

```

# Créé par alexanor, le 30/03/2016 en Python 3.2
#import
from tkinter import*
from random import*

#fonctions
def liste():
    l=list(range(60))
    for i in l:
        l[i]=list(range(60))
        for j in range(60):
            l[i][j]=0
    return l

def aléatoire(nombre): #rend vivantes aléatoirement "nombre" cellules du tableau
    n=0
    while n!=nombre:
        x=randint(0,59)
        y=randint(0,59)
        ini(x,y)
        n=n+1

def go(): #démarrage de l'animation
    global m
    if m==1:
        comptage()
        f.after(500,go)

def stop(): # arrêt de l'animation
    global m
    m=0

def continuer(): # poursuite de l'animation
    global m
    m=1
    go()

def rajouter(event): # poursuite de l'animation
    i=event.x
    j=event.y
    x=i//10
    y=j//10
    ini(x,y)

def ini(x,y): #active la cellule de coordonnée x,y
    l[x][y]=1
    ca.delete(r[x][y])
    r[x][y]=ca.create_rectangle(x*10,y*10,x*10+10,y*10+10,fill='black')

```

```

def détruit(x,y): #détruit la cellule de coordonnée x,y
l[x][y]=0
ca.delete(r[x][y])
r[x][y]=ca.create_rectangle(x*10,y*10,x*10+10,y*10+10,fill='white')

def comptage(): #fonction comptant le nombre de voisins de chaque cellule et remplit ainsi la liste c
global m
compt=0
for i in range(len(l)):
for j in range(len(l[i])):
if i==0 and j==0: #coin en haut à gauche
compt=l[0][1]+l[1][0]+l[1][1]
c[0][0]=compt
elif i==0 and j==59: #coin en haut à droite
compt=l[0][58]+l[1][58]+l[1][59]
c[0][59]=compt
elif i==59 and j==0: #coin en bas à gauche
compt=l[58][0]+l[58][1]+l[59][1]
c[59][0]=compt
elif i==59 and j==59: #coin en bas à droite
compt=l[58][59]+l[58][58]+l[59][58]
c[59][59]=compt
elif 0<i<59 and j==0: #bord de gauche
compt=l[i-1][0]+l[i-1][1]+l[i][1]+l[i+1][1]+l[i+1][0]
c[i][0]=compt
elif 0<i<59 and j==59 : #bord de droite
compt=l[i-1][59]+l[i-1][58]+l[i][58]+l[i+1][58]+l[i+1][59]
c[i][59]=compt
elif i==0 and 0<j<59 : #bord du haut
compt=l[0][j-1]+l[1][j-1]+l[1][j]+l[1][j+1]+l[0][j+1]
c[0][j]=compt
elif i==59 and 0<j<59 : #bord du bas
compt=l[59][j-1]+l[58][j-1]+l[58][j]+l[58][j+1]+l[59][j+1]
c[0][j]=compt
else : #toutes les autres
compt=l[i-1][j-1]+l[i-1][j]+l[i-1][j+1]+l[i][j+1]+l[i+1][j+1]+l[i+1][j]+l[i+1][j-1]+l[i][j-1]
c[i][j]=compt
régle()

def règle(): #fonction redessinant le tableau à partir de la liste c
for i in range(len(l)):
for j in range(len(l[i])):
if l[i][j]==0 and c[i][j]==3:
ini(i,j)
elif (l[i][j]==1 and c[i][j]<2) or (l[i][j]==1 and c[i][j]>3):
détruit(i,j)

#programme principal
m=1
l=liste()

```

```
c=liste()
r=liste()
f=Tk()
ca=Canvas(f,width=600,height=600,background="white")
for i in range(60):
    for j in range(60):
        lo=i*10
        ha=j*10
        r[i][j]=ca.create_rectangle(lo,ha,lo+10,ha+10,outline='black',fill='white')
ca.bind("<Button-1>",rajouter)
ca.pack()
s=Spinbox(f,values=(100,200,300,400,500,600))
s.pack()
bouton1=Button(f,text="générer les cellules vivantes
aléatoirement",command=lambda:aléatoire(int(s.get())))
bouton1.pack()
bouton2=Button(f,text="voir",command=go)
bouton2.pack()
bouton3=Button(f, text ="pause", command =stop)
bouton3.pack()
bouton4=Button(f, text ="continuer", command =continuer)
bouton4.pack()
f.mainloop()
```