****

# Examen final de Programación Imperativa

¡Llegó el momento de poner a prueba todo lo que estuvimos viendo a lo largo de estas semanas!

# Metodología de evaluación

Se evaluarán los siguientes conceptos sobre el código entregado:

* **FORMA**
  + Que el código esté prolijo e implemente buenas prácticas
  + Que las variables, métodos y funciones tengan nombres descriptivos
  + Que utilices nombres en español o en inglés pero no ambos
  + Que utilices camelCase donde corresponda
* **LÓGICA**
  + Que la lógica corresponda con lo que solicitan las consignas
  + Que utilices los métodos más adecuados para cada caso
* **FUNCIONAMIENTO**
  + Que el código funcione correctamente, sin arrojar errores
  + Que el código produzca el resultado esperado a partir de los datos suministrados

# Duración, formato y entrega

El examen tendrá una duración de 100 minutos. La hora de finalización será informada por el docente. ⚠️**Las entregas realizadas después del tiempo estipulado no serán tenidas en cuenta**.  
  
Al terminar el examen, deben entregarlo haciendo uso del formulario que les enviará el docente por chat de zoom.

⚠️**Recuerden verificar la recepción del formulario con el/la docente antes de retirarse del zoom**⚠️

# Algunos consejos 🤓👌

* Es normal tener nervios en esta etapa. Recordá que practicamos mucho estas semanas y tratá de mantener la mayor calma que puedas.
* Lee todas las consignas antes de empezar el examen para tener una idea general de lo que se pide.
* Si alguna consigna no queda clara, recordá que podés pedir aclaración por privado al docente.
* Si una consigna te bloquea demasiado, pasá a la siguiente, lo más probable es que luego se te ocurra una solución.

# Ejercicio 1

Dado el siguiente arreglo de objetos:

let cuadros = [

{

nombre : "Mona Lisa",

creador : "Leonardo Da Vinci",

creacion : 1503

},

{

nombre : "La ultima cena",

creador : "Leonardo Da Vinci",

creacion : 1495

},

{

nombre : "La noche estrellada",

creador : "Vincent van Gogh",

creacion : 1889

},

{

nombre : "El grito",

creador : "Edvard Munch",

creacion : 1893

},

{

nombre : "Trigal con cuervos",

creador : "Vincent van Gogh",

creacion : 1890

},

{

nombre : "Maria Magdalena",

creador : "Leonardo Da Vinci",

creacion : 1495

}

]

Realizar una función que reciba por parámetro dicho arreglo y **retorne** un **nuevo** array con los objetos cuyo año de creación sea **mayor** a 1800.

# Ejercicio 2

Realizar una función que reciba por parámetro el arreglo de objetos del ejercicio **1** y un **String**. Esta función deberá ordenar los objetos mediante el algoritmo **Bubble Sort** según su año de **creación** de forma ascendente o descendente según lo indique el string recibido por parámetro, teniendo en cuenta los siguientes criterios:

* Si el String es “ASC” o “asc” encontes ordenar de **menor** a **mayor**
* Si el String es “DESC” o “desc” entonces ordenar de **mayor** a **menor**

# Ejercicio 3

Dada la siguiente matriz:

let matriz = [

[1,2,3,4,5],

[6,7,8,9,10],

[11,12,13,14,15]

]

1. Desarrollar una función que **retorne** la **suma** de todos los valores de la **fila 2.**
2. Desarrollar una función que **retorne** un **nuevo array** que contenga **solo** los elementos **pares** que se encuentren dentro de la matriz.

***IMPORTANTE:*** *Realiza el correspondiente llamado a cada una de las funciones para demostrar el correcto funcionamiento de las mismas.*