

TP Tkinter ISN

Création d'interface graphique (GUI : Graphical User Interface) permettant l'interaction entre le programme et l'utilisateur .

L'objectif de ce TP est de découvrir quelques possibilités de ce module (Tkinter) qui sont très nombreuses : création d'une fenêtre (Tk) , de bouton (Button) , de cases à cocher (Checkbutton) , d'étiquette (Label) , de zone de texte simple (Entry), de menu (Menu) , de zone graphique (Canvas), de cadre (Frame)

On peut également gérer de nombreux événements : clic de la souris, déplacement de la souris, appui sur une touche du clavier, top horloge ...

I- Un premier essai

Essayer ce premier script qui crée votre première fenêtre graphique :

```
from Tkinter import *
fen1=Tk( )
tex1=Label(fen1,text='bonjour tout le monde',fg='red')
tex1.pack( )
bou1=Button(fen1,text='Quitter',command=fen1.destroy)
bou1.pack( )
fen1.mainloop( )
```



Explication :

- première ligne : on importe tout le module Tkinter qui contient des classes c'est à dire des morceaux de programmes réutilisables
- ligne 2 : on utilise le classe Tk() pour créer une fenêtre appelée fen1
- ligne 3 : on utilise la classe Label pour créer un autre objet (un widget) nommé tex1. Le premier argument transmis fen1 indique que le nouveau widget sera contenu dans un autre widget préexistant que nous appellerons le parent (fen1) . Ainsi fen1 est le parent de tex1
- ligne 4 : La méthode pack() réduit automatiquement la taille de la fenêtre parent afin qu'elle soit juste assez grande pour contenir les widgets enfants
- ligne 5 : facile à comprendre
- ligne 6 : fen1.mainloop() est très importante, c'est elle qui met en marche le programme

II- Les canevas

Un canevas est une fenêtre rectangulaire délimitée dans laquelle on peut installer divers dessins et images avec des méthodes spécifiques. Un Canvas est un enfant d'une fenêtre Tk().

Ouvrir le fichier dessine_moi_une_ligne.py sur mon site et observer son fonctionnement

Exercice : Modifier ce programme selon le cahier des charges suivants :

- Avoir un canevas de largeur 500 et de hauteur 650
- Ne plus disposer que de trois couleurs : cyan , marron ou green
- Ne plus avoir que des lignes horizontales ou verticales dont les extrémités se confondent avec les bord du canevas
- ajouter une fonction dessine_croix qui tracera deux lignes rouge en croix au centre du canevas

NB : Remplacer la méthode create_line par create_oval, create_rectangle, create_arc, ou encore create_polygon.

A vous d'aller chercher les paramètres dont ils ont besoin

La méthode create_oval permet de créer des cercles

Un premier TP

Créer un court programme qui dessinera les 5 anneaux olympiques dans un rectangle de fond blanc.

On entrera les coordonnées des centres de ces cercles dans une liste de même pour les 5 couleurs à utiliser.

On pourra remplacer l'attribut « fill= "red" » par l'attribut « outline="red"» afin d'obtenir un cercle vide de contour rouge (par exemple)

III- Récupérer un clic de souris sur un canvas

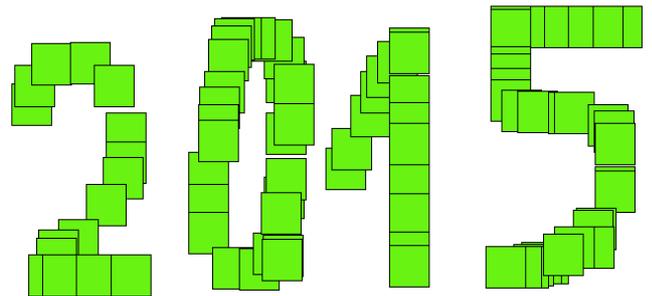
Ouvrir le fichier clic_souris.py .

Ce programme dessine un carré à l'endroit du clic de la souris.

Pour cela, on utilise l'événement associé au clic gauche de la souris (vous trouverez en annexe d'autres événements)

A noter ici :

- l'utilisation de la méthode *bind()* du canevas pour associer au clic gauche de la souris (<Button-1>) l'appel de la fonction *clic()* définie dans le programme
- l'argument *event* passé à la fonction *clic()* active le gestionnaire d'événement et permet de récupérer les coordonnées de la souris.



effacer quitter

Exercice :

Modifier ce programme pour générer des cercles dont le diamètre varie aléatoirement de 10 à 25 et dans une palette de huit couleurs choisies aléatoirement par le programme

Quelques noms de bindings courants

Nom	Signification
<KeyPress>	Pression sur une touche
<KeyPress-a>	Pression sur la touche 'A' (minuscule)
<KeyPress-A>	Pression sur la touche 'A' (majuscule)
<Return>	Pression sur entrée
<Escape>	Touche d'échappement
<Up> <Down>	Pression sur les flèches
<Button-1>	Click souris gauche
<Button-2>	Click souris milieu (ou les deux)
<Button-3>	Click souris droit
<ButtonRelease>	Fin de click gauche
<Motion>	Mouvement de la souris
<B1-Motion>	Mouvement de la souris avec click gauche
<Enter> <Leave>	Entrée et sortie souris d'un widget
<Configure>	Redimensionnement de la fenêtre
<Map> <Unmap>	Ouverture et iconification de la fenêtre

IV- Interaction avec l'utilisateur

On va utiliser ici le module `tkMessageBox` qui offre un certain nombre de boîtes de dialogue standard pour l'affichage de messages (avec une icône associée). Les trois boîtes de bases sont :

- `showinfo` pour donner de l'information
- `showwarning` pour signaler un warning
- `showerror` pour signaler une erreur

Ouvrir le fichier `motdepasse.py` et observer son fonctionnement.

Plusieurs choses à noter ici :

- l'utilisation du widget `Entry` qui permet une interaction
- la déclaration de la variable `Motdepasse` pour récupérer ce que rentre l'utilisateur

V- D'autres exemples

a) Ajout d'un menu déroulant

Ouvrir le fichier `menu_deroulant.py`

La plupart des applications modernes possèdent un menu déroulant. Cet exemple illustre la méthode permettant d'en ajouter un facilement à votre projet.

L'argument `tearoff=0` permet de rendre le menu non détachable.

Remarquez la présence du lambda dans l'appel de la fonction lambda : `chcolor('blue')`. Un simple `chcolor('blue')` ne marcherait pas car la fonction à appeler par le menu ne doit contenir aucun paramètre.

Exercice : Ajoutez à ce menu, un 3ème onglet de votre choix ainsi qu'un sous menu associé .

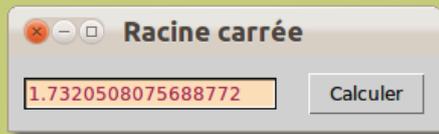
Rendez-vous sur le site http://fsincere.free.fr/isn/python/cours_python_tkinter.php où vous découvrirez de nombreuses et riches applications de cette interface graphique sous Tkinter.

Une autre site intéressant : <http://www.fil.univ-lille1.fr/~marvie/python/chapitre6.html#definition-de-bindings>

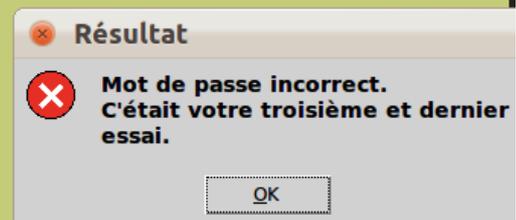
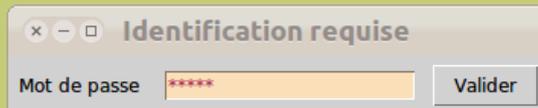
ou encore celui-ci : <http://www.jchr.be/python/tkinter.htm>

Exercice 7.1 ★ En s'inspirant des scripts de `.py` et `mot_de_passe.py`, écrire une application avec interface graphique qui calcule la racine carrée d'un nombre.

Par exemple, le calcul de $\sqrt{3}$ donne :



Exercice 7.2 ★ Reprendre le script `mot_de_passe.py` de manière à limiter le nombre d'essais à trois.



Exercice 7.3 ★ En s'inspirant des scripts de `.py` et `mot_de_passe.py`, écrire une application avec interface graphique qui calcule l'addition ou la soustraction de deux nombres :

