

Estructura for

Aquí vemos un ejemplo sencillo de un bucle del tipo for que hace lo mismo. Vamos a analizar cómo funciona. Primero vemos que aparece una variable que se llama `i`. Es un contador, para saber cuántas "vueltas" (iteraciones) llevamos. `range(1,5)` me indica los valores que va a ir tomando la variable `i`. Los dos puntos dan paso al bloque de código "interior" que es lo que se va a repetir cinco veces. Sacar por pantalla el mensaje y añadir un elemento a la lista. Así que `i` empieza en uno y termina en cuatro. Actividad 1: ejecuta el código y pon una captura del resultado

Rango

Aunque es habitual que el contador de un bucle vaya desde cero, de uno en uno, hasta un máximo, no es obligatorio. La función `range` nos devuelve una lista de números:

- `range(5)` nos devuelve 0,1,2,3,4
- `range(3,7)` devuelve 3,4,5,6
- `range(2,15,3)` el contador va desde 2 hasta 13, saltando de tres en tres. Es decir, nos da los números 2,5,8,11 y 14
- `range(14,2,-3)` el contador va decreciendo de 3 en 3. Es decir, nos da los números 14,11,8,5 y 2, en ese orden

Números pares

Ahora vamos a hacer un programa que muestre todos los números pares entre 1 y 100. Los números pares son divisibles por 2. Es decir, dan 0 de residuo al dividir. Aprovecharemos la función `%` que da el resto de una división. Es decir: Dividir 8 entre 3 da cociente 2 y de resto 2. Por tanto, $8\%3 = 2$. Dividir 6 entre 3 da cociente 2 y resto 0. Por tanto, $6\%3 = 0$. 8 no es divisible entre 3, pero 6 sí lo es entre 2.

Hacemos correr el contador entre 1 y 101 (ya sabes que se para justo antes, por eso para llegar al 100 tenemos que poner 101). `i`. Recuerda que para comparar usamos el doble `==`. Si se cumple, lo escribe, si no nada, y pasaría a la iteración siguiente, porque no hay más instrucciones. Corto, sencillo y rápido. Para eso queremos los bucles. Actividad 2: modifica el código anterior para muestre los múltiplos de 5. Adjunta una captura

Calcular potencias

Hagamos otro ejemplo, un programa que calcula potencias. Sería algo así como, me dices que la base vale 2, que el exponente vale 5 y te devuelvo un treinta y dos. Para hacerlo, tengo que ir multiplicando dos por dos, me sale cuatro, luego otra vez por dos... hasta hacerlo cinco veces. ¿Cuántas variables necesito? Necesito tener guardada la base, para ir multiplicándola una y otra vez. Necesito tener guardado el exponente para saber cuándo he hecho el número de iteraciones suficiente y parar. Necesito OTRA variable para ir guardando los números intermedios que me van saliendo. 4, 8, 16 y finalmente 32. A esta variable la voy a llamar `acumulado`. Primera prueba

Pongo base 2, exponente 5 y me sale... 64. Algo pasa, vamos a ver. Si `acumulado` lo inicio con el valor de base, en la primera iteración, ya tengo dos por dos, ya tengo cuatro. Así que en la primera iteración ya tengo dos al CUADRADO. Por lo tanto, tengo que hacer una iteración menos. Probemos. Ejemplo arreglado

Perfecto, así funciona. A veces cuesta ver a la primera donde tiene que empezar un contador o hasta donde tiene que ir, pero si probáis con lo que os parece resulta fácil ajustarlo.

Actividad 3: modifica el programa anterior para que dé la siguiente salida. Adjunta una captura

Contador de letras en una palabra Aunque es muy usual numerar las iteraciones, también podemos ir recorriendo con nuestro "contador" cualquier cosa contable... por ejemplo, una palabra. Contador de letras "a"

Funcionamiento del programa El "contador" i va recorriendo cada carácter de la palabra, y toma como valor ese carácter. Si ponemos "Dani". En la primera iteración i toma el valor 'D', se pasa al condicional, como no es igual a 'a' pues no hacemos nada. En la segunda iteración i toma el valor 'a', se pasa al condicional, como sí es igual a 'a' pues el acumulado se incrementa en una unidad. Etcétera. Finalmente imprimimos el valor acumulado y listo. Actividad 4: modifica el programa anterior para que, diciendo tu nombre y apellidos te diga cuantas a, e, i, o, u. La salida por pantalla debe ser la siguiente. Adjunta una captura

Actividad 5: modifica el programa anterior para que te diga cuantas vocales en total tiene la palabra. Adjunta una captura