

0.1 Tkinter: première approche

Tkinter (de l'anglais Tool kit interface) est la bibliothèque graphique libre d'origine pour le langage Python, permettant la création d'interfaces graphiques. Vous trouverez une documentation complète en suivant ce lien: <http://tkinter.fdex.eu/doc/sa.html> et des exemples bien posés en suivant celui-ci: <https://python.doctor/page-tkinter-interface-graphique-python-tutoriel>

Programme minimal pour afficher une fenêtre:

```
# coding: utf-8

from tkinter import *

fenetre = Tk()

fenetre.mainloop()
```

Les widget Il s'agira d'ajouter dans la fenêtre des composants graphiques appelés *widget* pour windows gadget. Ce sont des objets comme des boutons, des étiquettes, etc.

```
# coding: utf-8

from tkinter import *

fenetre = Tk()
label = Label(fenetre, text="Hello World")
label.pack()
bouton_sortie = Button(fenetre, text="Quitter", command=fenetre.destroy)
bouton_sortie.pack()
fenetre.mainloop()
```

L'appel de la méthode `pack()` assure l'affichage des différents éléments de cette fenêtre. Le constructeur de la classe `Button` retourne le nouveau widget `Bouton_sortie`. Ses options sont `text` et `command` que l'on comprend facilement, vous remarquerez que l'appel à la fonction `destroy` n'est pas associée à des parenthèses. Il faudra consulter la documentation de chaque classe pour les connaître. Le premier paramètre est le **parent** du widget: il faut en effet indiquer qui contient ce widget: ici l'objet instancié `fenetre`.

0.1.1 Les couleurs

Il est possible d'indiquer une valeur de couleur par son nom en anglais: "white", "black", "red", "yellow", etc. ou par son code hexadécimal: #000000, #00FFFF..., etc. Les couleurs sont donc représentée avec **Tkinter comme en HTML** par 3 nombres hexadécimaux représentant les tons de rouge, de vert et de bleu de la couleur choisie: c'est le codage RVB en français ou RGB en anglais. Ainsi la syntaxe de codage d'une couleur est la suivante :

```
couleur = "#RRVVBB"
```

Le type de cette variable est donc `str`: c'est une chaîne de caractère. Utilisez la palette de *Gimp* ou l'un des nombreux sites comme <https://html-color-codes.info/Codes-couleur-HTML/> pour manipuler quelques couleur avec leurs codes RVB et HTML.

Rappelons que #FF en hexadécimal représente le nombre 255 en décimal et que #00 représente 0 en décimal.

A faire

1. Ecrire une instruction permettant de créer une instance de la classe Tk affectée à la variable `fen` ?
2. Expliquer quelle est la différence entre `label` et `Label` dans l'instruction ci-dessous:

```
label = Label(fenetre, text="Hello World")
```

3. Utilisez un `help()` sur la fonction *built-in* `hex()` afin de lire sa documentation puis effectuez quelques tests de cette fonction.

4. Ecrire une fonction `dec2hex()` convertissant un entier compris entre 0 et 255 en sa valeur hexadécimale. Vous pouvez utiliser la fonction `hex()` de python mais attention, cette chaîne de caractère devra comporter *exactement* deux caractères. **Exemple**

```
>>> dec2hex(100)
'64'
>>> dec2hex(8)
'08'
```

5. Ecrire une fonction `rvb2html()` prenant en paramètre 3 entiers compris entre 0 et 255 et retournant le code tkinter/HTML de la couleur correspondant à ce code RVB.

Exemple

```
>>> rvb2html(100, 250, 255)
'#64faff'
```

6. Ecrire un programme affichant des lignes de couleur aléatoire et de position aléatoire. A l'appui sur une touche, l'image est à nouveau générée.

Exemple

```
>>> coul_alea()
```

