

5.- PROCESSING.PY: IMAXES,FONTES E MEDIA

Os ficheiros multimedia vanse gardar nun cartafol chamado *data* ao carón do ficheiro do programa. Cando engadimos o ficheiro, coa opción Sketch: *Add File...*, vaise crear automaticamente. Tamén podemos crealo manualmente. Coa opción *Ctrl+k* ou *Sketch: Show Sketch Folder...* podemos verificar se está o cartafol *data* e os ficheiros que contén.

loadImage(): Para cargar unha imaxe haberá sempre 3 pasos:

1. Engadir a imaxe ao ficheiro *data*
2. Crear unha variable baleira (nula, co valor *None*) para almacenar a imaxe
3. Cargar a imaxe na variable coa función *loadImage()*

image("imaxe", x,y): función para visualizar unha imaxe na pantalla na posición (x,y). O nome da imaxe en que ser exacto, incluíndo maiúsculas, e a extensión do ficheiro. Opcionalmente poderán engadirse os parámetro de anchura e altura (4º e 5º parámetro). Para cargar máis imaxes as engadimos e creamos máis variables.

```
1 img = None
2
3 def setup():
4     global img
5     size(480,480)
6     img = loadImage('lunar.jpg')
7
8 def draw():
9     image(img,0,0)
```

```
traballo con imaxes 1
1 img1 = None
2 img2 = None
3
4 def setup():
5     global img1,img2
6     size(480,120)
7     img1 = loadImage('lunar.jpg')
8     img2 = loadImage('capsule.jpg')
9
10 def draw():
11     image(img1,-120,0)
12     image(img1,130,0,240,120)
13     image(img2,300,0,240,120)
```



Movement das imaxes pode facerse por exemplo coas variables *mouseX* e *mouseY*.

```
1 img = None
2
3 def setup():
4     global img
5     size(480,420)
6     img = loadImage('lunar.jpg')
7
8 def draw():
9     background(0)
10    image(img,0,0,mouseX*2,mouseY*2)
```

Processing pode usar imaxes .jpg, .png e .gif., e por suposto .svg. Neste exemplo podes ver a diferenza de transparencia entre unha imaxe GIF (1 bit de transparencia) e unha PNG (8 bits).

```
1 img = None
2
3 def setup():
4     global img
5     size(480,420)
6     img = loadImage('clouds.gif')
7
8 def draw():
9     background(255)
10    image(img,0,0)
11    image(img,0,mouseY*-1)
```

```
1 img = None
2
3 def setup():
4     global img
5     size(480,420)
6     img = loadImage('clouds.png')
7
8 def draw():
9     background(255)
10    image(img,0,0)
11    image(img,0,mouseY*-1)
```

FONTES TIPOGRÁFICAS

Processing pode traballar con ficheiros de tipografía .ttf e .otf. É similar ao das imaxes:

1. Engadir unha fonte ao cartafol data.
2. Crear unha variable baleira.
3. Crear a fonte e engadila á variable coa función `createFont()`, indicando o tamaño.
4. Usar o comando `textFont()` para establecer a fonte a utilizar.

text(frase,x,y): comando que debuxa unha frase comenzando nas coordenadas (x,y)

Se engadimos un 4º e 5º parámetro debuxará a frase nunha caixa dese tamaño

```
1 fonte = None
2
3 def setup():
4     global fonte
5     size(480,420)
6     fonte = createFont("galega.otf",32)
7     textFont(fonte)
8
9 def draw():
10    background(102)
11    textSize(32)
12    text('adeus rios, adeus fontes',25,60)
13    textSize(16)
14    text('adeus rios, adeus fontes',27,90)
```

O normal é ter as frases gardadas en variables de tipo string (str), pois iso permite modificalas, editalas...

```
1 fonte = None
2 frase = 'Nunha galaxia moi moi lonxana..'
3
4 def setup():
5     global fonte
6     size(480,420)
7     fonte = createFont("starwars.ttf",24)
8     textFont(fonte)
9
10 def draw():
11    background(102)
12    text(frase,26,24,240,100)
```

Exercicio: crea un chiste (de cando menos 3 frases) que apareza en pantalla cada vez que pulses unha tecla ou cada vez que premas no botón (pista: Usa condicionais)

FORMAS (ficheiros vectoriais SVG)

Se queres traballar con ficheiros SVG é similar ás imaxe pixeladas.

1. Engade o ficheiro SVG no cartafol data
2. Crea unha variable baleira
3. Carga o ficheiro na variable coa función `loadShape()` e usa o comando `shape()`

```
1 forma = None
2
3 def setup():
4     global forma
5     size(480,420)
6     forma = loadShape('network.svg')
7
8 def draw():
9     background(0)
10    shape(forma,30,10)
11    shape(forma,180,10,280,280)
```

Unha das grandes ventaxas dos ficheiros vectoriais SVG é poder escalalo sen perder resolución.

```
1 forma = None
2
3 def setup():
4     global forma
5     size(240,120)
6     shapeMode(CENTER) #para centrar o debuxo
7     forma = loadShape('network.svg')
8
9 def draw():
10    background(0)
11    diametro = map(mouseX,0,width,10,800)
12    shape(forma,120,60,diametro,diametro)
```

Para crear unha nova forma usamos a función `createShape()` dentro de `setup()` e engadimos o contido nunha variable. **Fíxate como engadimos os vértices na variable.** As funcións `beginShape()` e `endShape()` van en parella.

```
1 dino = None
2 def setup():
3     global dino
4     size(480,120)
5     dino = createShape()
6     dino.beginShape()
7     dino.fill(153,176,180)
8     dino.vertex(50,120)
9     dino.vertex(100,90)
10    dino.vertex(110,60)
11    dino.vertex(80,20)
12    dino.vertex(210,60)
13    dino.vertex(160,80)
14    dino.vertex(200,90)
15    dino.vertex(140,100)
16    dino.vertex(130,120)
17    dino.endShape()
18 def draw():
19    background(204)
20    translate(mouseX-120,0)
21    shape(dino,0,0)
```