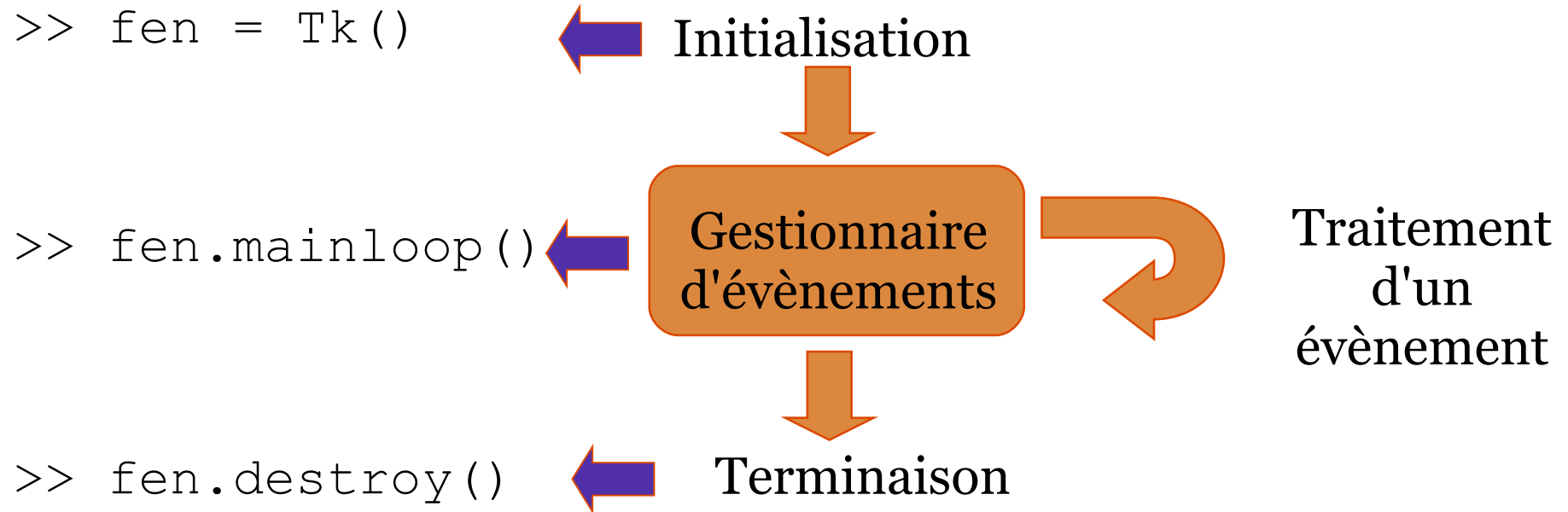


Introduction à TKinter



N. Liebeaux - (c) IUT CACHAN 2008

De nombreux extraits sont tirés du livre de G. Swinnen.



La fenêtre principale (le **parent**) reçoit donc plusieurs widgets (les **enfants**) pour créer une **structure arborescente**.

Qu'est ce que c'est ?

module python de création d'interfaces graphiques (GUI)

```
>> from tkinter import *
```

- ◆ il existe d'autres modules de création d'IHM : pyGTK, pyGTK
- ◆ ces "bibliothèques" définissent des classes d'objet (des familles) de base, dont il faudra étudier les attributs & les méthodes.
- ◆ la documentation TKinter est accessible [ici](#).

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



widget = window + gadget

Button : Un bouton classique, à utiliser pour provoquer l'exécution d'une commande quelconque.

Canvas : Un espace pour disposer divers éléments graphiques. Ce widget peut être utilisé pour dessiner, créer des éditeurs graphiques, et aussi pour implémenter des widgets personnalisés.

Checkbutton : Une « case à cocher » qui peut prendre deux états distincts (la case est cochée ou non). Un clic sur ce widget provoque le changement d'état.

Entry : Un champ d'entrée, dans lequel l'utilisateur du programme pourra insérer un texte quelconque à partir du clavier.

Frame Une surface rectangulaire dans la fenêtre, où l'on peut disposer d'autres widgets. Cette surface peut être colorée. Elle peut aussi être décorée d'une bordure.

Label : Un texte (ou libellé) quelconque (éventuellement une image).

Listbox : Une liste de choix proposés à l'utilisateur, généralement présentés dans une sorte de boîte. On peut également configurer la Listbox de telle manière qu'elle se comporte comme une série de « boutons radio » ou de cases à cocher.

Menu : Un menu. Ce peut être un menu déroulant attaché à la barre de titre, ou bien un menu « pop up » apparaissant n'importe où à la suite d'un clic.

widget = window + gadget

Menubutton : Un bouton-menu, à utiliser pour implémenter des menus déroulants.

Message : Permet d'afficher un texte. Ce widget est une variante du widget Label, qui permet d'adapter automatiquement le texte affiché à une certaine taille ou à un certain rapport largeur/hauteur.

Radiobutton : Représente (par un point noir dans un petit cercle) une des valeurs d'une variable qui peut en posséder plusieurs. Cliquer sur un « bouton radio » donne la valeur correspondante à la variable, et "vide" tous les autres boutons radio associés à la même variable.

Scale : Vous permet de faire varier de manière très visuelle la valeur d'une variable, en déplaçant un curseur le long d'une règle.

Scrollbar : « ascenseur » ou « barre de défilement » que vous pouvez utiliser en association avec les autres widgets : Canvas, Entry, Listbox, Text.

Text : Affichage de texte formaté. Permet aussi à l'utilisateur d'éditer le texte affiché. Des images peuvent également être insérées.

Toplevel : Une fenêtre affichée séparément, « par-dessus ».

Comment bien positionner les widgets sur votre fenêtre ?

- la méthode pack() : la + simple. Peu de liberté pour positionner finement les widgets. (cf. page 96)

```
>> Label(fen, text="ok!").pack()
```

- la méthode grid() : cette méthode considère l'IHM comme une matrice de lignes (row) et de colonnes (column). Pour placer un widget, on indique ces coordonnées dans la matrice

```
>> Label(fen, text="ok!").grid(row = 2, column = 1)
```

- la méthode place() : positionnement absolu au pixel près (attention à la portabilité).

```
>> Label(fen, text="ok!").place()
```

Ces méthodes sont mutuellement exclusives

Comment bien positionner les widgets sur votre fenêtre ?

● la méthode pack() :

```
from Tkinter import *
fen1 = Tk()
txt1 = Label(fen1, text = 'Premier champ :')
txt2 = Label(fen1, text = 'Second :')
entr1 = Entry(fen1)
entr2 = Entry(fen1)
txt1.pack(side =LEFT)
txt2.pack(side =LEFT)
entr1.pack(side =RIGHT)
entr2.pack(side =RIGHT)
fen1.mainloop()
```

● la méthode grid() :

```
from Tkinter import *
fen1 = Tk()
txt1 = Label(fen1, text =
txt2 = Label(fen1, text =
entr1 = Entry(fen1)
entr2 = Entry(fen1)
txt1.grid(row =0)
txt2.grid(row =1)
entr1.grid(row =0, column
entr2.grid(row =1, column
fen1.mainloop()
```

Objectif: créer une IHM pour une calculatrice financière déterminant les mensualités dans le cadre d'un prêt immobilier (exemples : ici ou bien là).



Question : quelle équation évidente relie K , N , M et I ?