

Operadores lógicos

Co operador AND comparamos varias condicións que teñen que ser todas certas para almacenar o valor TRUE. Co operador OR chega que unha sexa certa.

É dicir:

TRUE AND TRUE = TRUE TRUE AND FALSE = FALSE FALSE AND FALSE = FALSE
TRUE OR TRUE = TRUE TRUE OR FALSE = TRUE FALSE OR FALSE = FALSE

Exercicio: anticipa o funcionamento do código.

```
1 x = 80
2 y = 30
3 w = 80
4 h = 60
5
6 def setup():
7     size(240,120)
8 def draw():
9     background(204)
10    if mouseX>x and mouseX<x+w and mouseY>y and mouseY<y+h:
11        fill(0)#todas as condición teñen que ser TRUE
12    else:
13        fill(255) #chega con unha condición FALSE
14    rect(x,y,w,h)
```

Teclas pulsadas ou debuxadas

keyPressed: equivalente a mousePressed, almacena true se unha tecla está presionada.

key: almacena a tecla pulsada. Nas teclas especiais: key == CODED

text(): función que debuxa as teclas indicadas. Hai teclas especiais que non debuxan nada (Ctrl, Alt...).

Para debuxar texto usaremos funcións como textSize() e textAlign().

```
1 def setup():
2     size(240,120)
3 def draw():
4     background(204)
5     line(20,20,220,100)
6     if keyPressed:
7         line(220,20,20,100)
```

```
1 def setup():
2     size(240,120)
3     textSize(64)
4     textAlign(CENTER)
5 def draw():
6     background(0)
7     if keyPressed:
8         text(key,60,80)
```

Co condicional if podemos saber se se pulsa unha tecla determinada e modificar algo en función diso.

Exercicio: modifica o código para que debuxe outra letra máis ao premela.

```
1 def setup():
2     size(240,120)
3
4 def draw():
5     background(204)
6     if keyPressed:
7         if key == 'h' or key=='H':
8             line(30,60,90,60)
9         if key == 'n' or key=='N':
10            line(30,20,90,100)
11            line(30,20,30,100)
12            line(90,20,90,100)
```

Moverse con flechas (ou teclas)

keyCode: variable que permite identificar as tecla especiais (CTRL, ALT, SHIFT, UP...).

```
1 x = 215
2 def setup():
3     size(480,120)
4
5 def draw():
6     global x
7     if keyPressed and key == CODED:
8         if keyCode == LEFT:
9             x -= 1 #tecla esquerda diminúe o valor de x
10            elif keyCode == RIGHT: #elif significa else if, un segundo condicional
11                x += 1 #tecla dereita aumenta o valor de x
12            rect(x,45,50,50)
```

Neste exemplo usamos un condicional, if, e logo un segundo condicional, elif (else if). Non é o mesmo que escribir if e logo else.

Exercicio: intenta que o rectángulo non deixe rastro.

Un problema moi habitual é **transformar valores dun rango noutro**. Por exemplo, se dexesamos asignar cores distintas ao rango 0-800 píxeles de ancho (para ter cores distintas segundo movamos o rato pola pantalla en horizontal) os valores das cores, para ser útiles, estarán no rango 0-255. Para facer este cambio podemos utilizar matemáticas (alxebra) ou algunha función que fai este cálculo automaticamente.

map(): función que transforma un valor dentro dun rango a outro valor dentro doutro rango. Exemplo anterior: map(WIDTH, 0, 800, 0, 255) asignaría un valor ao ancho entre 0-255 en vez de 0-800.

Exercicio: Execuíta o programa e compara o modo alxebraico (o primeiro código) e o modo con función (o segundo).

```
1 def setup():
2     size(240,120)
3     strokeWeight(12)
4
5 def draw():
6     background(204)
7     stroke(102)
8     line(mouseX,0,mouseX,height)#liña gris
9     stroke(0)
10    mx = mouseX/2+60 #definimos unha función variable
11    line(mx,0,mx,height) #liña negra que varia con mx
```

```
1 def setup():
2     size(240,120)
3     strokeWeight(12)
4
5 def draw():
6     background(204)
7     stroke(102)
8     line(mouseX,0,mouseX,height)#liña gris
9     stroke(0)
10    mx = map(mouseX,0,width,60,180)
11    #función de processing, cambia os valores posibles de mx ao rango 60-180
12    line(mx,0,mx,height) #liña negra que varia con mx
```