***2-JVSI***

***Variable: Son cajitas donde de pueden guardar datos.***

***Funciones : Son programas que llamas con una orden, donde escribimos la pregunta para ordenar que tiene que hacer.***

***String: Son palabras que de guardan en una función.***

***Argumentos en lo que se encuentra dentro de las funciones.***

***If: Son preguntas que se hacen en la función, si esto es mayor que x?***

***Valores Booleano: Cuando una pregunta esta bien, entonces es verdadero, sino es falsa.***

***3-JVSII:***

***For: Es una función que se utiliza para hacer tareas repetitivas.***

## ***&&: //, ! : operadores lógicos***

***&&: Evalúa ambas expresiones dende devolverá verdadero si ambas expresiones son true, y si uno de las expresiones false devolverá false.***

***//: or determina si una expresión es true, devolverá true si ambas expresiones son true, y si ambas son false devolverá false.***

***!: NOT, devolverá el opuesto booleano en el contexto.***

***4-JVSIII:***

***Arrays: Son listas de elementos donde podemos acceder y como objetivo modificarlos o destruir esos elementos, se llaman según su posición, o su índice que a vez se guarda en una variable.***

***5-JVSIV***

* ***Objetos: Todo el tiempo de ejecución de Javascript es un objeto, por lo tanto, los objetos contienen mucha información de una sola cosa. Nos dan la posibilidad de hacer una estructura de datos donde tenga propiedades que sean nombres string que estén asociados algún valor.***
* ***Propiedades: Están dentro del objeto, se le puede asignar un valor y se acceden a cualquier nombre String y no números como en el array. Existen dos formas de ingresar a las propiedades, una nombrando el objeto con bracket notation por medio de corchetes [..], que es similar a las de los array, y el otro colocando el (nombre del objeto.nombre de la propiedad) que data notación.***
* ***Métodos: Lo valores se establecen en funciones, estas funciones guardadas de llaman métodos.***
* ***Bucle for…in: Los objetos no contiene índices numéricos como los array, debemos utilizar otra sintaxis del bucle “for in loop, se declara entre parentecis la variable,con la palabra ‘in’ y el nombre del objeto.***
* ***Notación de puntos vs notación de corchetes: La anotación por punto define al método, mientras que la anotación por corchetes define un valor asociado del objeto.***

***6-JVSV***

* ***Prototype: El prototype esta dentro del objeto, que a si ves tiene una propiedad que puede compartir otras propiedades que están dentro de la sintaxis del objeto.***
* ***Constructors (de Clases): es una planilla para la creación de objetos de datos predefinido***

***7-JVSVI***

***Funciones Callback: Pasa una función a otro argumento.***

***.forEach: Es un bucle for integrado en cada array, lo toma como único argumento, e itera cada elemeto, puede recibir donde puede tomar dos argumentos, primero el elemeto y segundo el índice.***

***.reduce: ejecuta un bucle y reduce cada elemento en un elemeto que se devuelve Como es el primer argumento, acepta un callback que toma dos argumentos, primero un ‘acumulador’acc (el resultado del método de reducción hasta ahora), y el segundo es el elemento en el que se encuentra actualmente. El callback debe contener siempre una declaración de devolución (“return”).***

***.map: .map se usa cuando queremos cambiar cada elemento de una matriz de la misma manera. .map toma una devolución de llamada como único argumento. Al igual que el método .forEach, el callback tiene el elemento y el índice de argumentos opcionales. A diferencia de .reduce, .map devolverá toda la matriz.***