Du code Python dans vos fichiers LaTeX

Ce fichier n'est qu'un exemple, il y a surement d'autres possibilités.
Le but étant de conserver l'indentation lors d'un copier/coller.
Hélàs, cela dépend du lecteur pdf. Ici, cela fonctionne avec Adobe Reader.
Les lignes vides seront remplacées par un # lors du copier/coller pour conserver l'espace.

Imporation d'un fichier Python externe :

```
from tkinter import *

def KeyBoard(event):
    dx=0
    dy=0
    Key = event.keysym
    if Key == 'Right':
        dy = -10
    if Key == 'Up':
        dy = 10
    if Key == 'Left':
        dx = 10
    if Key == 'Down':
        dx = -10
    MyCanvas.move(Piece,dx, dy)

#Création de la fenêtre:
MyWindow = Tk()
MyWindow.title('Piece')
#Création du canvas:
MyCanvas = Canvas(MyWindow, width = 480, height =320, bg ='white')
#Création de la balle:
Piece = MyCanvas.create_oval(230,150,250,170,width=2,outline='black',fill='red')
MyCanvas.focus_set()
#Association du clavier à la fonction KeyBoard:
MyCanvas.bind('<Key>',KeyBoard)
#Marges:
MyCanvas.grid(padx =50, pady =50)
#Un bouton quitter:
Button(MyWindow, text ='Exit', command = MyWindow.destroy).grid(padx=5,pady=5)
#Lancement de la boucle principle:
MyWindow.mainloop()
```

Un programme inséré directement dans le source tex :

```
#Fonction ensorcelle :
def ensorcelle(x):
    resultat=-1/x+1
    return resultat

a=int(input("Entrez un nombre : "))
a=ensorcelle(a)
print(a)
```

Du code dans une ligne : `print(3+4)`