**MISION DE COEX**

**1. ¿Qué es una variable en JavaScript y cómo se declara?**

Una variable en JavaScript es un tipo de almacenamiento de datos, se pueden asignar en el principio o al final de alguna operación, además se puede llamar para ejecutar otras acciones (para hacerlo se debe escribir el nombre de la variable).   
Para declarar un variable existen 3 formas:   
  
var: hace que la variable sea universal se pueda llamar en cualquier parte del código y su contenido se puede manipular.   
  
Let: Es el mismo concepto que ‘var’ sin embargo un poco más restringido ya que dependiendo en qué parte del código este (queremos llamarla al principio pero esta creada al final de código), o si está entre ‘{}’ la variable puede que no aparezca en alguna parte que deseamos llamarla.   
  
Constant: Esta determinación hace que la variable esté totalmente restringida y que su almacenamiento no se pueda cambiar para nada (se usa cuando queremos que haya un dato fijo y no se pueda cambiar).

**2. ¿Cuál es el alcance de una variable en JavaScript?**

El alcance se recibe desde que tipo de variable es, en estos casos (var, let, constant), quiere decir que dependiendo de la declaración la variable puede tener “libertad o restricciones”.

**3. ¿Cuáles son los operadores que manejan JavaScript?**

Hay varios tipos de operadores los cuales son:   
Operadores de asignación: Tal como dice su nombre de asignación de ese tipo hay varias variaciones pero todas vienen a partir del cual es usado para asignar datos dentro de otro.   
   
Operadores de comparación: Utilizando el mismo ‘=’, sin embargo se debe añadir uno más ‘==’, aparte de eso existen’<, > o !=’, los cuales son comparativos mayor menor igual diferente etc.   
   
Operadores aritméticos: Estos son utilizados para operaciones matemáticas como ‘suma, resta, multiplicación o división’, existen muchos más y varias formas de aplicarlos para dar respuesta a varias condiciones o problemas.   
   
Operadores lógicos: Los lógicos son aquellos que usan ‘&& o ||’, para comparar variables y que cumplan cierto criterio dentro de una operación.   
   
Operadores de cadena: Es aquel que relaciona varias operaciones en una o varias acciones, lo cual sirve para cumplir dichas acciones por ejemplo la creación de un menú).   
   
Operador condicional (ternario): Un ejemplo más conocido es cuando usamos ‘switch - case’ la cual es un tipo de función que sirve para asignar una operación para que se ejecute con una acción.

**4. ¿Cuáles son las estructuras de control que maneja JavaScript?**

JavaScript maneja ciertos tipos lo cual ayuda con las ejecuciones de una operación, sirve además para la validación de datos.

if: Sirve para indicar si una operación es verdadera o dependiendo de lo que tenga dentro de sus {} puede ser falsa

else: Sirve para ejecutar una operación de manera falsa sin embargo igual que el ‘if’ depende del contenido puede ser verdadera

switch: Se utiliza para ejecutar varios datos de códigos y varias opciones en su mismo rango que viene analizado con ‘case’.

Teniendo en cuenta estos datos de tipo selección, también están los tipos de repetición.

while: Se utiliza para repetir un código mientras se cumpla una condición especificada.

do/while: igual que su anterior sin embargo se utiliza para repetir el código al menos una vez y luego continuar repitiéndolo mientras se cumpla una condición especificada.

for: Es un tipo de repetidor pero con una limitación si es puesta, lo particular de esta es que se repite la secuencia haciendo un tipo de suma la cual se almacena y pasa a la siguiente

**5. ¿Cuál es la sintaxis del operador ternario en JavaScript?**   
a) if (condición) {valor si verdadero} else {valor si falso}   
b) (condición) ? valor si verdadero : valor si falso   
c) switch (condición) {case valor1: código; break; case valor2: código; break;}

**B**

**6. ¿Cuál es la sintaxis básica del ciclo for en JavaScript?**

a) for (inicialización; condición; incremento) {código}

b) while (condición) {código}

c) do {código} while (condición)

**A**

**7. ¿Cuál es la sintaxis básica del ciclo while en JavaScript?**

a) for (inicialización; condición; incremento) {código}

b) while (condición) {código}

c) do {código} while (condición)

**B**

**8. ¿Cuál es la sintaxis básica del ciclo do-while en JavaScript?**

a) for (inicialización; condición; incremento) {código}

b) while (condición) {código}

c) do {código} while (condición)

**C**

**9. ¿Cuál es la sintaxis básica del condicional if-else en JavaScript?**

a) if (condición) {código si verdadero}

b) if (condición) {código si verdadero} else {código si falso}

c) switch (condición) {case valor1: código; break; case valor2: código; break;}

**B**

**10. ¿Cuál es la sintaxis básica del condicional switch en JavaScript?**

a) if (condición) {código si verdadero}

b) if (condición) {código si verdadero} else {código si falso}

c) switch (condición) {case valor1: código; break; case valor2: código; break;}

**C**

**Ejercicio con estructuras de control:**

**11. Escribir una función en JavaScript que tome como parámetro un número y devuelva, "Es par" si el número es par y "Es impar" si el número es impar.**

function fParImpar(numero) {

if (numero % 2 === 0) {

return "Es par";

} else {

return "Es impar";

}

}

let vN = Number(prompt("Ingrese un numero"));

console.log(fParImpar(vN));

**Ejercicio con operador ternario:**

**12. Escribir una función en JavaScript que tome como parámetro un número y devuelva, "Positivo" si el número es mayor que cero, "Negativo" si el número es menor que cero y "Cero" si el número es cero.**

function fPosiNega(numero) {

if (numero > 0) {

return "Es positivo";

} else {

return "Es negativo";

}

}

let vN = Number(prompt("Ingrese un numero"));

console.log(fPosiNega(vN));

**13. Escribir una función en JavaScript que tome como parámetro un número y muestre**

**por consola los números del 1 al número ingresado.**

**Ejercicio con estructuras de control y repetitivas:**

**14. Escribir una función en JavaScript que tome como parámetro un número y muestre por consola los números del 1 al número ingresado, pero si el número es múltiplo de 3 mostrar "Fizz" en lugar del número y si el número es múltiplo de 5 mostrar "Buzz" en lugar del número**

function fFizz(numero) {

for (var i = 1; i <= numero; i++) {

if (i % 3 === 0 && i % 5 === 0) {

console.log("FizzBuzz");

} else if (i % 3 === 0) {

console.log("Fizz");

} else if (i % 5 === 0) {

console.log("Buzz");

} else {

console.log(i);

}

}

}