

```

9  import pygame
10 from pygame.locals import *
11 import random
12
13 nMAX_SPRITES = 10 # numero de naves
14
15 #-----
16 # Carga imagenes y convierte formato pygame
17 #-----
18 def Load_Image(sFile,transp=False):
19     try: image = pygame.image.load(sFile)
20     except pygame.error,message:
21         raise SystemExit,message
22     image = image.convert()
23     if transp:
24         color = image.get_at((0,0))
25         image.set_colorkey(color,RLEACCEL)
26     return image
27
28 #-----
29 # Carga imagenes a array de sprites
30 #-----
31 def Set_Img():      #carga las imagenes, fondo y naves
32     aImg = []
33     aImg.append(Load_Image('bg.jpg' ))
34     aImg.append(Load_Image('n2.png',True ))
35     return aImg
36
37 #-----
38 # Pinta Fondo
39 #-----
40 def Draw_Bg():
41     wm.blit(aFig[0],(0,0))
42     return
43
44 #-----
45 # Main.-
46 #-----
47 pygame.init()
48
49 aSize  = [500,600] ; Black  = [0,0,0] ; aNaves = [] ; lok = True; nSel = 1
50
51 wm = pygame.display.set_mode(aSize)
52 pygame.display.set_caption('Mini Games')
53
54 clock = pygame.time.Clock()
55 aFig = Set_Img()
56 oxNaves = []
57 oyNaves = []
58 # ----- Setea las posiciones iniciales de las naves -----
59 for i in range(nMAX_SPRITES):
60     x = random.randrange(0,440)
61     y = random.randrange(0,aSize[1])
62     aNaves.append([x,y])
63     sentido = random.randint(-1,1)
64     while sentido == 0:      # Sentido a donde se mueve la nave
65         sentido = random.randint(-1,1)
66     oxNaves.append(sentido)
67     sentido = random.randint(-1,1)
68     while sentido == 0:      # Sentido a donde se mueve la nave
69         sentido = random.randint(-1,1)
70     oyNaves.append(sentido)
71
72 while lok:
73     for event in pygame.event.get():
74         if event.type == pygame.QUIT:
75             lok = False
76     Draw_Bg()      #pinta el fondo
77     for i in range(len(aNaves)):      # movimiento de las naves
78         wm.blit(aFig[nSel],aNaves[i])
79         aNaves[i][1] += random.randint(1,2) * oyNaves[i]      # coordenada Y
80         aNaves[i][0] += random.randint(1,2) * oxNaves[i]      # coordenada X
81
82         if aNaves[i][1] > aSize[1]:      # si sale de la pantalla
83             aNaves[i][1] = -55

```

```
84         if aNaves[i][0] > aSize[0]:           # si sale de la pantalla
85             aNaves[i][0] = -60
86
87         if aNaves[i][1] < -55:                 # si sale de la pantalla
88             aNaves[i][1] = aSize[1]
89         if aNaves[i][0] < -60:                 # si sale de la pantalla
90             aNaves[i][0] = aSize[0]
91
92
93     pygame.display.flip()    #actualiza la pantalla
94     clock.tick(100)
95     pygame.quit ()
```