

# Práctico N3: Ciclos

## Programación imperativa en JavaScript

Antes de comenzar, deberás crear una carpeta con el nombre **practico3** en tu repositorio de github. Dentro de la misma deberás ir agregando los ejercicios de este práctico a medida que los vayas resolviendo. Al finalizar todos los ejercicios deberás crear un tag con el nombre de **practico3** para realizar la entrega del mismo.

Los ejercicios EXTRA son opcionales, no restarán puntos a la nota final pero sí podrán sumar en caso de que sus resoluciones sean correctas. No es necesario realizar dicho ejercicios, pero recomendamos que si tienen tiempo, al menos lo intenten.

### CUENTA REGRESIVA

En este ejercicio, queremos que imprima una cuenta regresiva de lanzamiento de un cohete a Marte. Dicha cuenta irá desde 10 hasta el despegue. Específicamente, queremos que:

- Cree un bucle de 10 a 0. Ayuda: inicializador: **let i = 10;**
- Para cada iteración, imprima por consola usando **console.log**.
- Diferentes números de iteración requieren que se coloque un texto diferente en el párrafo para esa iteración (necesitará una declaración condicional):
  - Si el número es 10, imprima **"Cuenta regresiva 10"** en el párrafo.
  - Si el número es 0, imprime **"Lanzamiento!"**, al párrafo.
  - Para cualquier otro número, imprima solo el número al párrafo.
- ¡Recuerda incluir un *iterador*! Sin embargo, en este ejemplo estamos contando hacia atrás después de cada iteración, no hacia arriba, por lo que **\*\*no\*\*** quiere **i++** — ¿cómo se itera hacia abajo?
- Debes guardar este programa en un archivo llamado **lanzamiento.js**

## LISTA DE INVITADOS

En este ejercicio, que guardaremos en un archivo `listaDeInvitados.js`, queremos que tomes una lista de nombres almacenados en un arreglo y los coloques en una lista de invitados. Pero no es tan fácil: no queremos dejar entrar a **Jose** y **Sofia** porque son codiciosos y groseros, ¡y siempre se comen toda la comida! Tenemos dos listas, una para admitir invitados y otra para rechazar invitados.

La lista de personas es la siguiente: Jose, Maria, Juan, Andres, Lionel, Sofia, Leandro y Emilia.

Específicamente, queremos que:

- Escribas un bucle que itere desde 0 hasta la longitud del arreglo `personas`. Deberá comenzar con un inicializador de `let i = 0`; pero ¿qué condición de salida necesita?
- Durante cada iteración del ciclo, verifique si el elemento del arreglo actual es igual a **"Jose"** o **"Sofia"** usando una declaración condicional:
  - Si es así, concatene el elemento al arreglo de **rechazados**.
  - Si no es así, concatene el elemento al arreglo de **admitidos**.
- Finalmente, queremos que escribas dos bucles más; uno para imprimir la lista de invitados admitidos y otra para imprimir la lista de invitados rechazados:
  - a. Para la lista de invitados admitidos deberás imprimir por consola un mensaje inicial que dirá: *"La lista de invitados admitidos es:"*, seguido por tantos párrafos como tenga el arreglo **admitidos** conteniendo los respectivos nombres de los invitados.
  - b. Para la lista de invitados rechazados deberás imprimir por consola un mensaje inicial que dirá: *"La lista de invitados rechazados es:"*, seguido por tantos párrafos como tenga el arreglo **rechazados** conteniendo los respectivos nombres de los invitados.
- **EXTRA:** Después de completar con éxito las tareas anteriores, te quedarán dos listas de nombres, los **admitidos** y los **rechazados** pero estarán desordenados. El desafío es que puedas repetir el inciso anterior, pero esta vez imprimiendo los nombres de manera ordenada alfabéticamente. Deberás agregar la palabra "ordenada" posterior a la palabra lista, es decir, para el caso de los admitidos deberás imprimir: *"La lista ordenada de invitados admitidos es:"*.