



Práctica 14: Percorrendo Listas

O bucle "Para cada elemento" e o procesamento de datos
Módulo 4: Datos Compostos

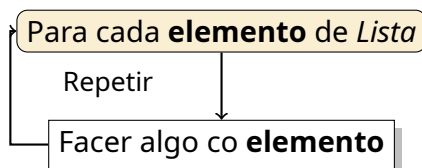
Como ler todo o armario?

Imaxina que tes unha lista de 100 prezos e queres saber canto suman todos. Non vas sumar elemento 1 + elemento 2... a man! Para iso usamos os bucles. Percorrer unha lista significa visitar un a un todos os seus caixóns automaticamente.

Conceptos Clave

1. O bloque "Para cada elemento" (For Each)

Este é o bloque máis potente para traballar con listas en Snap!. O programa encárgase de saber cantos elementos hai e cando chega ao final:



- **A Variable "Elemento":** É unha variable "temporal". No primeiro xiro vale o contido do caixón 1, no segundo o do caixón 2, e así sucesivamente.

2. Esquema Visual dun Percorrido

Imaxina unha lista de froitas: [mazá, pera, plátano].

Xiro (Iteración)	Valor de "elemento"	Acción do programa
1º	mazá	Di "mazá" por 2 seg.
2º	pera	Di "pera" por 2 seg.
3º	plátano	Di "plátano" por 2 seg.

O RETO: O Pase de Lista

Imos programar un personaxe que actúe coma un profesor pasando lista. Debe saudar a cada alumno que estea gardado na nosa lista.

Instrucións en Snap!:

1. Crea unha lista chamada Clase e engade 4 ou 5 nomes (ex: Ana, Lois, Marta, Brais).
2. Usa o bloque `para cada (item) de (Clase)`.
3. Dentro do bucle, o personaxe debe `dicir` un saúdo.
4. **Truco:** Usa o bloque `unir [Hola] (item)` para que o saúdo inclúa o nome do alumno.

Pseudocódigo: O algoritmo de sumar

```
Algoritmo SumarPrezos
  Lista Prezos <- [10, 5, 20]
  Variable Total <- 0

  Para Cada p en Precios:
    Total <- Total + p // Acumulamos o valor de cada caixón

  Escribir "A conta total é: ", Total
FinAlgoritmo
```

Ampliación (Para os máis rápidos)

Serías capaz de facer que o personaxe só saúda aos alumnos cuxo nome empece pola letra "A"?
Pista: Precisarás un bloque "Se..." (Control) dentro do bucle.

Este documento publícase baixo licenza
 Creative Commons Atribución 4.0 Internacional (CC BY 4.0)