe de la manière suivante :
 - `hbar=Scrollbar(cframe, orient=HORIZONTAL)`
 - `hbar.grid(row=0, column=1, sticky=NW + SE)`
 - `vbar=Scrollbar(cframe, orient=VERTICAL)`
 - `vbar.grid(row=0, column=1, sticky=NW + SE)`
- 5- Connectez vos scrollbar sur le canvas de la manière suivante :

```
canvas.configure(  
    xscrollcommand=hbar.set,  
    yscrollcommand=vbar.set,  
    scrollregion=(0, 0, 800, 600),  
)  
  
hbar.configure(command=canvas.xview)  
vbar.configure(command=canvas.yview)
```
- 6- Créez 13 LabelFrame sur le canvas, sur la colonne 0 et sur les lignes de 2 à 14. Exemple de création d'un LabelFrame :
 - `self.labelFrame1 = LabelFrame(canvas, text="", width=150, height=40)`
 - `self.labelFrame1.grid(column=0, row=2, columnspan=2, padx=100, pady=20)`

- 7- Sur votre fenêtre **self**, Créez un bouton nommé **self.button_loadfile** pour charger le chemin complet d'un fichier excel dans le **labelFrame1**.

La démarche de création du bouton est la suivante :

- Vous utiliserez la fonction prédéfinie **Button(...)** pour créer votre avec les caractéristiques :
 - o **text** = "Charger le fichier",
 - o **command** = **self.nom_de_votre_fonction_chargement_fichier** (nommée ici **fileDialog**),
 - o **bg**='white'

Votre bouton devra être placé sur la grille (grid), à la colonne 1 et la ligne 1

- Votre fonction de chargement de fichier (**self.fileDialog**) se fera en utilisant la fonction prédéfinie **askopenfilename(...)** avec les caractéristiques :
 - o **initialdir**="/",
 - o **title**="Choisir le fichier",
 - o **filetype**=(("Excel", "*.xlsx"), ("Tous les fichiers", "*"))

Le chemin d'accès complet de votre fichier devra être sauvegardé sur votre fenêtre **self** dans un label. Pour cela, vous devriez :

- o Créer un **Label** (que vous nommerez **self.label**) sur votre **LabelFrame1** avec pour caractéristique **text= ""**.
- o Vous devriez ensuite placer votre label (**self.label**) sur la grille (**grid**) à la colonne 1 et la ligne 2.
- o Enfin, vous devriez affecter le chemin d'accès complet de votre à votre label (**self.label**) en utilisant le mot la fonction prédéfinie **configure** avec pour caractéristique **text = self.nom_fichier**

- 8- Ecrire la fonction **button_parametres(self)**, permettant de créer, sur votre fenêtre **self**, un bouton nommé **self.button_parametres** pour afficher les noms des paramètres (1^{ère} ligne contenant les entêtes de vos données) dans votre fichier excel.

- Votre bouton **self.button_parametres** devra être situé sur votre label **self.labelFrame5** avec les caractéristiques :
 - o **Text** = "Afficher les paramètres"
 - o **Command** = **self.afficheparametres** (**afficheparametres** étant la fonction à écrire permettant l'affichage effectif)

Votre bouton **self.button_parametres** devra être placé sur la grille (**grid**), à la colonne 1 et sur la ligne 3

- 9- **Ecriture de votre fonction **afficheparametres(self)** :**

- Afin de lire le contenu de votre fichier excel, vous devriez importer la librairie **pandas**.
- Utilisez la fonction prédéfinie **read_excel** de **pandas** pour lire le contenu de votre fichier que vous affecterez dans une variable nommée **f** de la manière suivante :

f = pandas.read_excel(self.nom_fichier, skiprows=0)

skiprows permet de supprimer la ligne 0 d'Excel, qui est l'entête d'Excel (lettres alphabétiques A,B,C,) et de n'avoir que le contenu de votre fichier

- Récupérez les entêtes des colonnes de votre fichier excel (ligne 0 du contenu de votre fichier excel) et stockez les dans le tableau **parametres** en utilisant la fonction prédéfinie **head** de la manière suivante :

parametres=f.head(0)

- Créez une **Listbox** sur votre **self.labelFrame6** et affectez la dans une variable nommée **liste**
#cette Listbox contiendra à la fin la liste des entêtes-colonnes de votre fichier excel
- Parcourez votre tableau **parametres**, récupérez chaque élément et insérez le dans votre listbox **liste**.
- Affichez votre listbox **liste** en utilisant la fonction prédéfinie **pack**.