ARRAYS : Introducción a la Informática

UTP | Pereira

sIMON MORALES TORRES

NOVIEMBRE DE 2020

2020

# CONTENIDO

[1 CONTENIDO 1](#_Toc58066404)

[2 PRESENTACIÓN 2](#_Toc58066405)

[3 ARRAYS EN JAVASCRIPT 4](#_Toc58066406)

[4 CONCLUSIONES 20](#_Toc58066407)

[5 BIBLIOGRAFÍA 21](#_Toc58066408)

# PRESENTACIÓN

Este documento presenta la explicación de los diferentes códigos escritos en la clase Introducción a la informática sobre el tema de los ARRAYS en JavaScript, en este se especifican la importancia y la utilización de los mismos los cuales , son ejecutados y utilizados en la plataforma de Repl.it donde se verán el manejo de estos arrays.

**AUTOR: Simón Morales Torres**

**1004700556**

[**simon.morales@utp.edu.co**](mailto:simon.morales@utp.edu.co)

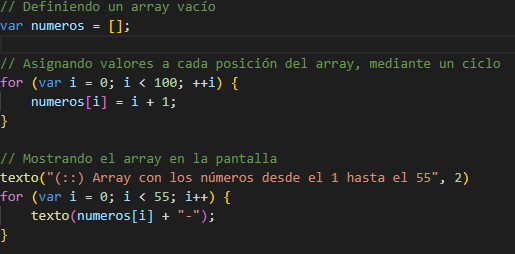
**github.com/SimonMorales36**

# ARRAYS EN JAVASCRIPT

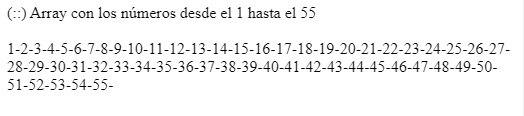
ARRAYS EN JAVASCRIPT

En el siguiente apartado se comentarán los códigos realizados en clase explicando cada una de sus funciones para una mejor comprensión de lo que es un array por medio de imágenes y su código.

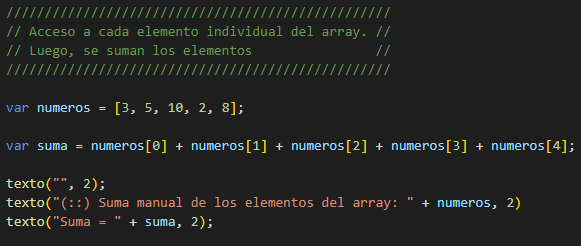
1. IMPRIMIR NUMEROS EN PANTALLA DE MANERA RECURSIVA EN UN ARRAY



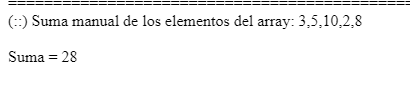
En el código vemos que en el array “números” no hay valores, esta vacío, y podemos ver que hemos creado un ciclo, el cual tiene una función recursiva adentro, la cual va sumando de 1 en 1 hasta llegar al limite o condición de parada y va llenando el array y mostrándolos en pantalla



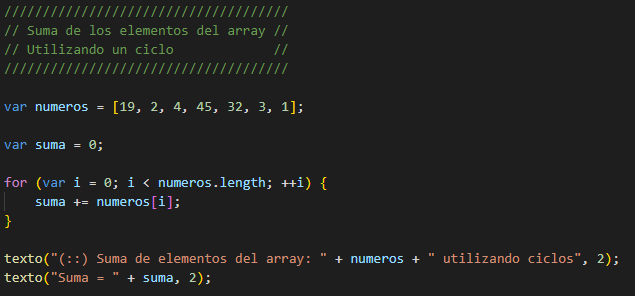
1. SUMA DE VALORES EN UN ARRAY ITERARIVO



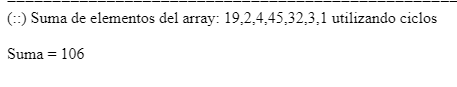
Esta es una suma iterativa en un array, en el cual se toma cada elemento del array basados en un índice y el resultado es mostrado por pantalla.



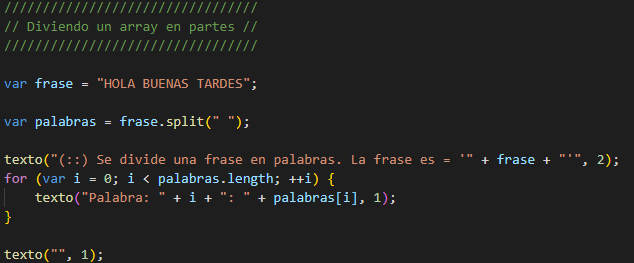
1. SUMA DE ARRAYS RECURSIVO



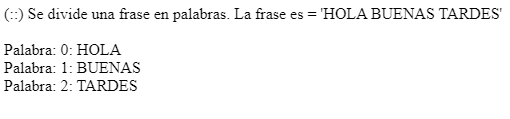
En el código creamos un array de cantidad n de elemento y por medio de un ciclo que va incrementándose 1 a 1 hasta llegar al índice máximo del array, tenemos que la variable suma recibe un numero dependiendo de la posición en la que este el índice y se va sumando con el siguiente.



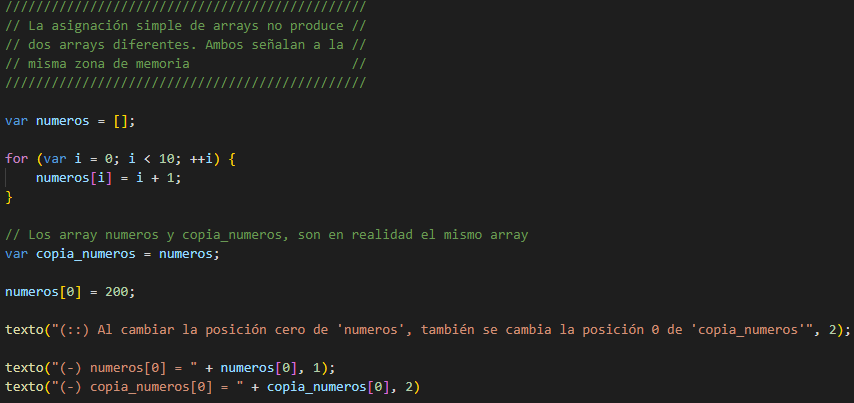
1. DIVIDIR UN ARRAY EN PARTES



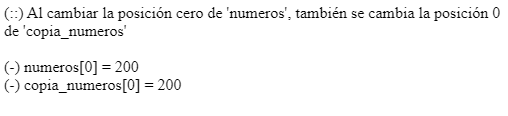
Para dividir esta array llamaremos una función de JavaScript llamada Split, la cual separara las palabras cada vez que encuentre un espacio entre ellas y con un ciclo le indicamos el recorrido de la frase.

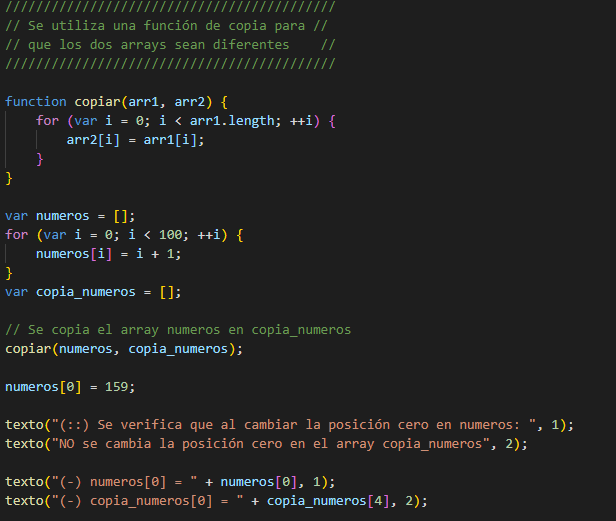


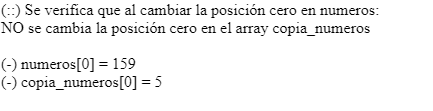
1. ASIGNACION DE ARRAYS



RESULTADO

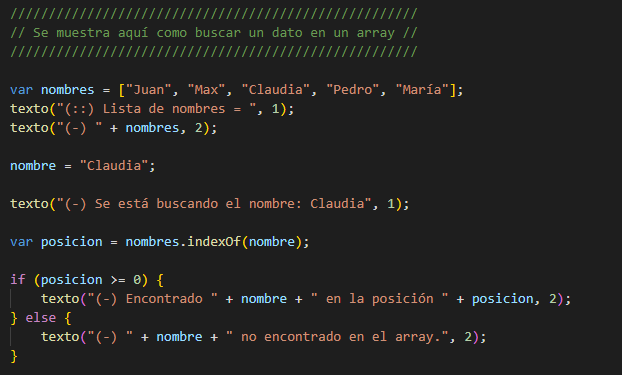






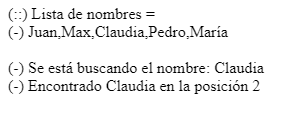
En la asignación de arrays al asignarle a la variable números un valor, y después copiar ese array en otra variable, obtenemos que cuando cambiamos la primera también cambiamos la segunda, para corregir esto tenemos que crear una función, que al ingresar dos arrays, estos tendrán la misma información, pero podemos cambiar cualquiera de los dos y no afectaría al otro.

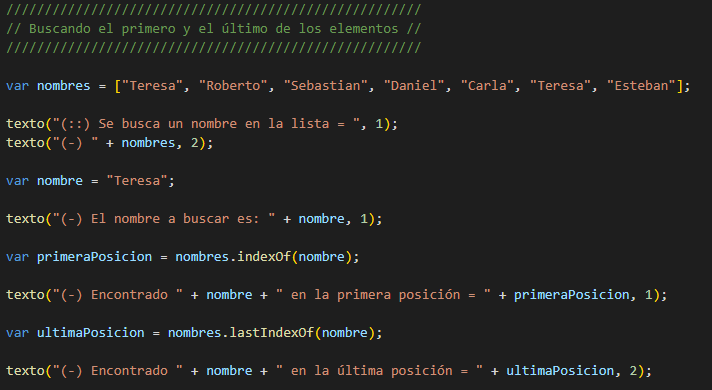
1. BUSCAR DATOS EN UN ARRAY

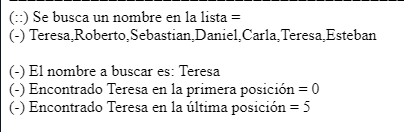


Para encontrar un dato en el array, utilizamos la función de JavaScript IndexOf, el cual busca en el índice de un array comenzando en 0, lo que le pidamos que busquemos en este caso el nombre Claudia, el cual lo encuentra en la posición 2 del array, en caso de no encontrarlo creamos un condicional el cual indica si se encontró o no en el array despliegue un texto.

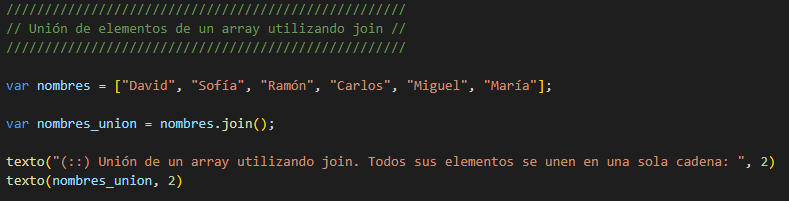
También el lenguaje posee la función LastIndex el cual busca lo que hay en la ultima posición en la que se encuentra un dato en el Array.



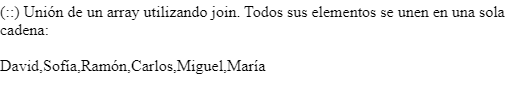


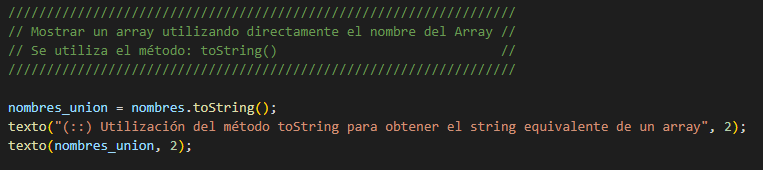


1. UNION DE ELEMENTOS EN UN ARRAY

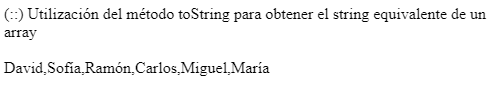


Función join la cual une todos los elementos de un array y los convierte en una cadena.

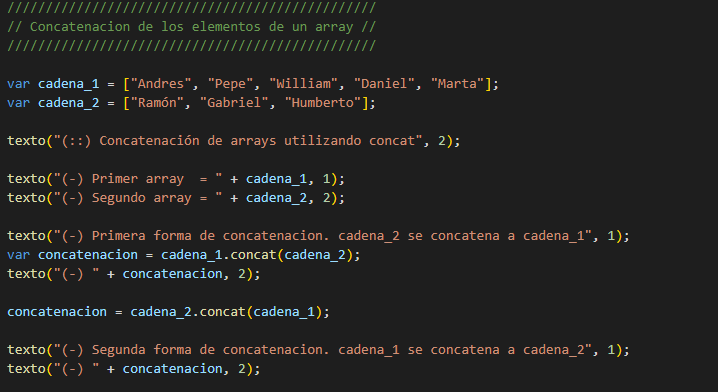




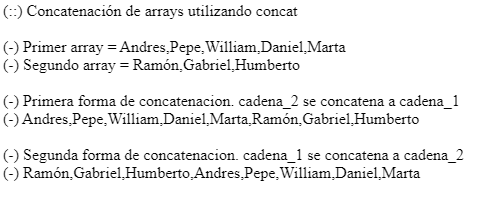
También podemos utilizar la función ToString.



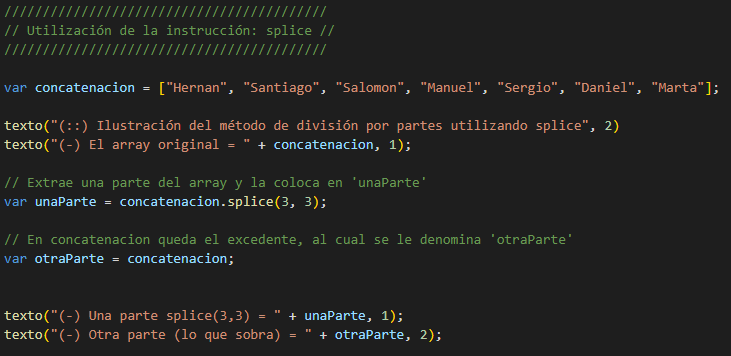
1. CONCATENACION



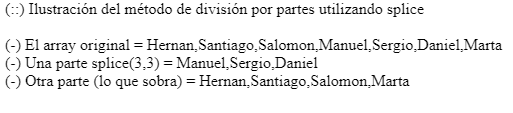
La concatenación es unir los elementos de dos diferentes arrays y unirlos en uno solo, para esto usamos la función concat , el orden depende de que array queremos primero que otro.



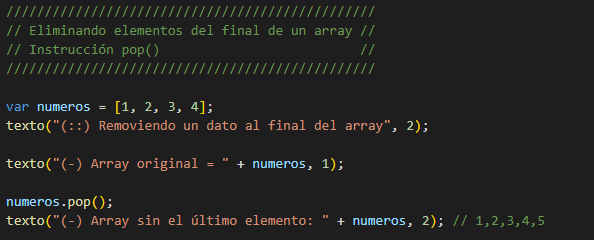
1. SEPARACION DE ARRAYS CON SPLICE



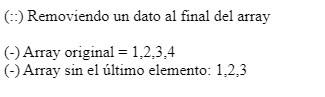
Aquí se usa la función splice , que recibe desde que posición del índice se empieza, hasta la cantidad de elemento se van a utilizar, esto afecta a el array , el cual si lo mostramos otra vez después del proceso, solamente nos arrojara el sobrante de los elementos y no todo el array completo.

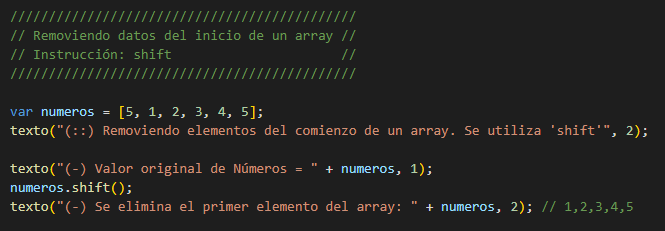


1. REMOVER ELEMENTOS DE LOS ARRAYS

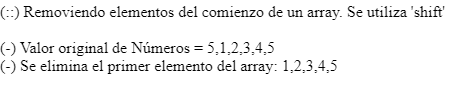


Con esta función la cual recibe por parámetros el array y el .pop elimina el último elemento de un array

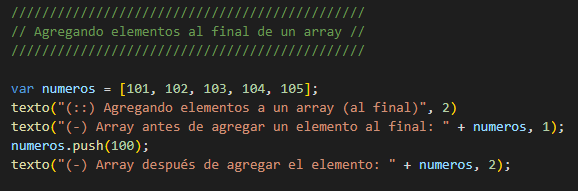




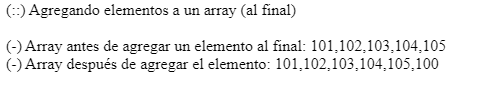
función Shift elimina el primer elemento de un array.

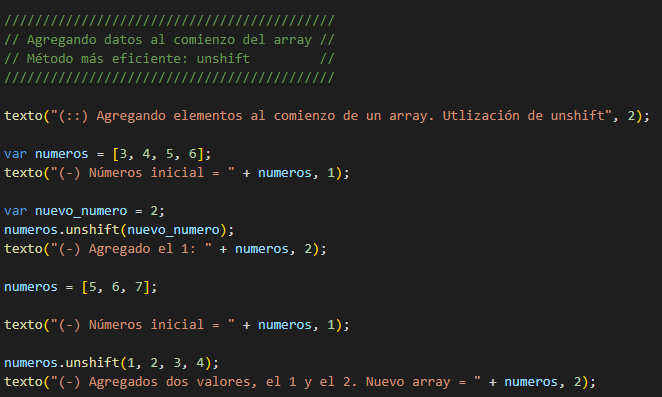


1. AGREGAR ELEMENTOS A UN ARRAY

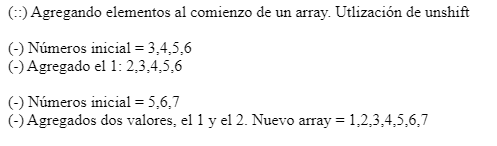


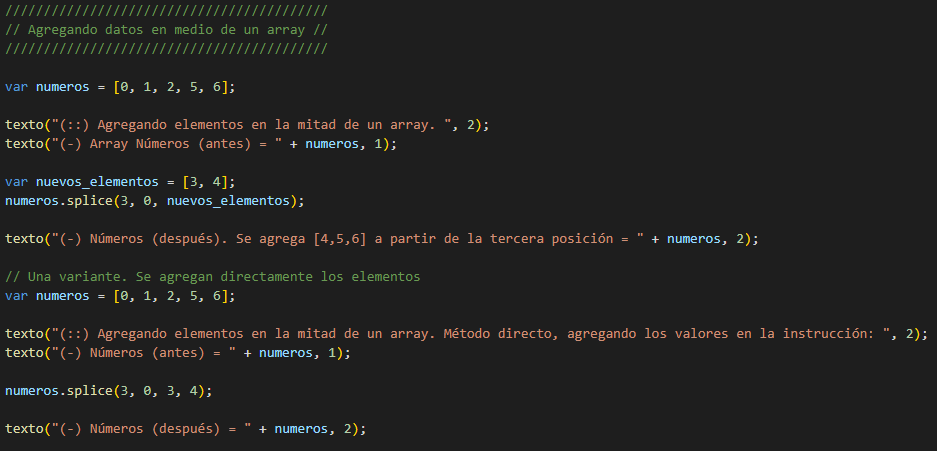
La función push recibe un array y “empuja” o agrega un elemento a un array en la última posición.



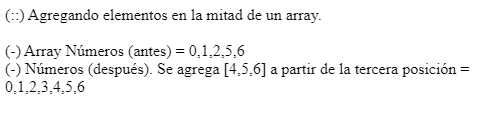


Esta función agrega al comienzo de un array cualquier cantidad de elementos.

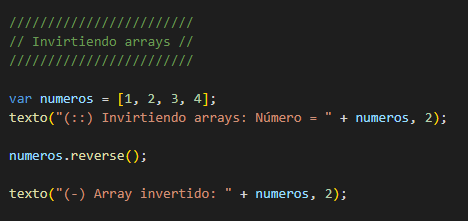




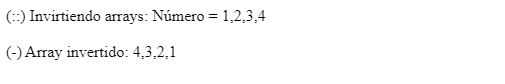
Para agregar datos en el medio de un array se utiliza también la función splice, la cual tiene como parámetro para agregar elemento, el nombre de la función . splice desde donde se va a empezar a modificar el array , 0 para indicar que no se retirara ningún elemento y los nuevos elementos ingresados al array.

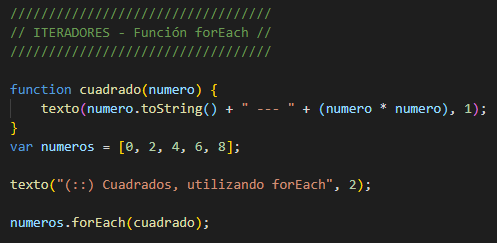


1. ITERADORES Y MODIFICADORES DE ARRAYS

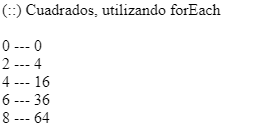


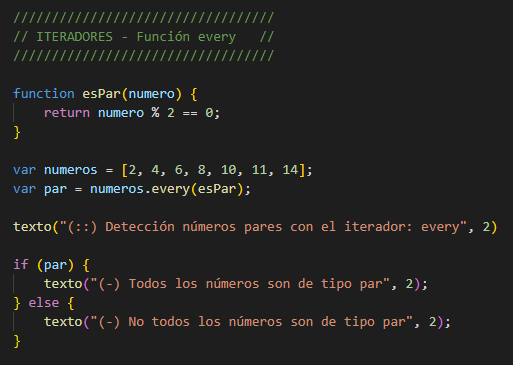
Función para invertir un array , al invertirlo , cuando se vuelva a llamar al array ya quedara como un array invertido



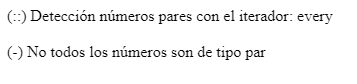


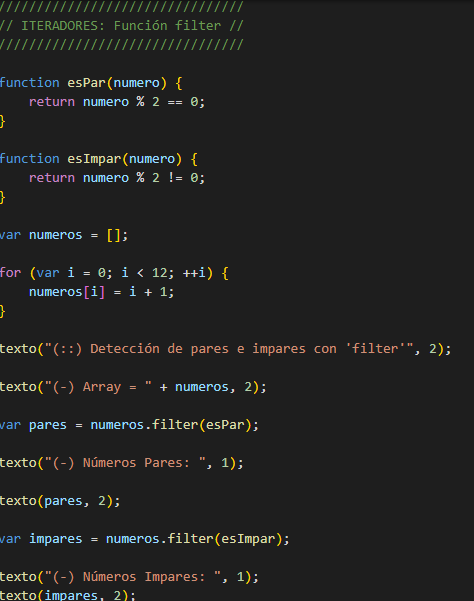
La función ForEach toma cada uno de los valores del array y les aplica el procedimiento requerido, es más practico que hacer un ciclo recursivo.



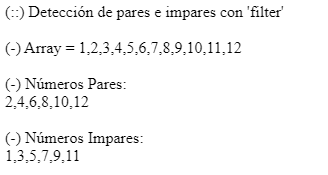


En este código tenemos un arrays con diferentes números, creamos la función para identificar números pares, después aplicamos la función, every la cual evalúa cada uno de los elementos de un array y lo analiza con la función de los pares para ver si alguno de los elementos de tipo par ,si todos son pares , indicara que todos son pares , sino indicara lo contrario.





La función filter, como su nombre en inglés lo indica filtra los valores que le entregamos de acuerdo a unos parámetros establecidos, los cuales son números pares o números impares, esta vez vamos a ver si un array con los números del 1 al 12 generados recursivamente tienen la propiedad de ser pares o no con la función filter, después vamos a mostrar por pantalla cuales son pares o impares



ARREGLOS EN DOS DIMENSIONES JAVASCRIPT

JavaScript en su lenguaje no posee manejo de arreglos bidimensionales , pero podemos crear un mecanismo el cual recorra las filas y las columnas en los arreglos para que nos muestren los diferentes datos que tenemos almacenados conjuntamente así :

Texto

Descripción generada automáticamente

Este es el código de cada uno de los arrays en los que queremos ingresar los datos de persona en este caso pero como podemos acceder a ellos siendo estos diferentes, pues se almacenan dentro una variable , la cual contiene todos los arrays, y como podemos acceder a los elementos de uno en específico, de la siguiente manera :

Texto

Descripción generada automáticamente

De esta manera, creando un ciclo recursivo tenemos que podemos recorrer todas y cada una de los elementos en los arrays esto , es un manejo básico de arreglos bidimensionales en JavaScript.

Imagen que contiene Texto

Descripción generada automáticamente

Aquí mostramos el resultado de tener los datos de cada uno de los arrays que tenemos en el código.

# 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.

Podemos encontrar en los arrays maneras de almacenar diferentes tipos de elementos y manejarlos de la manera mas conveniente posible dependiendo de las necesidades del usuario, tenemos varias funciones en el lenguaje que nos ayudan a ejecutar esto de manera mas sencilla para el programador y como podemos utilizarlos como una forma muy básica de un pequeña base de datos, donde podemos manipular los elementos que tengamos en cualquier momento.

# BIBLIOGRAFÍA

<https://repl.it>

SimonMorales36/github.com