

PROCESSING.PY: VARIABLES E BUCLE FOR

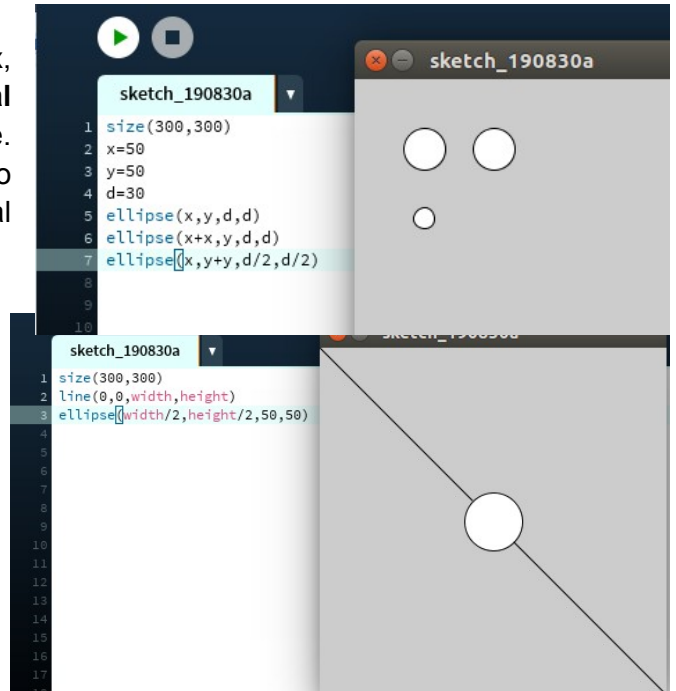
<https://py.processing.org/tutorials/overview/>

As variables se definen coma en Python, asignando un valor a un nome ($x = 50$). Logo poden utilizarse, e modificar o valor (suma, resta, asignación...). Ver operadores e métodos abreviados en [Reference Processing](#).

MOI IMPORTANTE:

se escribes $x = x + 50$ modifica o valor de x , isto non ten sentido matemático. **O uso de igual é para asignar**, introducir un valor na variable. Cando digamos $=$ en sentido matemático usaremos $==$ (por exemplo nun condicional comparativo: `if x == 3, then print(x)`)

Haberá variables especiais, nomes asignados. As máis importante ao principio son **width e height** que se definen ao asignar valores en `size()`. Con eles podemos debuxar obxectos en sitios concretos aínda que logo modifiquemos o tamaño da ventá.



BUCLE FOR (ITERACIÓN)

O bucle for é o máis utilizado en programación textual, xunto a while. A súa sintaxe é:

for variable_contadora in range (start, stop, step):

comandos a executar en bucle (isto vai tabulado, indentado)

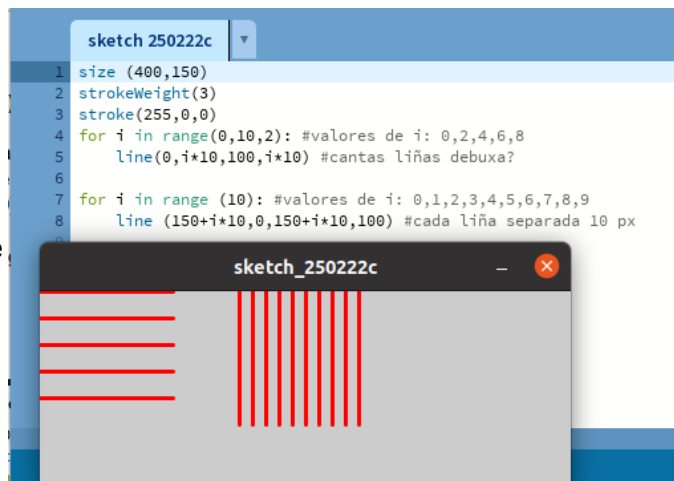
A **variable contadora** de uso habitual é i (e logo, j, k, \dots). O seu valor varía dende **start** ata **stop** en pasos (**step**). Por exemplo en **for i in range (0,10,2)** o valor de i vale 0,2,4,6,8 pero (IMPORTANTE) non chega a valer 10. Se escribimos **for i in range (10)**, i vale dende 0 a 9.

Imos usar o bucle for para debuxar en bucle. Primeiro poñemos o grosor (`strokeWeight`) e a cor (`stroke(R,G,B)`), o fondo (`background()`), o recheo (`fill()`),...

No primeiro bucle debuxamos liñas horizontais separadas unha distancia de $y = i * 10$: 0,20,40,60,80 por que i toma os valores de 0,2,4,6,8 no bucle.

No segundo bucle liñas verticais separadas 10 píxeles, sendo os valores de x : 150,160...ata chegar a 240, pois os valores de i van de 0 a 9.

FAI OS EXERCICIOS DE BUCLE FOR QUE ESTÁN NA AULA VIRTUAL



EXERCICIOS BUCLE FOR

1.- Debuxa as liñas antes de executalas:

```
sketch_04loops_bucle_for
1 size(480,120)
2 strokeWeight(8)
3 for i in range(20,400,60):
4     line(i,40,i+60,80)
5
```

2.- Loop aplicado 2 veces (debuxar antes):

```
sketch_05loopaplicado2veces
1 size(480,120)
2 strokeWeight(2)
3 for i in range(20,400,20):
4     line(i,0,i+i/2,80)
5     line(i+i/2,80,i*1.2,120)
6
```

3.- Loop duplicado(debuxar antes):

```
sketch_06loopduplicado
1 size(480,120)
2 background(0)
3 noStroke()
4
5 for y in range(0,height+45,40):
6     for x in range(0,width+45,40):
7         fill(255,140)
8         ellipse(x,y,40,40)
```

4.- Dous loops:

```
sketch_07loops2veces
1 size(480,120)
2 background(0)
3 noStroke()
4
5 for y in range(0,height+45,40):
6     fill(255,140)
7     ellipse(0,y,40,40)
8
9 for x in range(0,width+45,40):
10    fill(255,140)
11    ellipse(x,0,40,40)
```

5.- Escribe i executa:

```
sketch_08loopduplicado_pinsandlines_
1 size(480,120)
2 background(0)
3 fill(255)
4 stroke(102)
5 for y in range(20,height-15,10):
6     for x in range(20,width-15,10):
7         ellipse(x,y,4,4)
8         line(x,y,240,60) #liña ata o centro da pantalla
```

6.- Escribe e executa:

```
sketch_09loopsduplicados_halfone_dots
1 size(480,120)
2 background(0)
3 for y in range(32,height,8):
4     for x in range(12,width,15):
5         ellipse(x+y,y,16-y/10.0,16-y/10.0)
```