

Introducción a ciclos anidados

Introducción a ciclos anidados	1
¿Qué aprenderás?	2
Introducción	2
Escribiendo las tablas de multiplicar	3
Tabla de un número	3
Tabla de todos los números	3
Complejidad de un programa	4



¡Comencemos!

¿Qué aprenderás?

- Utilizar ciclos anidados para resolver problemas.
- Conocer el concepto de complejidad

Introducción

Un ciclo anidado es simplemente un ciclo dentro de otro ciclo. No existe ningún límite respecto a cuántos ciclos pueden haber anidados dentro de un código, aunque por cada uno aumentará la **complejidad temporal del programa**. Los ciclos anidados expanden el universo de los tipos de problemas que podemos resolver.

¡Vamos con todo!



Escribiendo las tablas de multiplicar

Tabla de un número

Supongamos que queremos mostrar una tabla de multiplicar. Por ejemplo la tabla del número 5:

```
10.times do |i|  
  puts "5 * #{i} = #{5 * i}"  
end
```

Tabla de todos los números

¿Cómo podríamos hacer para mostrar todas las tablas de multiplicar del 1 al 10?

Fácil, envolviendo el código anterior en otro ciclo que itere de 1 a 10.

```
10.times do |i|  
  10.times do |j|  
    puts "#{i} * #{j} = #{i * j}"  
  end  
end
```

```
0 * 0 = 0  
0 * 1 = 0  
0 * 2 = 0  
0 * 3 = 0  
0 * 4 = 0  
0 * 5 = 0  
0 * 6 = 0  
0 * 7 = 0  
0 * 8 = 0  
0 * 9 = 0  
1 * 0 = 0  
1 * 1 = 1  
...  
9 * 6 = 54  
9 * 7 = 63  
9 * 8 = 72
```

$$9 * 9 = 81$$

Complejidad de un programa

Los ciclos anidados son una buena herramienta cuando es necesario usarlos, pero hay que tener cuidado con exagerar.

Cuando utilizamos un ciclo para recorrer un conjunto de datos, la cantidad de iteraciones depende directamente de la cantidad de datos que se recorren.

Cuando se ocupan ciclos anidados para revisar conjuntos de datos, lo que está dentro de ambos ciclos se ejecutará una cantidad de veces mucho mayor. Si es posible hacer la misma tarea usando un ciclo después de otro, o incluso un solo ciclo, el tiempo que se demora el programa será considerablemente menor. Esto es más notorio para conjuntos más grandes de datos.

Por ejemplo, si tenemos un conjunto de 200 datos y otro de 50, y hacemos ciclos anidados recorriendo ambos, tendríamos ¡10.000 iteraciones! Por otro lado, si dejamos un ciclo después de otro, tendremos sólo 250 iteraciones en total.