

Como_fazer_animação_mes_sem_bloqueio_usando_sprites

ID		Creation date	March 11, 2009
Platform	S60 1st Edition, S60 2nd Edition, S60 3rd Edition	Tested on devices	Nokia N95
Category	Python	Subcategory	

Keywords (APIs, classes, methods, functions): Animação, sprite

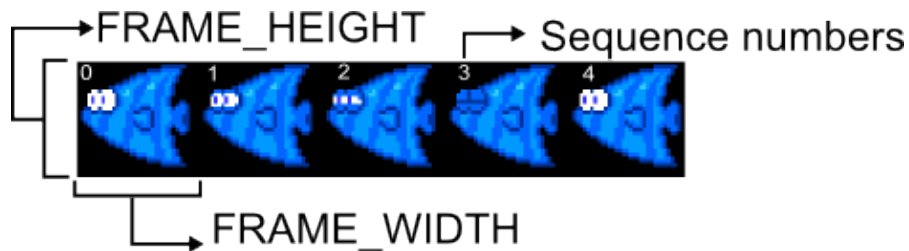
Aqui está um exemplo simples de como fazer animações usando sprites no [PyS60](#). Este exemplo não bloqueia a thread principal porque ele usa a função assíncrona `e32.ao_sleep`.

Nesse exemplo, nós usaremos a imagem sequencial do peixe mostrada abaixo:



Pode-se usar qualquer imagem sequencial nesse exemplo. Basta alterar as constantes `IMAGE_LOCATION`, `FRAME_WIDTH`, `FRAME_HEIGHT`, `SEQUENCE`, `POS_X`, `POS_Y` e `DELAY`

As constantes `FRAME_WIDTH` e `FRAME_HEIGHT` são a largura e a altura de cada quadro da imagem, como mostrado a seguir.



A constante `SEQUENCE` é uma lista de inteiros que determina a sequência de aparecimento de cada quadro como mostrado na imagem acima. `POS_X` e `POS_Y` são a posição em que a animação aparecerá na tela.

```
import appuifw, graphics, e32, sysinfo

IMAGE_LOCATION = "c:\\python\\sprite.GIF"
FRAME_WIDTH = 35
FRAME_HEIGHT = 33
SEQUENCE = [0, 1, 2, 3, 2, 1]
POS_X = 50
POS_Y = 50
DELAY = 0.2

def handle_redraw(rect):
    if img:
        canvas.blit(img)

def quit():
    running = 0
    lock.signal()

lock = e32.Ao_lock() #Cria um lock
```

Como fazer animações sem bloqueio usando sprites

```
display_size = sysinfo.display_pixels() #Obtem o tamanho da tela
appuifw.app.screen = 'full'             #Configura a aplicação para fullscreen
img = graphics.Image.new(display_size)  #Cria uma imagem de base
canvas = appuifw.Canvas(redraw_callback=handle_redraw) #cria um canvas
appuifw.app.body = canvas               #Atribui o canvas como sendo o corpo da aplicação
sprite_image = graphics.Image.open(IMAGE_LOCATION) #Carrega a imagem sequencial
appuifw.app.exit_key_handler = quit     #Configura a função de saída

running = 1
count = 0

def animation():
    global count
    img.clear(0)
    img.blit(sprite_image,
              target = (POS_X, POS_Y),
              source = ((SEQUENCE[count]*FRAME_WIDTH, 0),
                       (FRAME_WIDTH*(1+SEQUENCE[count]), FRAME_HEIGHT)))
    handle_redraw(())

    if count < len(SEQUENCE) - 1:
        count += 1
    else:
        count = 0

    if running:
        #Chama a função de animação depois de DELAY segundos sem bloquear
        e32.ao_sleep(DELAY, animation)

animation()
lock.wait()
```