SISTEMAS NUMÉRICOS: Introducción a la Informática

UTP | Pereira

simón morales torres

NOVIEMBRE DE 2020

2020

# CONTENIDO

[1 CONTENIDO 1](#_Toc56081332)

[2 PRESENTACIÓN 2](#_Toc56081333)

[3 CONVERSIÓN BASADA EN DIVISIONES SUCESIVAS 6](#_Toc56081334)

[4 CONVERSIÓN EXTENDIDA 8](#_Toc56081335)

[5 CONVERSIÓN: 0, 1 A PALABRAS: CERO, UNO 11](#_Toc56081336)

[6 CONTAR NÚMERO DE UNOS EN UN BINARIO 13](#_Toc56081337)

[7 ARRAYS EN JAVASCRIPT 16](#_Toc56081338)

[8 CONCLUSIONES 45](#_Toc56081339)

[9 BIBLIOGRAFÍA 46](#_Toc56081340)

**AUTOR: <Simón Morales Torres >**

**<CÓDIGO 1004700556>**

**<CORREO simon.morales@utp.edu.co>**

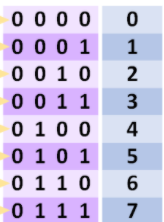
**<GITHUB SimonMorales36>**

# PRESENTACIÓN

La presente monografía describe la implementación de un conjunto de programas que le dan soporte a la teoría numérica básica de la materia INTRODUCCIÓN A LA INFORMÁTICA.

En los siguientes párrafos se presenta una descripción básica del significado de lo que es un sistema numérico, especialmente el sistema en base 2.

Un sistemas numérico es una forma de representar valores numéricos de una manera en que puedan ser contados e interpretados en otros sistemas , como por ejemplo, el sistema binario, donde las cantidades se representan por la unión de unos y ceros, como en el siguiente ejemplo:

 (Imagen 1)

Donde en el lado derecho tenemos valores del sistema decimal, que usamos usualmente, y en el lado izquierdo tenemos lo que representa en el sistema numérico.

El desarrollo de estas actividades está realizado en JavaScript, en la plataforma de repl.it o en github, con esto nos damos cuenta la funcionalidad de los programas en tiempo real con el código.

Cada uno de los programas tiene una funcionalidad diferente, pero tienen el mismo objetivo, mostrar como por medio de la programación podemos ilustrar o poner en contexto grafico las soluciones a los problemas planteados como en el siguiente ejemplo, donde vemos cómo funcionan las compuertas AND, OR Y NOT

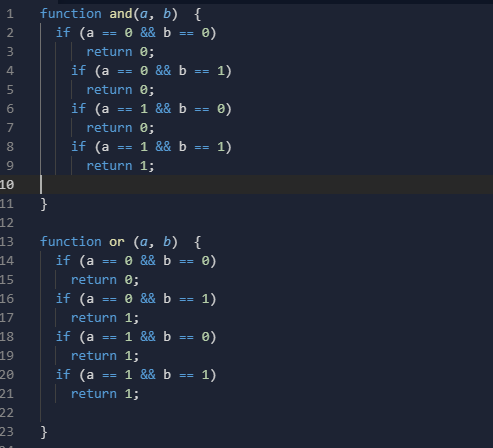


Imagen 3.

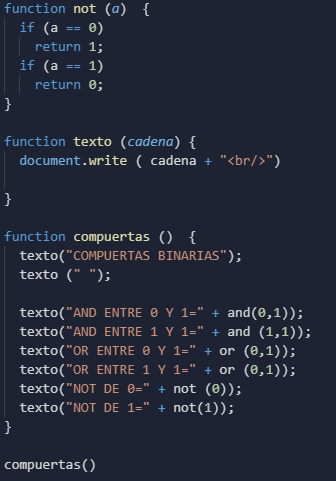
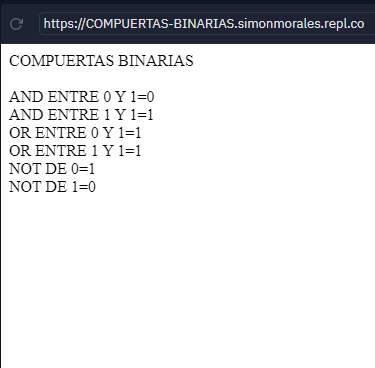


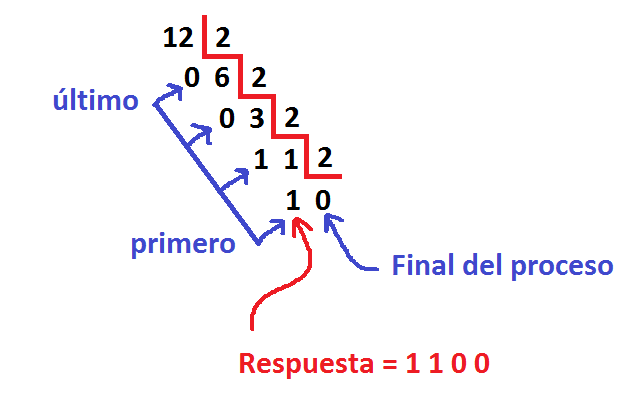
Imagen 3.



Este es el resultado al ejecutar el programa, obtenemos las combinaciones de cada una de las compuertas binarias.

# CONVERSIÓN BASADA EN DIVISIONES SUCESIVAS

A continuación se presenta el algoritmo básico para la conversión numérica basada en divisiones sucesivas.



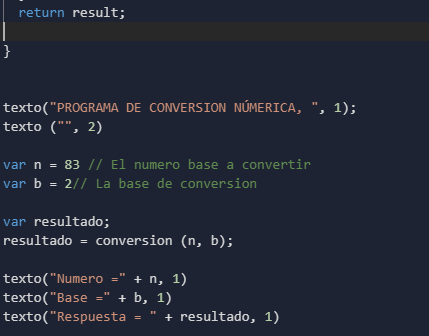
Como se ve en el diagrama, la conversión se realiza dividiendo el número a convertir entre la base seleccionada.

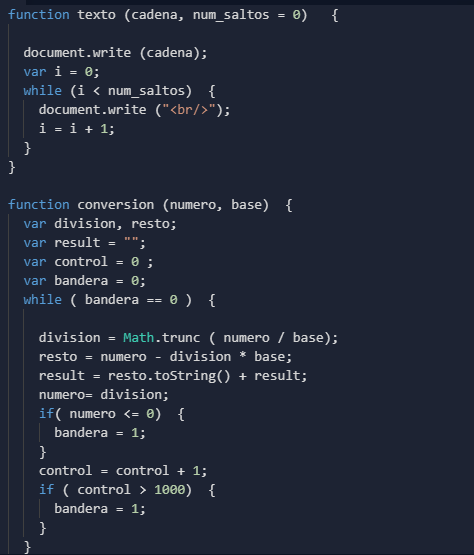
El resultado se obtiene con base en los residuos de las divisiones.

El proceso finaliza cuando se obtiene cero en el resultado de las divisiones.

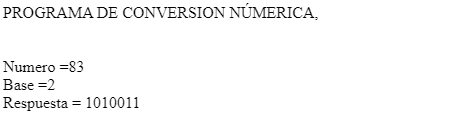
A continuación se presentan las imágenes de los códigos requeridos, para implementar el proceso mostrado en JavaScript. Cada imagen presenta una función distinta, o la ejecución final del programa. Se debe escribir en un solo archivo el código mostrado, y se sugiere un entorno como repl.it.

A continuación se muestra el programa en el entorno repl.it, con los datos de ejecución del programa.





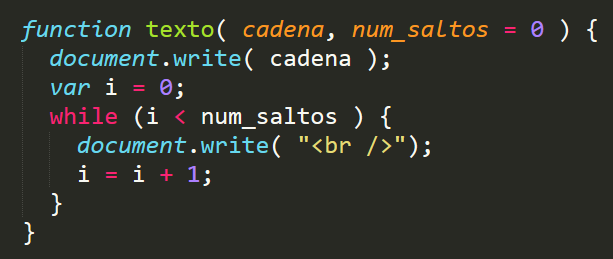
El resultado que arroja el programa es el siguiente:

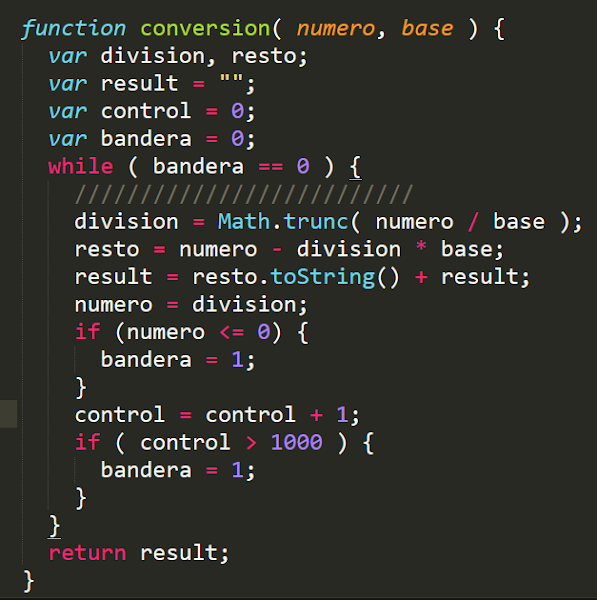


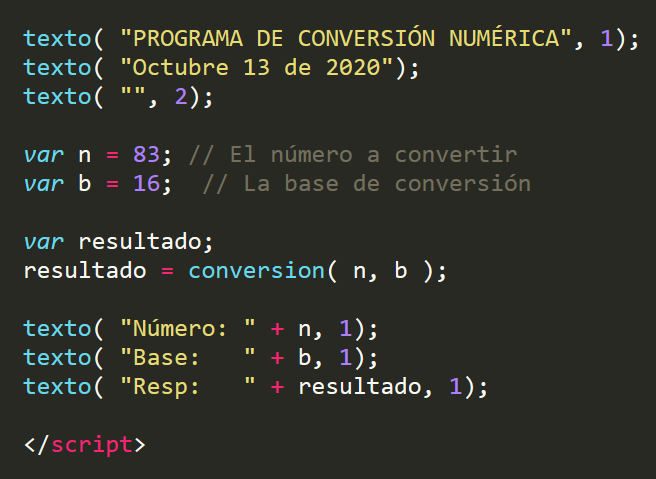
Este el resultado de los residuos, al dividir el número 83 sucesivamente en 2, esto lo hace automáticamente JavaScript con los parámetros que le hemos establecido anteriormente.

# CONVERSIÓN EXTENDIDA

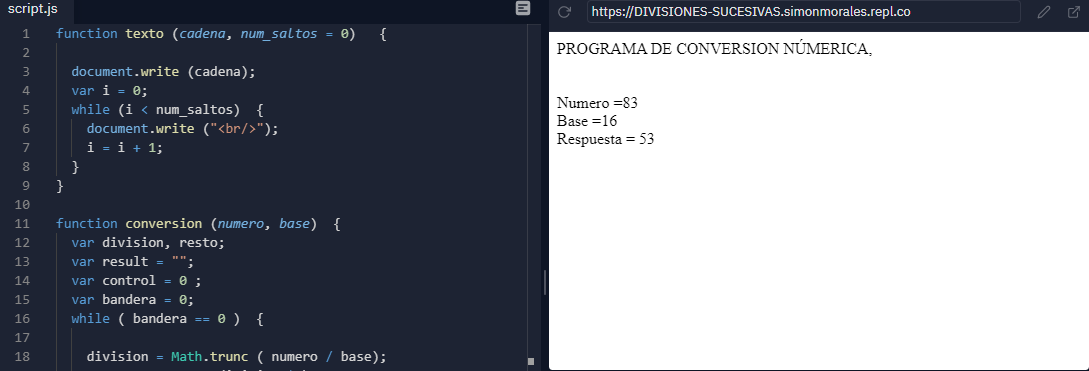
A continuación presentamos el programa de conversión extendida, la cual se encarga de dar tratamiento a los números en base 16.







El resultado que arroja el programa para convertir un número en base 16 es:

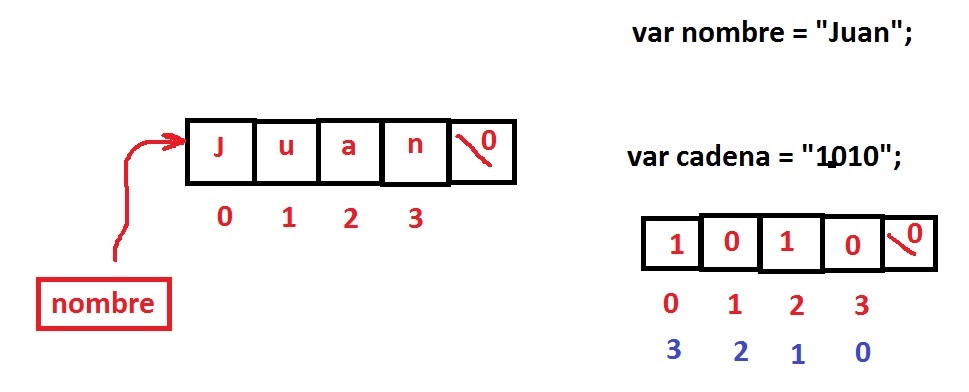


En el programa ejecutado por JavaScript podemos apreciar que la respuesta de pasar el número 83 a base 16 o hexadecimal, nos da el resultado de 53, ya que al dividir el número, el restante de la primera división es 3, y ejecutando un segundo paso recursivo obtenemos el número 5, estos se unen y forman el 53 que vemos en la pantalla de iteraciones.

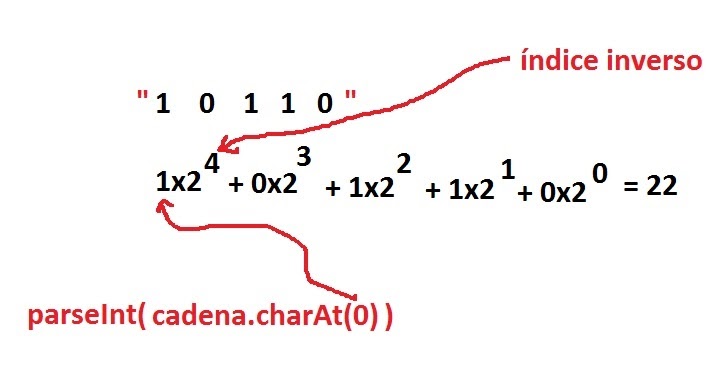
# CONVERSIÓN: 0, 1 A PALABRAS: CERO, UNO

Vamos a presentar el programa que convierte los dígitos binarios 0, 1 a las palabras Cero, Uno.

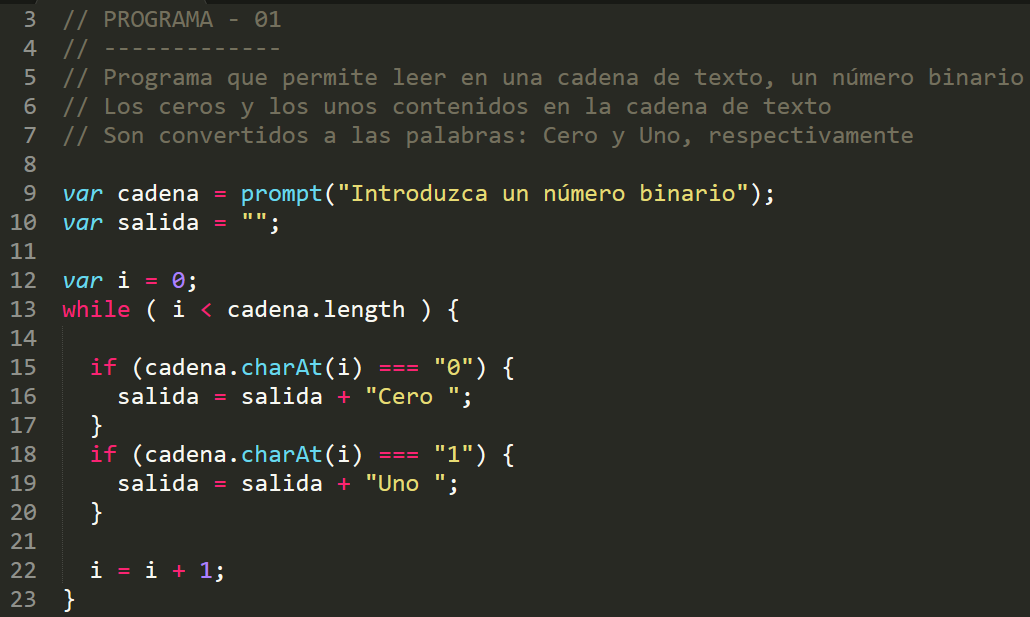
Algunos elementos teóricos. En primer lugar, observamos cómo se almacena una cadena en la memoria:



En la siguiente gráfica, presentamos el índice inverso:



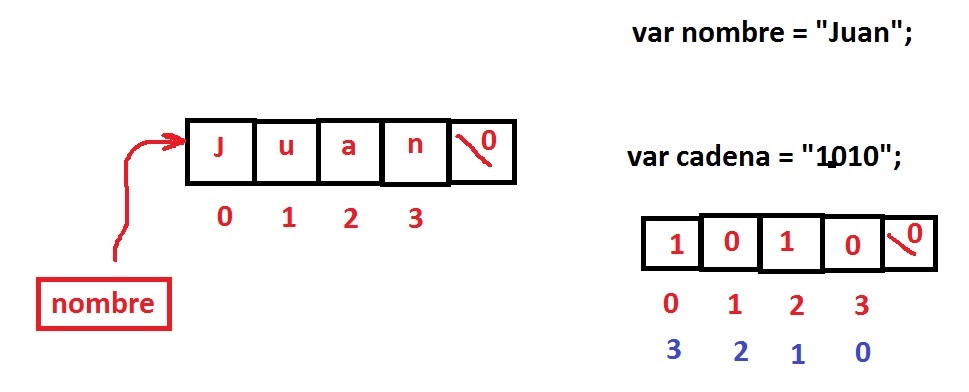
A continuación, presentamos el código fuente, seccionado por partes:



En estas líneas se comenta el programa, se crean algunas variables, se pregunta por una cadena binaria y se procede luego a realizar la conversión, dígito a dígito.

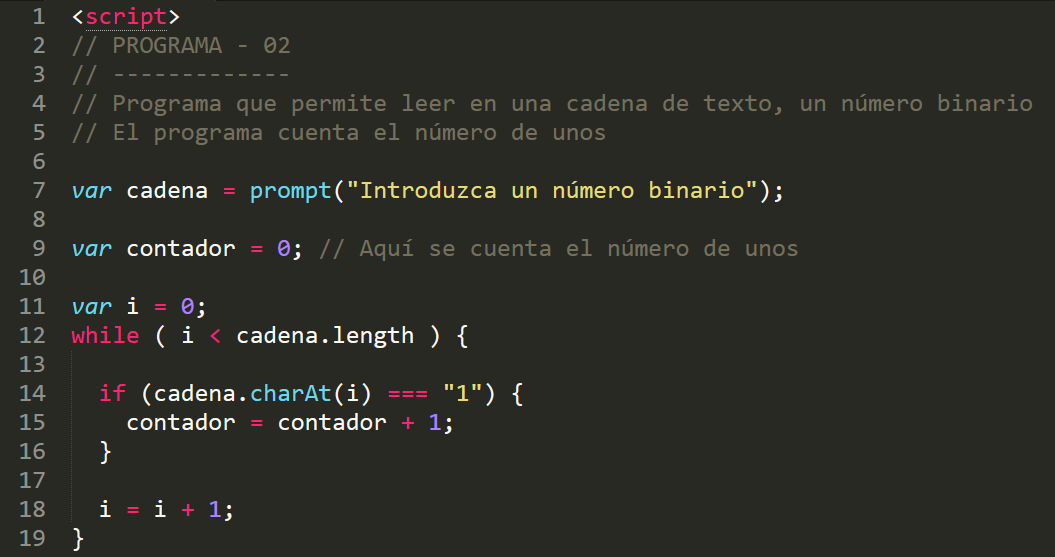
# CONTAR NÚMERO DE UNOS EN UN BINARIO

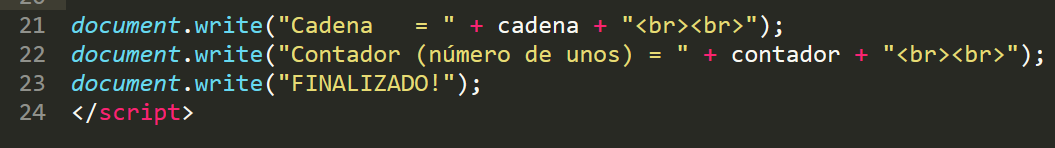
Vamos a presentar el programa que cuenta el número de unos disponibles en un binario. Veamos la estructura de la cadena que contiene el binario:



El número de unos se calcula recorriendo la cadena de izquierda a derecha, evaluando cada una de las posiciones en la cadena, y cada vez que el contenido de dicha posición sea uno, sumaremos 1 a un contador preparado para ello.

Seguidamente presentamos el código fuente.

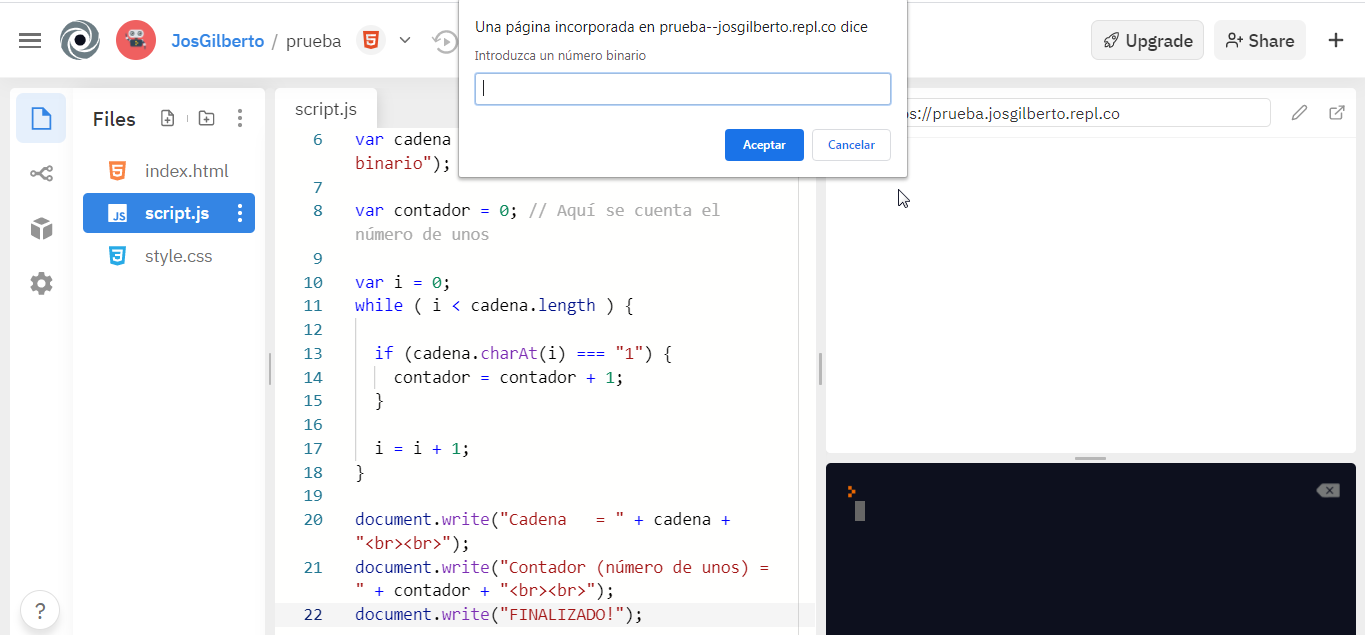


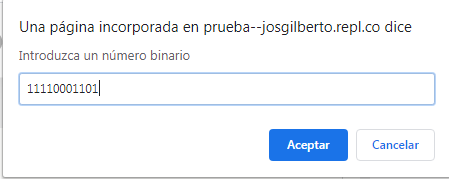


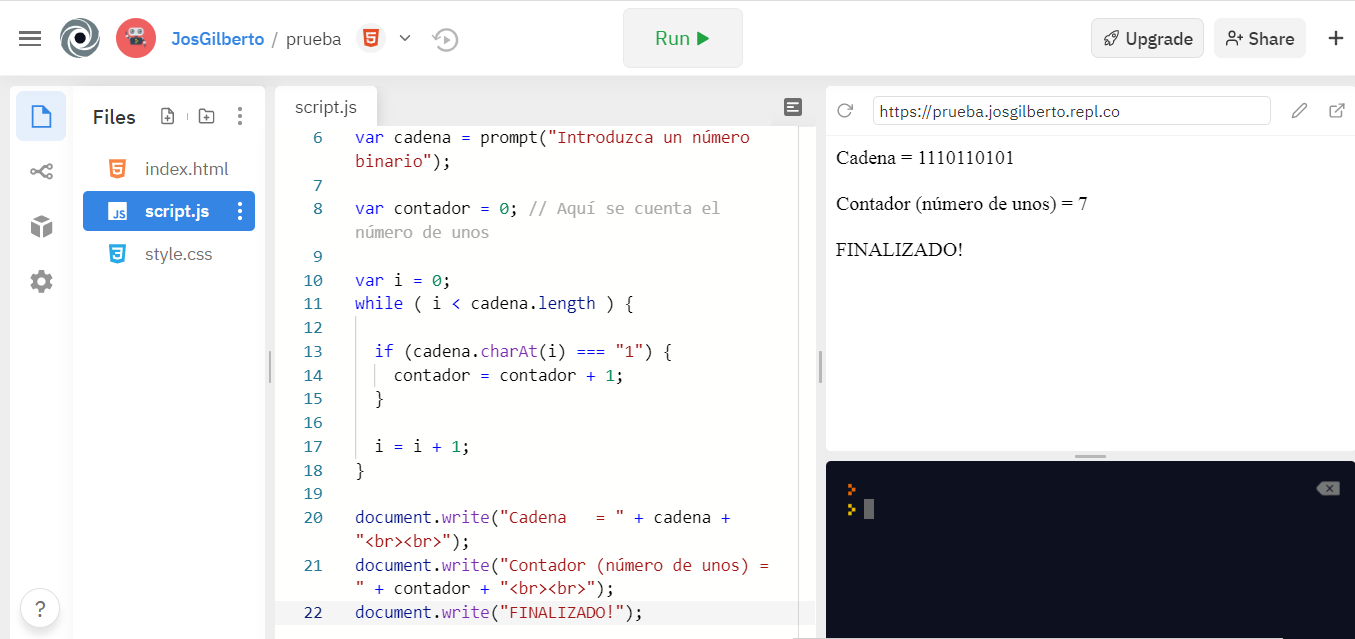
El programa ejecutado, primero, pide por medio de una ventana emergente que ingrese un numero binario, seguidamente define a este número ingresado por el usuario como “cadena”.

Después de esto se crea otra variable llamada “contador” que recibe el valor 0 y también otra llamada “i” que recibe el valor de 0, se crea un condicional, con una condición de parada la cual es, sí la variable “i” es menor a la longitud de caracteres que tenga “cadena”, cree otro condicional If , que sí , en la posición “i”esima encuentra el número 1 , que contador se llame recursivamente para sumarse 1, al final se muestre la suma recursiva de la variable “contador”.

Al ejecutar en repl.it, se obtiene el resultado mostrado seguidamente:

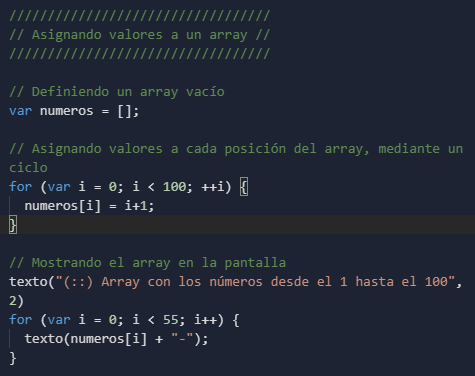




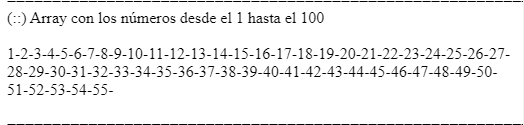


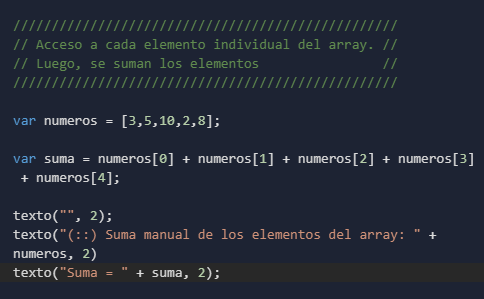
# ARRAYS EN JAVASCRIPT

A continuación se presenta el código del programa de arrays o arreglos en JavaScript, los cuales ejecutan diferentes acciones dependiendo de lo que se requiera:

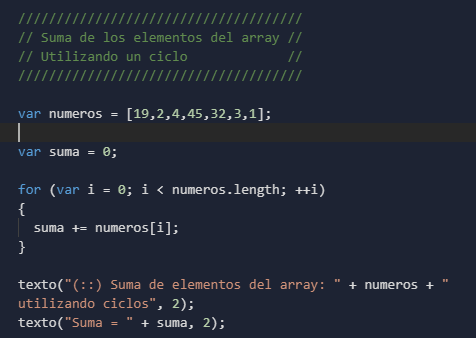
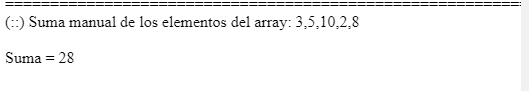


RESULTADO

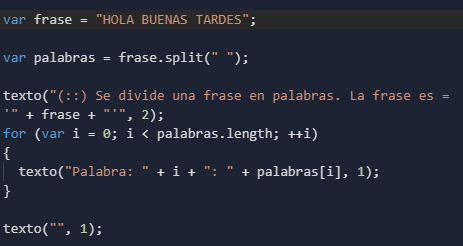
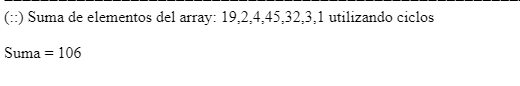




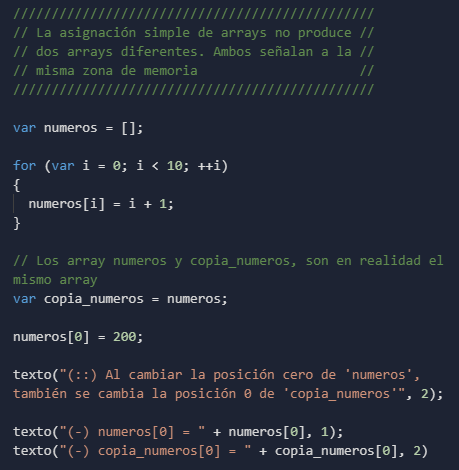
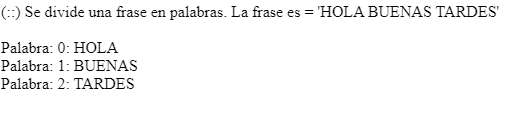
RESULTADO



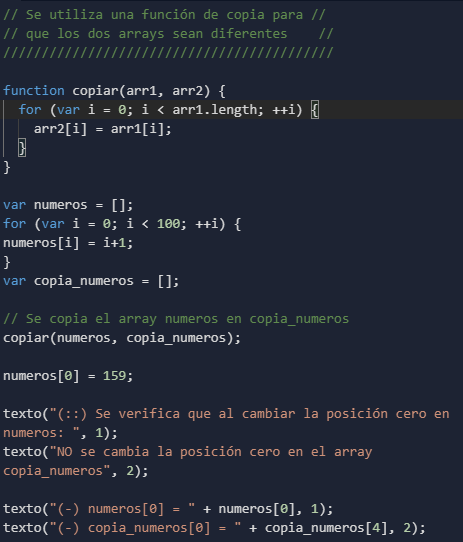
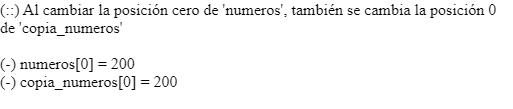
RESULTADO



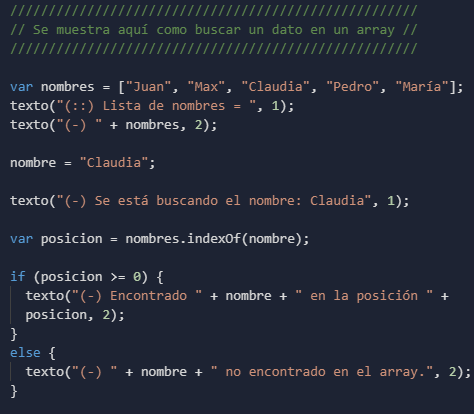
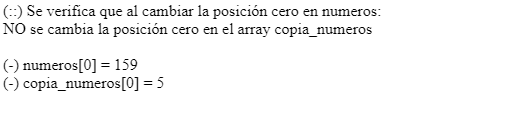
RESULTADO



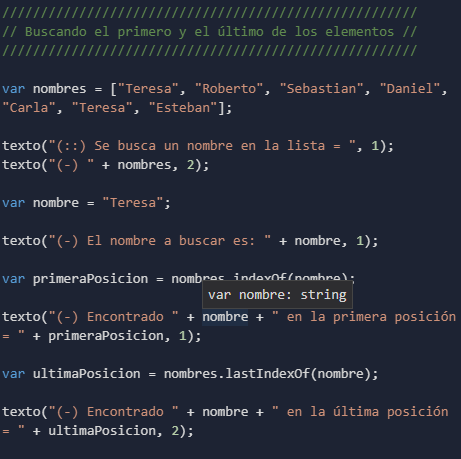
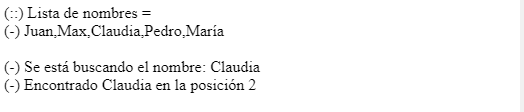
RESULTADO



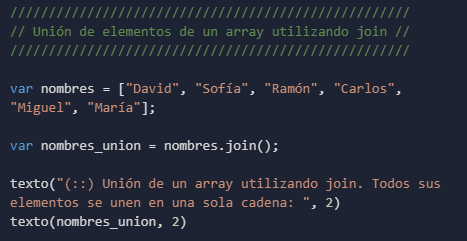
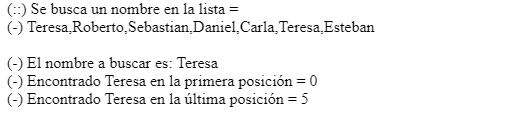
RESULTADO



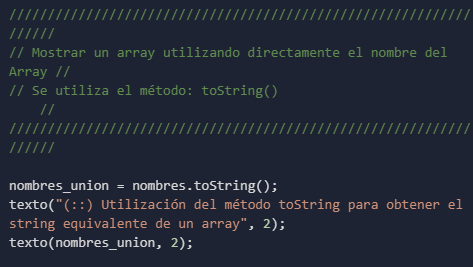
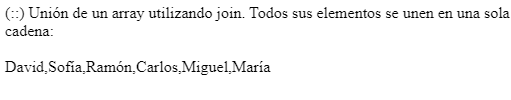
RESULTADO



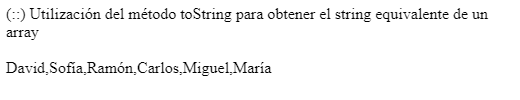
RESULTADO



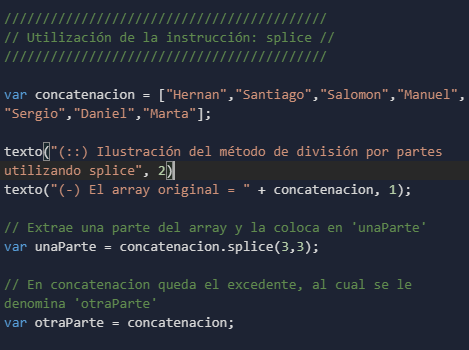
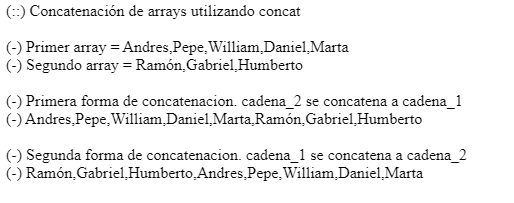
RESULTADO



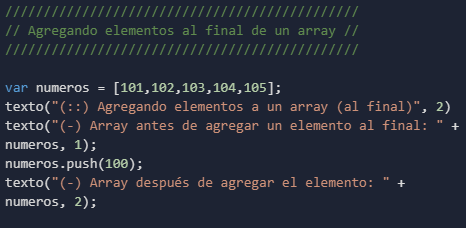
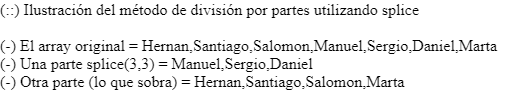
RESULTADO



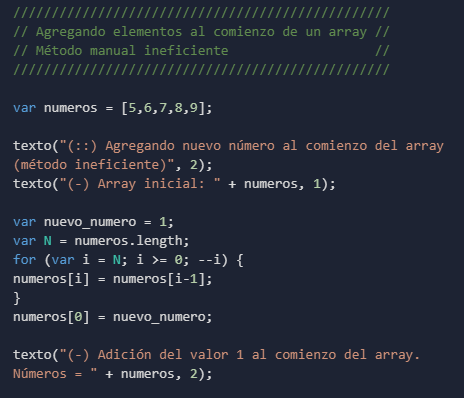
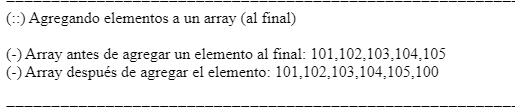
RESULTADO



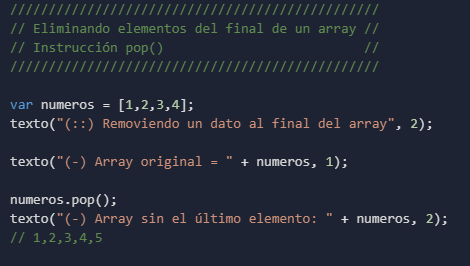
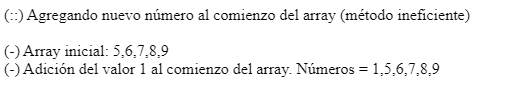
RESULTADO



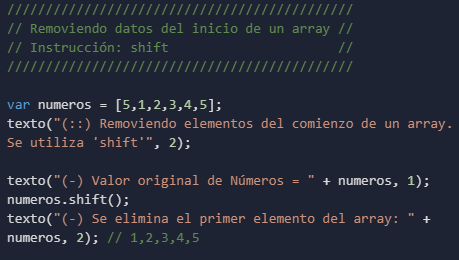
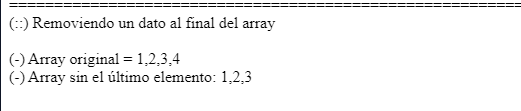
RESULTADO



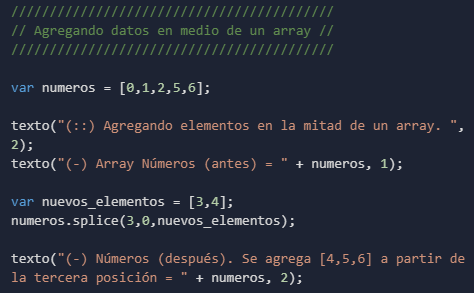
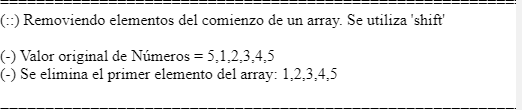
RESULTADO



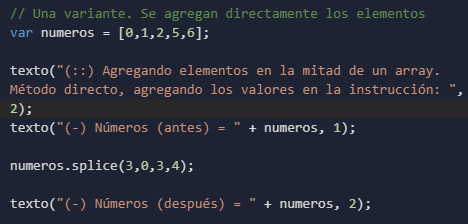
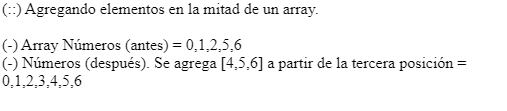
RESULTADO



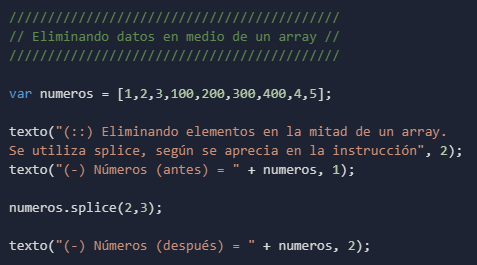
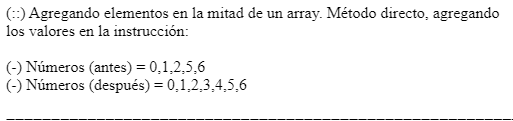
RESULTADO



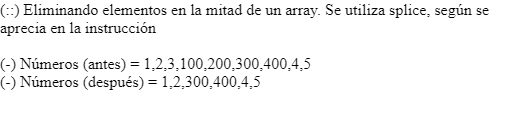
RESULTADO

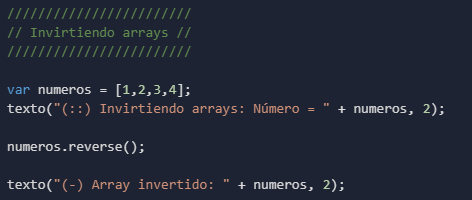


RESULTADO

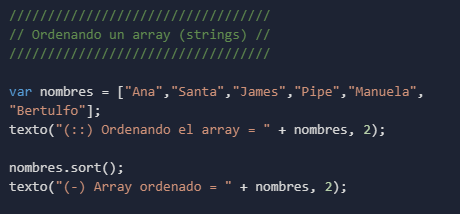
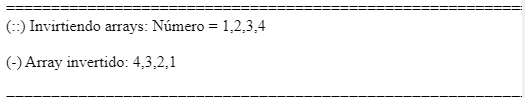


RESULTADO

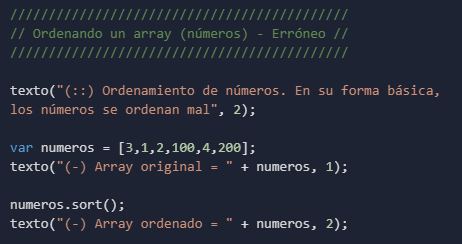
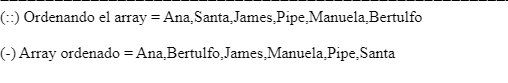




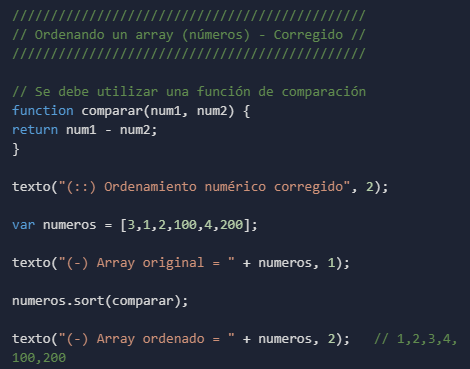
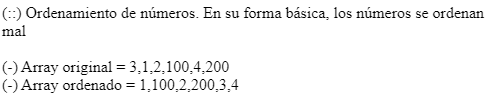
RESULTADO



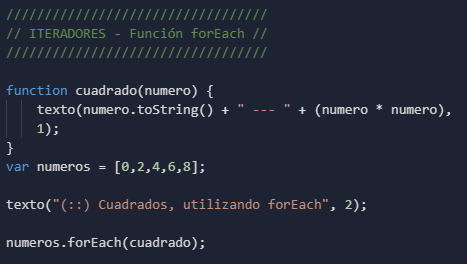
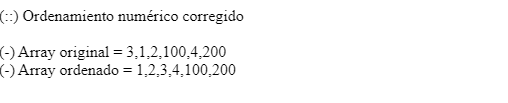
RESULTADO



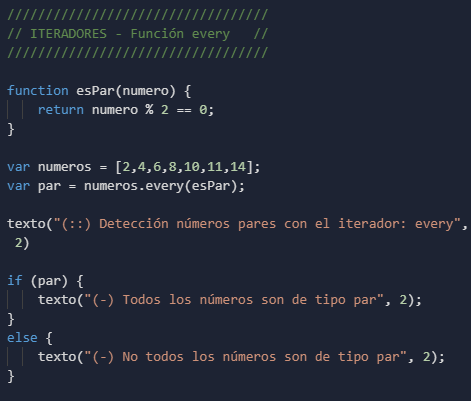
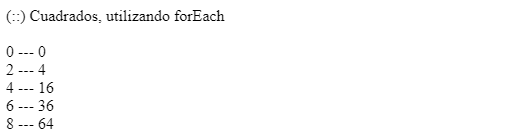
RESULTADO



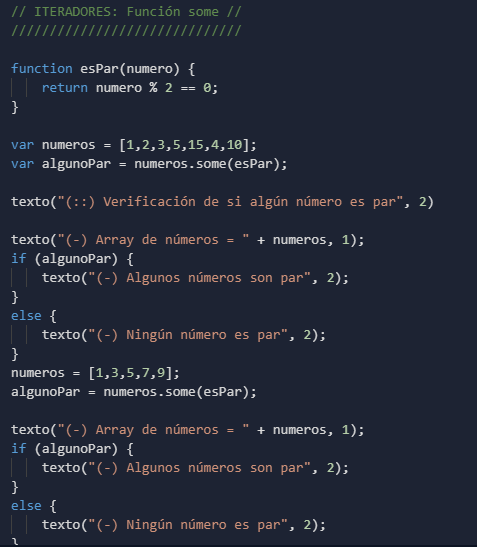
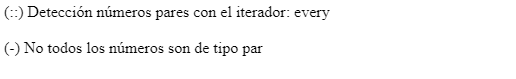
RESULTADO



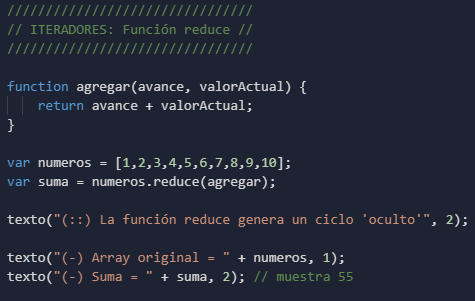
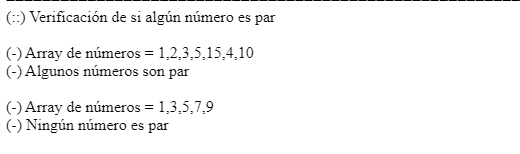
RESULTADO



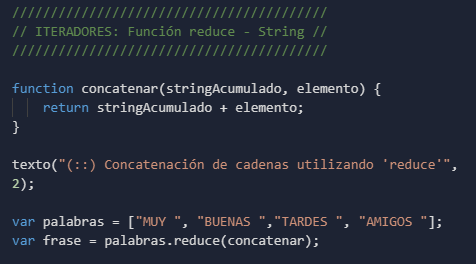
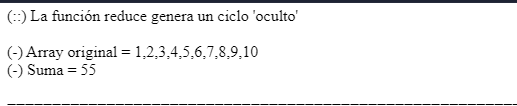
RESULTADO



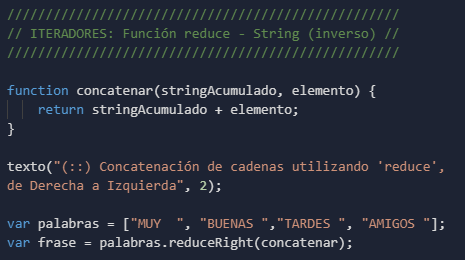
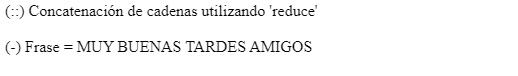
RESULTADO



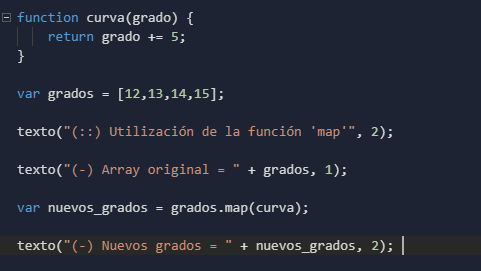
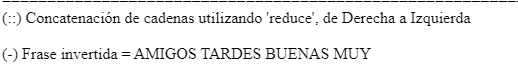
RESULTADO



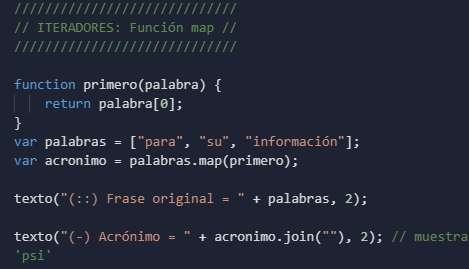
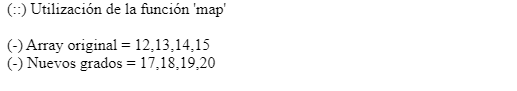
RESULTADO



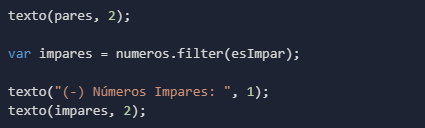
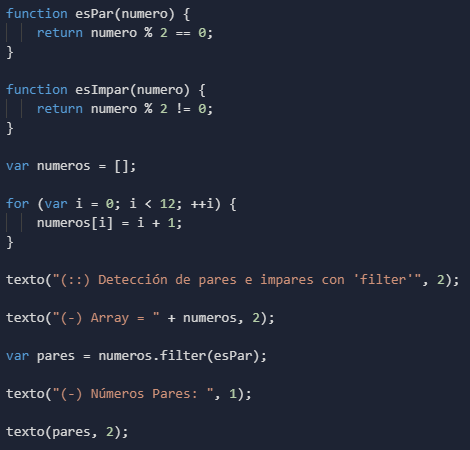
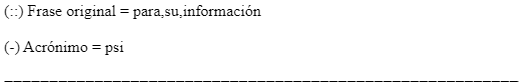
RESULTADO



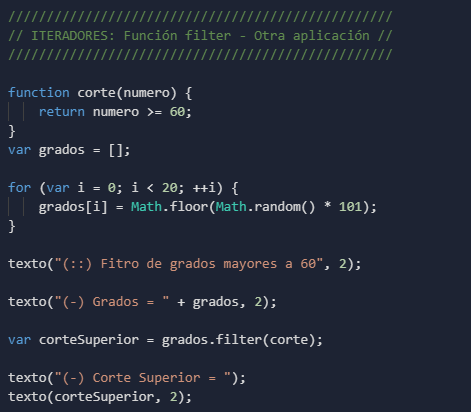
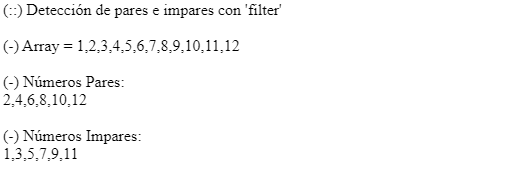
RESULTADO



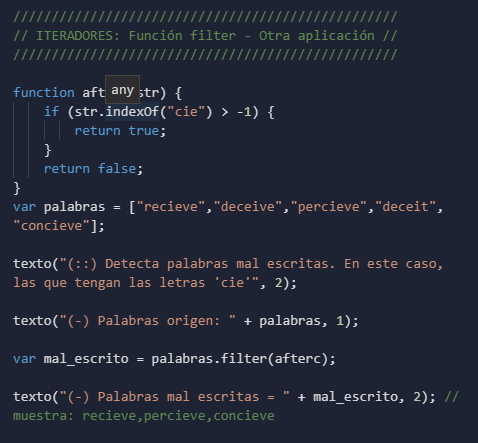
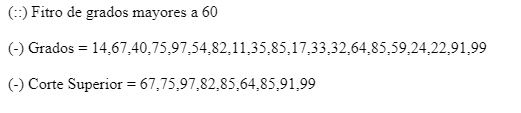
RESULTADO



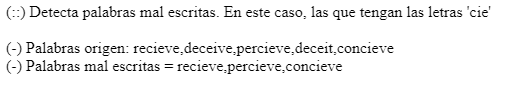
RESULTADO

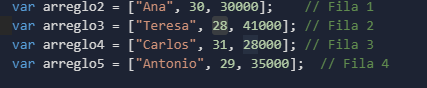
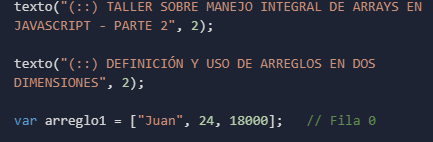


RESULTADO

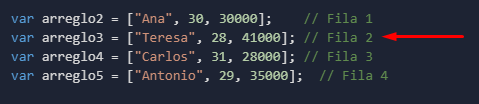
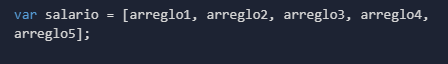


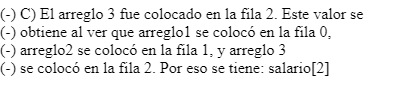
RESULTADO

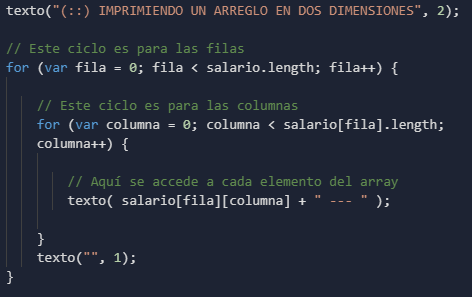


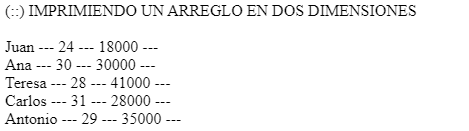


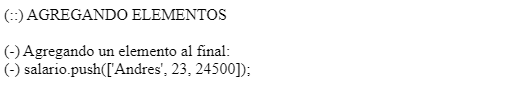
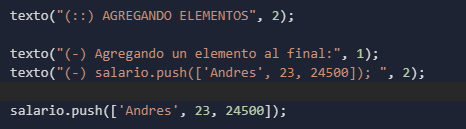
RESULTADO



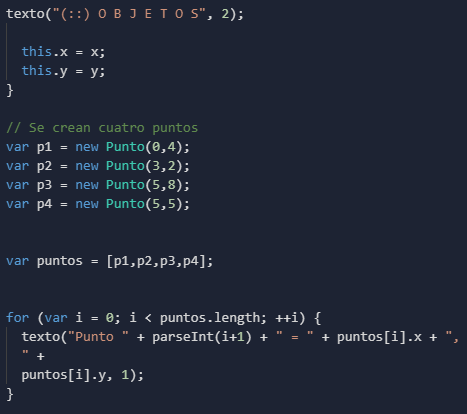




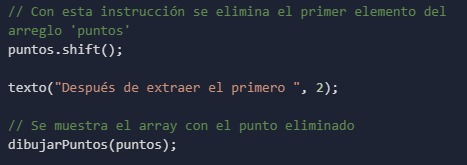
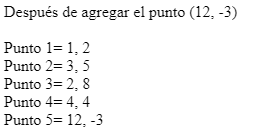
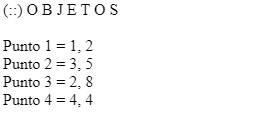








RESULTADO



# 

# CONCLUSIONES

El desarrollo de las temáticas elaboradas en clase utilizando el lenguaje JavaScript prueba ser un mecanismo de gran valor para el aprendizaje de los conceptos básicos de la materia.

En el transcurso de la materia se vienen adelantando diferentes temáticas pero con la implementación de un lenguaje de programación como es JavaScript para resolver los diferentes problemas que se nos presentan al momento de hablar sobre los sistemas numéricos, porque es un lenguaje muy intuitivo y fácil de comprender.

# BIBLIOGRAFÍA

<https://repl.it>

<https://github.com/SimonMorales36/SEGUNDA-PREVIA-INT-INF/tree/main>

<https://classroom.google.com/u/1/c/MTQzMTUwOTE2MTc2/a/MTY1NDQxOTAzOTY1/details>